

Article

Visible Light Spectrum Extraction from Diffraction Images by Deconvolution and the Cepstrum

Mikko E. Toivonen , Topi Talvitie , Chang Rajani  and Arto Klami 

Department of Computer Science, University of Helsinki, 00560 Helsinki, Finland; topi.talvitie@iki.fi (T.T.); chang.rajani@helsinki.fi (C.R.); arto.klami@helsinki.fi (A.K.)

* Correspondence: mikko.e.toivonen@helsinki.fi

Abstract: Accurate color determination in variable lighting conditions is difficult and requires special devices. We considered the task of extracting the visible light spectrum using ordinary camera sensors, to facilitate low-cost color measurements using consumer equipment. The approach uses a diffractive element attached to a standard camera and a computational algorithm for forming the light spectrum from the resulting diffraction images. We present two machine learning algorithms for this task, based on alternative processing pipelines using deconvolution and cepstrum operations, respectively. The proposed methods were trained and evaluated on diffraction images collected using three cameras and three illuminants to demonstrate the generality of the approach, measuring the quality by comparing the recovered spectra against ground truth measurements collected using a hyperspectral camera. We show that the proposed methods are able to reconstruct the spectrum, and, consequently, the color, with fairly good accuracy in all conditions, but the exact accuracy depends on the specific camera and lighting conditions. The testing procedure followed in our experiments suggests a high degree of confidence in the generalizability of our results; the method works well even for a new illuminant not seen in the development phase.

Keywords: spectrum; spectrometer; cepstrum; deconvolution; diffraction imaging



Citation: Toivonen, M.E.; Talvitie, T.; Rajani, C.; Klami, A. Visible Light Spectrum Extraction from Diffraction Images by Deconvolution and the Cepstrum. *J. Imaging* **2021**, *7*, 166. <https://doi.org/10.3390/jimaging7090166>

Academic Editors: Alois Herkommer and Norimichi Tsumura

Received: 5 June 2021

Accepted: 24 August 2021

Published: 28 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In everyday speech, we use the word “color” to refer to the perceived appearance of objects, using named color categories such as “red” that represent a broad range of visible appearances. *Color spaces* such as RGB or CMYK can be used to define specific colors, but there is no universal mapping from perceived color to these spaces since the former depends on the lighting and the observer. Both natural (humans and animals) observers and digital sensors capture light—and hence colors—differently. The CIE L*a*b* color space, based on a standard observer, is a good approximation for a perceptually uniform color space, but for still more comprehensive characterization of color, we can consider the *spectrum* an object reflects. Together with the light spectrum and a perception model, this is enough for determining the perceived color in all conditions.

The conventional way of determining the absorption spectrum is *spectroscopy*, typically carried out by sensors that record the intensity of light for each wavelength separately to form the full intensity spectrum. For the measurement of light, a spectroradiometer is typically used, while spectrophotometers are used to measure the reflectance spectra of objects [1]. We, instead, considered the task of determining the light spectrum with imaging. Imaging-based methods have also been studied for a long time [2–6], but the classical solutions are relatively complex optical devices. In contrast to these works, we used ordinary digital CMOS sensors designed to mimic human perception. We acquired an image with a camera that only records the color information in the RGB color space in some lighting conditions, and the task was to infer the full reflectance spectrum of the imaged object from a single image or multiple images of the same scene, using a computational

algorithm that reconstructs the spectrum that is not directly measured by the device. Compared to the classical approaches, this has the advantage of having significantly lower cost as complicated hardware is replaced with computational algorithms.

The most relevant previous work considered the task of estimating the reflectance spectrum with a computational method from an image acquired with a monochromatic or a trichromatic camera, typically equipped with passive color filters. Several methods combining multispectral imaging with algorithms that estimate the reflectance spectrum (and subsequently, the color) by explicitly assuming it to be smooth were proposed [7–10]. Similarly, the observation that most of the variance in the reflectance spectra can be explained by a small number of principal components [7,11] has been used for estimating them using principal component analysis of multispectral images [8,12]. Other works assumed that Gaussian distributions are good estimates for reflectance spectra [10,13] and used these for the recovery of the spectrum. An alternative to using color filters is to use multiple illuminants for multispectral image capture [14]. A dedicated hyperspectral camera can also be used for estimating the radiance spectrum, as well as the reflectance spectrum with an additional calibration for the illuminant spectrum [15]. While recovering the reflectance spectrum from a hyperspectral image can be easy in controlled environments, greater care is needed when imaging uncontrolled outdoor scenes [16].

Our approach shares the conceptual motivation of the previous research covered above, but uses a simpler optical construction and makes less assumptions about the spectra. In particular, we used a purely passive add-on device attached to ordinary consumer cameras and did not require knowing its spectral response, and our computational method requires only mild smoothness assumption, in contrast to the strong parametric assumptions of, e.g., [8,10,12,13]. The core idea of our approach is in the use of a specific diffractive element—a transmissive diffraction grating—placed in front of the camera sensor. The diffraction displaces light at different wavelengths spatially across the image area. By carefully designing the diffraction element, which also includes a suitably placed field stop, we can control the diffraction in a manner that allows computational reconstruction of the spectrum. By using broad-spectrum lighting (e.g., halogen illumination) and assuming a single reflectance spectrum per object, we can then provide an estimate for the full radiance spectrum of the imaged object. Diffraction images have previously been used, e.g., for the acquisition of hyperspectral images [17–20], camera sensor response calibration [21,22], multiplane imaging [23], and single-camera 3D imaging [24,25]. Similar to our method, Javoršek et al. [26] used a diffraction grating together with a biconvex lens for camera sensor spectrum response estimation. These works focused on specific imaging configurations, whereas we provide the first practical method for estimating the radiance spectrum with demonstrated high accuracy across different lighting conditions and imaging devices.

This paper introduces both the optical element that performs diffraction in a controlled manner and two alternative computational algorithms for reconstructing the spectrum. Both methods are based on supervised machine learning trained to reconstruct the spectrum from specific forms of signal representation, but differ in terms of the initial processing of the diffraction image. The first approach uses the deconvolution operation to revert the diffraction, whereas the latter relies on computing the cepstrum of the image [27].

We empirically validated the approach with experiments specifically designed to highlight its generality in terms of imaging devices and lighting conditions. Rather than demonstrating the technique using a single prototype device, we conducted experiments using three different types of cameras—a smartphone, a DSLR, and a camera module for embedded imaging applications—using three different lighting conditions. By comparing against the ground truth spectrum obtained by a dedicated hyperspectral camera, we show that the approach can estimate the spectrum under two test broad-spectrum lighting conditions and for all three cameras. The exact accuracy naturally depends on the camera and the lighting, and for some conditions, the reconstructions were less accurate especially for longer wavelengths for which the sensitivity of typical camera sensors is worse; how-

ever, the results demonstrated that the same general principle works for all cases. We used a strict testing policy where the final test set was kept secret from the persons conducting all computational experiments until the final evaluation. The main result was that the proposed approach can estimate the reflectance spectrum, and consequently the color, in the studied lighting conditions. In terms of the computational algorithms, the method based on the deconvolution operation was found to be more accurate and is to be preferred in practice.

The rest of the manuscript is organized as follows. We first introduce the foundations for diffraction imaging and the computational analysis of diffraction images in Section 2. Building on this, we proceed to introduce our concrete approach, first explaining the computational algorithms for reconstructing the spectrum in Section 3 and the specific imaging setup in Section 4, followed by additional details on the implementation of the computational approach and the testing setup in Section 5. Section 6 presents the results with the discussion, followed by the conclusions in Section 7.

2. Foundations

Our approach was based on a combination of an imaging device that disperses light with a diffractive element to distribute the spectral information spatially and a computational method for reconstructing the spectrum from the resulting image. We first provide a brief overview to diffraction and explain the specific form of diffraction images we used, then discuss the general principle for computational modeling of diffraction images by representing them as the results of convolution with a diffraction kernel. This provides the necessary background for understanding the specific computational methods presented in Section 3.

2.1. Diffraction Imaging

In our work, the phrase *diffraction imaging* refers to taking photographs with a photographic camera that has a transmissive diffraction grating placed on its optical path. Previously, diffraction images have been used for spectral and hyperspectral imaging [17,20,28,29], as part of the so-called *computed tomography spectral imaging (CTSI)* approach, as well as for estimating camera spectral responses [21]. In these uses, the purpose of the diffraction grating is to disperse light into multiple dislocations at the camera's sensor plane in relation to the wavelength of light. Assuming perpendicularly incident rays to a regular transmissive diffraction grating (We use "regular diffraction grating" to refer to a regular pattern of a rectangular or parallelogram grid. This is in contrast to an arbitrary diffraction grating pattern, for which diffraction can be described by Fourier diffraction.), the relation between the wavelength and direction of diffraction in one dimension is described by $D \sin \theta_n = n\lambda$, where D is the spacing between the grating grooves, θ_n is the angular direction of the diffracted light, n is an integer denoting the diffraction order, and λ is the wavelength of light. For small θ_n , we can assume the small angle approximation form of $D\theta_n = n\lambda$, giving us a linear relation between the direction of diffracted light and the wavelength of light. Two-dimensional diffraction gratings function in a similar way, producing diffraction components in directions defined by the axes of the diffraction grating grooves. While the extent of diffraction, i.e., the extent of spatial dislocation, is approximately defined by $D\theta_n = n\lambda$, the exact geometry of the diffraction grating determines the shape of the diffraction kernel $h(\cdot, \cdot, \lambda)$. Figure 1 shows two diffraction images of point-like illuminants: one for monochromatic red light and one for broadband white light. The diffraction image in Figure 1a reveals the shape of the diffraction kernel for a single wavelength λ_m , while Figure 1b reveals the shape of the diffraction kernel by superimposing all the wavelengths onto a single image.

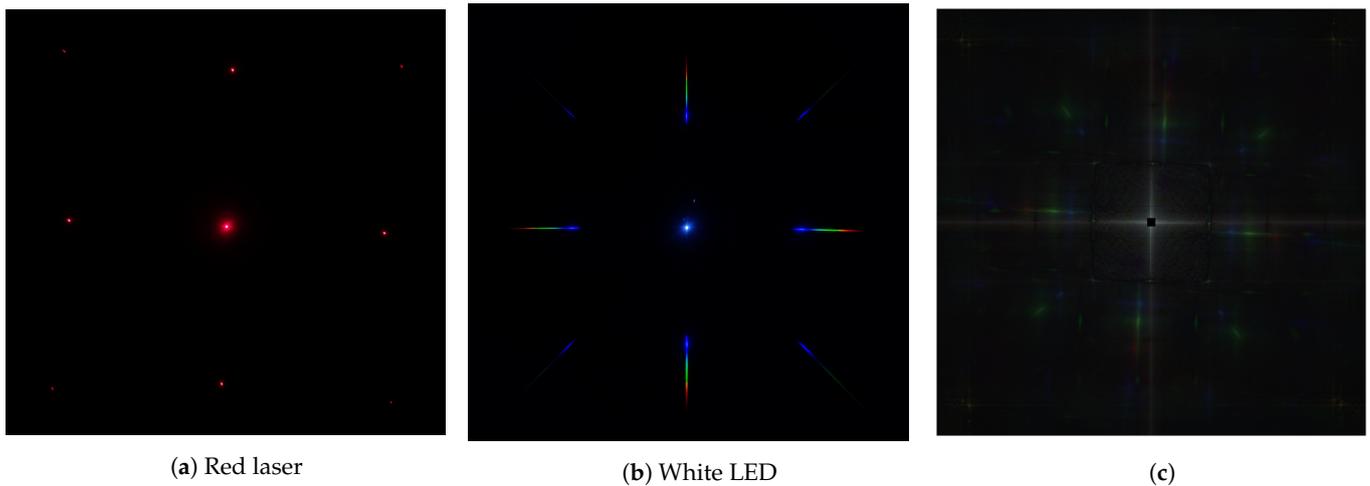


Figure 1. Example diffraction images of point-like light source for (a) a red laser and (b) a white LED, and (c) an example cepstrum image. The diffraction pattern becomes evident when a point-like light source is imaged through a diffraction grating in a dark environment. For (a) the red laser, the light is essentially monochromatic, so the diffraction pattern is also point-like. For (b), the intensity of the diffraction in for each wavelength is dependent on the diffraction grating efficiency and the spectrum of the light. The diffraction grating used to capture these images is composed of two regular one-dimensional diffraction gratings with their axes at approximately a 90° angle. Due to imperfect alignment, the diffraction pattern is not entirely symmetrical. (c) The example mean cepstrum image formed from the mean of 60 cepstrum images of a diffused RGB illuminant captured using a Canon M3 camera combined with a diffraction grating, a Sigma 35 mm lens, and a field stop construct attached in front of the lens. The resulting cepstrum image was shifted such that the center of the cepstrum image is displayed in the center of the image. The center of the image was masked out as it contains high intensity pixels that are not relevant to spectrum extraction. Figure best viewed in color.

For a naive diffraction image, acquired using a diffraction grating with no additional optical elements, the zeroth-order diffraction components, i.e., for $n = 0$, are superimposed with higher-order diffraction components, i.e., for $n > 0$; a specific pixel recording may correspond to either zeroth-order component or a higher-order component with dislocation. Even though some methods attempt to handle this by using global [17] or local [30] image structure, we prefer using field stops, similar to [19], to eliminate the overlaps. By limiting the scene to what is visible through the field stop, we can enforce all higher-order diffraction components to be outside of the area covered by the zeroth-order components; see Figure 2 for an illustration of the images used in our setup.

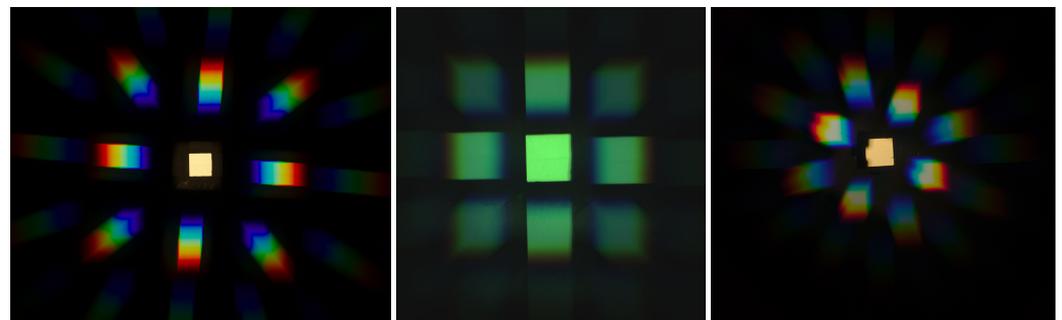


Figure 2. Diffraction images captured using three cameras: Canon M3, Raspberry Pi HQ camera, Samsung S10 main camera. The Raspberry Pi HQ image appears excessively green because it has not been corrected for white balance, which is mostly relevant for presentation purposes only.

2.2. Modeling Diffraction Images

Mathematically, the production of a diffraction image can be expressed as a convolution with the diffraction kernel of the grating element. We assumed that the image consists of only one color spectrum, which can be achieved using a black mask, i.e., a

mask essentially void of color. We denote the convolution of two functions f and g by $f * g$ and frequently use Fourier transforms for manipulating and analyzing convolutions because of the convenient property: a Fourier transform of the convolution can be computed as the product of the transforms for the two functions. That is, the Fourier transform of $(f * g)(x, y)$ is $F(f_x, f_y)G(f_x, f_y)$, where $F(f_x, f_y)$ is the Fourier transform of $f(x, y)$ and $G(f_x, f_y)$ of $g(x, y)$ and f_x and f_y are the frequency parameters for the x and y dimensions, respectively. Denoting wavelengths by λ and spatial coordinates with variables x and y , for the diffraction image $d(x, y)$, illuminant intensity distribution $i(\lambda)$, scene reflectance $r_\lambda(x, y) = r(x, y, \lambda)$, camera spectral sensitivity $s(\lambda)$, and the diffraction kernel $h_\lambda(x, y) = h(x, y, \lambda)$, we have for each color $c \in \{R, G, B\}$:

$$d_c(x, y) = \int_{\lambda_{min}}^{\lambda_{max}} i(\lambda)s_c(\lambda)(h_\lambda * r_\lambda)(x, y)d\lambda \tag{1}$$

This formulation assumes flat illumination $i(\lambda)$ such that illumination intensity is spatially constant and is only a function of wavelength λ . The diffraction kernel $h_\lambda(x, y)$ is assumed to be linear w.r.t. to the intensity of light and isoplanatic and oriented parallel to the imaging sensor. Furthermore, the model, and subsequent models herein, assumes the image is free of noise. This does not strictly hold in real imaging conditions due to electronic and shot noise, but in practice, the overall method will later be shown to work well even if taking images without any noise cancellation functionality.

Since we are not concerned with estimating the reflectance, but instead just the spectra, because reflectance spectra can be calculated from a given reflected spectrum by knowing the spectral power distribution of the illuminant, we can reduce the product of the illuminant intensity distribution and scene reflectance to $u_\lambda = u(x, y, \lambda) = i(\lambda)r(x, y, \lambda)$, so that we have:

$$d_c(x, y) = \int_{\lambda_{min}}^{\lambda_{max}} s_c(\lambda)(h_\lambda * u_\lambda)(x, y)d\lambda. \tag{2}$$

For a monochromatic illuminant at wavelength λ_m , the diffraction image reduces to a single convolution defined by the diffraction kernel at λ_m and the product of the sensor sensitivity, scene reflectance, and illuminant intensity at λ_m :

$$d_{c,\lambda_m}(x, y) = s_c(\lambda_m)(h_{\lambda_m} * u_{\lambda_m})(x, y).$$

After calibration, estimating the wavelength λ_m would then be a simple matter of determining the extent to which the higher-order diffraction components have been displaced in relation to the zeroth-order diffraction component. Unfortunately, this simple computation does not generalize directly for arbitrary illuminants. Instead, we need dedicated computational algorithms, explained next.

3. Method

In this section, we present two computational algorithms for reconstructing the spectrum from diffraction images acquired using the imaging setup explained later in Section 4. The first approach is based on directly reverting the convolutional model (2) for the diffraction image using the *deconvolution* operation, whereas the latter computes the *cepstrum* of the image and uses it for extracting the spectrum, building on the desirable mathematical properties of cepstra [27]. Both methods contain some elements that are trained using supervised machine learning, based on a collection of diffraction images for which ground truth spectra are available, carried out using appropriately chosen algorithms for each of the methods.

3.1. Algorithm 1: Deconvolution

Since we modeled the diffraction images $d_c(x, y)$ as convolutions of $u(x, y)$, a natural approach for extracting the spectrum is to invert the convolution with the *deconvolution* op-

eration. Algorithm 1 builds on this basic intuition, solving the deconvolution problem—for the specific types of diffraction images considered here—in the space of Fourier transforms.

Assume that the image consists of only one color spectrum with varying brightness; more exactly, there exists a single spectrum f and a per-pixel scene brightness map b such that the hyperspectral image $u(x, y, \lambda)$ is given by the product $f(\lambda)b(x, y)$. For the derivation of the algorithm, we assumed this to hold, and for practical imaging, the assumption can be satisfied by using a black mask that blocks the areas outside the color patch. By substituting the product form of u into (2), we obtain:

$$d_c(x, y) = \int_{\lambda_{min}}^{\lambda_{max}} s_c(\lambda)f(\lambda)(h_\lambda * b)(x, y)d\lambda.$$

By interchanging the order of integration and convolution, we see that $d_c = p_c * b$, where:

$$p_c(x, y) = \int_{\lambda_{min}}^{\lambda_{max}} s_c(\lambda)f(\lambda)h_\lambda(x, y)d\lambda. \tag{3}$$

The convolution kernel p_c is the diffraction pattern of a point with spectrum f generated by the whole system, taking into account the diffraction kernel h of the diffraction grating, and the camera spectral sensitivity s_c . Figure 1b shows an example of such a diffraction pattern.

We recovered the color spectrum f in two phases. In the first phase, we solved p_c for all colors c from the convolution equation $d_c = p_c * b$, using deconvolution. The deconvolution was possible because we knew both d_c and b : the diffraction image d_c is directly measured by the camera and the brightness map b can also be extracted from the intensities of the center component of the diffraction image (provided that the diffraction components do not overlap). In the second phase, we extracted the spectrum f from the system diffraction pattern p_c using the fact that for each wavelength $\lambda \in [\lambda_{min}, \lambda_{max}]$, the monochromatic diffraction pattern h_λ is nonzero only close to a small number of isolated peaks, one per diffraction component. Thus, by combining (3) with suitable smoothness assumptions for s_c and f , we can estimate $f(\lambda) \approx S(x, y, \lambda)p_c(x, y)$ for each peak (x, y) of h_λ , where the coefficients $S(x, y, \lambda)$ depend only on the camera spectral response s_c and grating pattern h_λ , and hence can be calibrated per imaging system. For many trichromatic cameras, the camera spectral response s_c for each channel can be assumed to be smooth due to the Luther condition [31]. The spectrum f can also be assumed smooth by assuming a smooth material reflectance spectrum [32–34] and a smooth spectral power distribution of the illuminant. The latter may not hold for example if using artificial LED or fluorescent illumination.

Let us now consider the first phase of solving p_c from the convolution equation $d_c = p_c * b$. Because convolution corresponds to pointwise multiplication in the Fourier space, it holds that $\hat{d}_c = \hat{p}_c \hat{b}$, where \hat{d}_c , \hat{p}_c , and \hat{b} are the two-dimensional Fourier transforms of d_c , p_c , and b , respectively, and the arithmetic on the functions is carried out pointwise. In the absence of noise, we could simply solve $\hat{p}_c = \hat{d}_c / \hat{b}$ and obtain p_c by applying the inverse Fourier transform to \hat{p}_c . However, this kind of naive deconvolution does not typically handle noise in d_c and b well.

We made the deconvolution more noise resistant using regularization: we obtained the estimate p'_c for p_c by minimizing a cost function:

$$p'_c = \underset{g: \mathbb{R}^2 \rightarrow \mathbb{R}}{\operatorname{argmin}} \left(\|g * b - d_c\|_2^2 + \|k * g\|_2^2 \right),$$

where $\|\cdot\|_2^2$ is the absolute square integral over the domain and k is a real-valued regularization kernel. By the Parseval theorem, we can express this for the Fourier transforms of the functions using pointwise multiplications instead of convolutions:

$$\hat{p}'_c = \operatorname{argmin}_{g: \mathbb{R}^2 \rightarrow \mathbb{R}} \left(\|\hat{g}\hat{b} - \hat{d}_c\|_2^2 + \|\hat{k}\hat{g}\|_2^2 \right).$$

If we allow g to be a complex function, then the minimization task can be carried out for each point $\xi = (\xi_x, \xi_y)$ in the frequency domain independently:

$$\hat{p}'_c(\xi) = \operatorname{argmin}_{z \in \mathbb{C}} \left(|z\hat{b}(\xi) - \hat{d}_c(\xi)|^2 + |\hat{k}(\xi)z|^2 \right).$$

Each minimizing value $\hat{p}'_c(\xi) \in \mathbb{C}$ can be found by solving for the value at which the partial derivatives of the cost function with respect to the real and imaginary part of z are zero. This calculation yields the result:

$$\hat{p}'_c(\xi) = \frac{\overline{\hat{b}(\xi)}\hat{d}_c(\xi)}{|\hat{b}(\xi)|^2 + |\hat{k}(\xi)|^2}. \tag{4}$$

The estimate p'_c obtained from this $\hat{p}'_c(\xi)$ using the inverse Fourier transform is actually a real function and hence a solution to the original minimization problem. This is because b , d_c , and k are real functions, which means that the reality conditions $\hat{b}(\xi) = \overline{\hat{b}(-\xi)}$, $\hat{d}_c(\xi) = \overline{\hat{d}_c(-\xi)}$, and $\hat{k}(\xi) = \overline{\hat{k}(-\xi)}$ hold, and they imply the reality condition for \hat{p}'_c as well.

Figure 3 shows an example of a diffraction pattern p obtained using 2D deconvolution (4) from a diffraction image. While the diffraction pattern is clearly visible, the image contains a significant noise pattern outside the diffraction components, which may skew the signal in the actual diffraction patterns. To avoid this, we chose to limit the degrees of freedom in the deconvolution by using a 1D deconvolution instead.

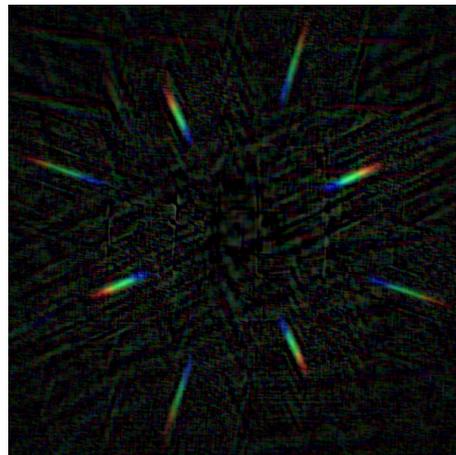


Figure 3. The 2D deconvolution result obtained from a diffraction image using a Dirac delta distribution as the regularization kernel k . In this case, the regularization term $|\hat{k}(\xi)|^2$ in (4) is a constant (independent of ξ). The computations were discretized using the discrete Fourier transform.

From the diffraction pattern obtained using 2D deconvolution, it is easy to detect the diffraction directions by finding the angles in which the total brightness is highest. Let us consider the modified instance of the problem where we detected the diffraction directions, rotated the diffraction image such that one of the first-order diffractions is made horizontal, and cropped out all the other diffraction components from the image. Now, the problem is 1D in the sense that the diffraction pattern $p_c(x, y)$ should be zero if $y \neq 0$.

Let us now reformulate the deconvolution optimization problem in terms of 1D convolution. Consider the diffraction pattern p_c to be a one-dimensional function, replacing $p_c(x, 0)$ by $p_c(x)$. Denote $b_y(x) = b(x, y)$ and $d_{c,y}(x) = d_c(x, y)$. Now, the estimate p'_c for p_c is obtained by minimizing a cost function as follows:

$$p'_c = \operatorname{argmin}_{g: \mathbb{R} \rightarrow \mathbb{R}} \left(\int_{-\infty}^{\infty} \|g * b_y - d_{c,y}\|_2^2 dy + \|k * g\|_2^2 \right). \tag{5}$$

If we allow g to be a complex function and rewrite the convolutions using pointwise multiplications of 1D Fourier transforms, we obtain that:

$$\hat{p}'_c(\xi) = \operatorname{argmin}_{z \in \mathbb{C}} \left(\int_{-\infty}^{\infty} |z \hat{b}_y(\xi) - \hat{d}_{c,y}(\xi)|_2^2 dy + |\hat{k}(\xi)z|_2^2 \right).$$

By solving for the point at which the partial derivatives are zero, we obtain the optimum:

$$\hat{p}'_c(\xi) = \frac{\int_{-\infty}^{\infty} \overline{\hat{b}_y(\xi)} \hat{d}_{c,y}(\xi) dy}{\int_{-\infty}^{\infty} |\hat{b}_y(\xi)|^2 dy + |\hat{k}(\xi)|^2}. \tag{6}$$

Similarly to (4), the estimate p'_c obtained from this \hat{p}'_c using the inverse Fourier transform is actually a real-valued function, and thus, it is also a solution to the original minimization problem (5).

3.2. Algorithm 2: Cepstrum

Instead of solving the deconvolution, we can alternatively approach the problem by using the cepstrum operation that has been used, e.g., in pitch determination [35], signal detection and extraction [36], image registration [37,38], and echo detection and removal [39]. The cepstrum operator has interesting properties that make it convenient for relative spectra extraction. Following [27], we summarize the most notable properties as:

- Scaling, i.e., multiplication, of the diffraction image affects only the center value of the convolution image;
- Biasing, i.e., addition, the diffraction image leads to a biasing in the convolution image that is nonlinearly relational to the biasing factor;
- Convolution in the diffraction image can be expressed as the sum of two cepstrum images;
- Rotation of the diffraction image results in a rotation of the cepstral image;
- The cepstrum image is invariant to shifts in the diffraction image.

Adjusting the intensity of the illuminating light or the exposure time of the diffraction image is essentially a scaling operation, which does not, in theory, affect the cepstrum image other than at the center of the cepstrum image. The core phenomenon of diffraction can be modeled accurately using the convolution operation, which in the cepstrum image can be expressed as the sum of two cepstrum images. The rotation property helps in finding the rotation of the diffraction grating in relation to the camera sensor’s axes directly from the computed cepstrum image. The cepstrum image’s invariance to shifts in the diffraction image make it easy to automatically locate and extract the salient parts of information from the cepstrum image. An example of a cepstrum image formed from the mean of 60 individual cepstrum images can be seen in Figure 1c, where the diffraction grating’s axes are at an angle with respect to the camera sensor’s axes.

In defining the cepstrum operation, we first denote \mathcal{F}_{2D} and \mathcal{F}_{2D}^{-1} as the two-dimensional Fourier and inverse Fourier transformations, respectively. The two-dimensional cepstrum

\mathcal{C}_{2D} , also referred to as the cepstral image, of a two-parameter function $g(x, y)$, and denoting $G(f_x, f_y) = \mathcal{F}_{2D}(g(x, y))$, is then defined as:

$$\begin{aligned} \mathcal{C}_{2D}(g)(x, y) &= \mathcal{F}_{2D}^{-1}(\log(|\mathcal{F}_{2D}(g(x, y))|)) \\ \mathcal{C}_{2D}(g)(x, y) &= \mathcal{F}_{2D}^{-1}(\log(|G(f_x, f_y)|)) \\ \mathcal{C}_{2D}(g)(x, y) &= G_C(x, y). \end{aligned}$$

By first denoting the product of the spectral sensitivity $s(\lambda)$, scene reflectance $r_\lambda(x, y) = r(x, y, \lambda)$, and illuminant intensity $i(\lambda)$ as $q_\lambda(x, y) = q(x, y, \lambda) = s(\lambda)r(x, y, \lambda)i(\lambda)$, the two-dimensional cepstrum \mathcal{C}_{2D} of the diffraction image is:

$$\mathcal{C}_{2D}(d_c)(x, y) = \mathcal{F}_{2D}^{-1}(\log(|\mathcal{F}_{2D}(\int_{\lambda_{min}}^{\lambda_{max}} h_\lambda(x, y) * q_\lambda(x, y))|)).$$

Since the Fourier transform of a convolution is the product of the Fourier transforms, this becomes:

$$\mathcal{C}_{2D}(d_c)(x, y) = \mathcal{F}_{2D}^{-1}(\log(|\int_{\lambda_{min}}^{\lambda_{max}} H_\lambda(f_x, f_y) Q_\lambda(f_x, f_y)|)),$$

where $H_\lambda(f_x, f_y)$ and $Q_\lambda(f_x, f_y)$ are 2D Fourier transformations of $h_\lambda(x, y)$ and $q_\lambda(x, y)$, respectively. In the case of a single monochromatic light source $\lambda = \lambda_m$, the cepstrum image becomes:

$$\begin{aligned} \mathcal{C}_{2D}(d_c)(x, y) &= \mathcal{F}_{2D}^{-1}(\log(|H_{\lambda_m}(f_x, f_y) Q_{\lambda_m}(f_x, f_y)|)) \\ \mathcal{C}_{2D}(d_c)(x, y) &= \mathcal{F}_{2D}^{-1}(\log(|H_{\lambda_m}(f_x, f_y)|)) + \mathcal{F}_{2D}^{-1}(\log(|Q_{\lambda_m}(f_x, f_y)|)) \\ \mathcal{C}_{2D}(d_c)(x, y) &= \mathcal{C}_{2D}(h_{\lambda_m}(x, y)) + \mathcal{C}_{2D}(q_{\lambda_m}(x, y)). \end{aligned}$$

Thus, for a known diffraction kernel $h_\lambda(x, y)$, solving for $\mathcal{C}_{2D}(q_{\lambda_m})(x, y)$ is possible by computing the 2D cepstrum of the diffraction image. In practice, the computation is performed over discrete representations of diffraction images. To find the shape of the diffraction kernel, i.e., the locations where $h_\lambda(x, y)$ is mostly nonzero, we can use the average of the cepstra of a collection of diffraction images, similar to [17]. For regular diffraction gratings, the diffraction kernel is relatively simple and can be assumed to be known a priori with the additional assumption that the camera is not sensitive to wavelengths below and above λ_{min} and λ_{max} .

For monochromatic light, we have an analytic relationship between the values of the cepstrum image and $q_\lambda(x, y)$, so that the relationship between the cepstrum image and $q_{\lambda_m}(x, y)$ is linear, as demonstrated in [40]. By simulation, we found that for the case of two monochromatic lights, the relation is very nearly linear if the two wavelengths are not too close to each other. For the general case, i.e., a distribution of $q_\lambda(x, y)$ over λ , we found that the linearity assumption between the cepstrum image and $q_\lambda(x, y)$ does not hold true. While estimating a higher-order analytic estimate is a difficult ordeal [41], the cepstrum image has interesting properties noted before that would make the estimation of $q_\lambda(x, y)$ convenient. Despite these convenient properties, there are two drawbacks: the lack of an analytical relationship between the cepstrum image and $q_\lambda(x, y)$ and the noisy nature of the cepstrum images.

The noise can be reduced by computing the mean of multiple cepstrum images, by capturing multiple photographs. We did not investigate the factors effecting noise thoroughly, but did study the relationship between the noise and the number of cepstrum images being averaged. One-hundred forty images were captured using the setup involving a Canon M3, described later in detail in Section 4, and the mean of all of them was used as a proxy for a noiseless image. The number of cepstrum images to be averaged was tested over the range from 1 to 80, and for each test, 100 iterations were made by randomly sampling a combination of images from the set of 140 images. The cepstrum profiles were

also extracted from the proxy ground truth and the averaged cepstrum image, as described in Section 3.2.1.

Figure 4 shows the errors on a semilog scale, displaying an initially rapidly decreasing rate from 1 to approximately 20 images. The error rates continued to decline with the diminishing rate in the marginal error. Based on this result, we captured a minimum of 10 images for each color sample for the computation of the cepstrum profiles.

3.2.1. Cepstrum Profile

As shown in an example cepstrum image in Figure 1c, most of the cepstrum image is dark background with higher intensity values at the axes of the image and along the cepstrum profile lines that follow the axes of the diffraction grating. If the diffraction grating's axes are perfectly aligned with the camera sensor's axes, the cepstrum profiles would be superimposed on the cepstrum image axes with the high intensity values at these axes dominating over the cepstrum profile. Based on this observation, the diffraction grating for each camera was rotated slightly with respect to the camera sensor's axes so that the cepstrum profile lines would not align with the cepstrum images axes. The center of the cepstrum image contains information mostly about the center of the diffraction image and is therefore filtered out, i.e., a square area corresponding to the size of the center image in the diffraction image is set to zero in the cepstrum image. To extract the cepstrum profile, starting from the center to the edge of the cepstrum image, the `profile_line` function from the Python package `scikit-image` [42] was used, with a line width of 5 and reducing by sum. The angle of rotation of the diffraction grating was estimated using Bayesian optimization [43] by maximizing the sum of the profile line and searching from an angle range provided by manual inspection. A total of 8 cepstrum profiles corresponding to the first order diffraction components are visible in Figure 1c. However, the diagonal diffraction components had significantly lower intensity and were not used: only the cepstrum profiles for the four major direction diffraction components were extracted and used throughout.

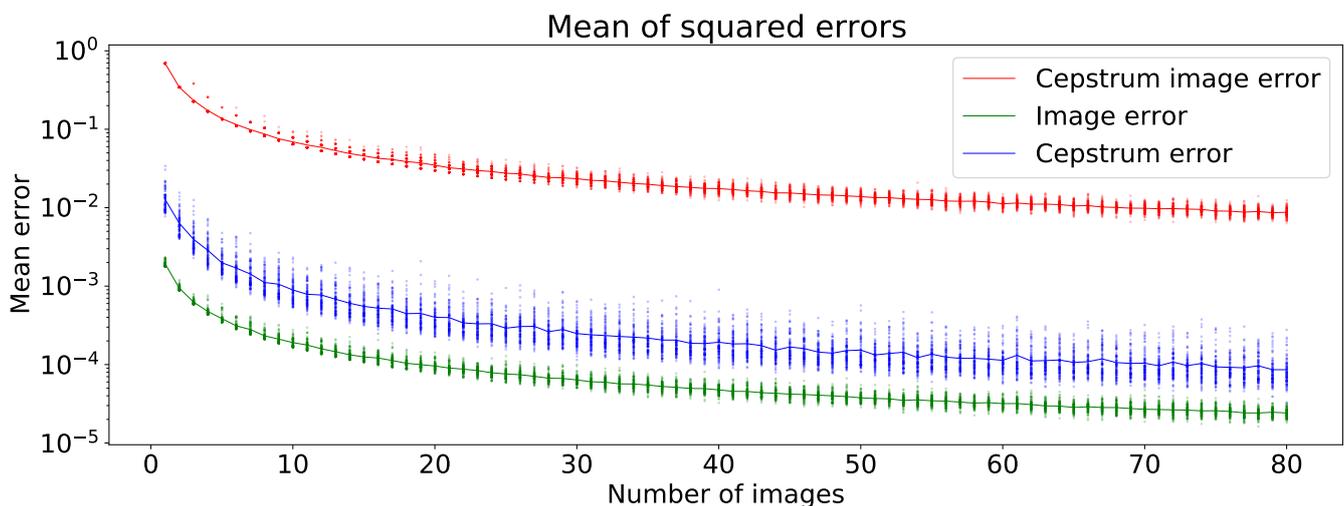


Figure 4. Cepstrum profile and cepstrum image mean-squared errors as a function of the number of cepstrum images from which the mean cepstrum image is computed. The error is the difference between the mean cepstrum image compared to the the mean cepstrum image computed from 140 cepstrum images, i.e., the proxy ground truth. The cepstrum profile error (labeled cepstrum error) is significantly smaller than the cepstrum image error.

4. Imaging

In this section, we describe the imaging setups for the different cameras used to obtain the diffraction images. Three cameras were used to image the same set of color samples: a Canon M3 digital single-lens reflex (DSLR) equipped with a Sigma 35mm F1.4 DG Art series

lens (adapted using a EOS to EOS-M adapter), a Raspberry Pi HQ camera (Sony IMX477 sensor) with a 4mm CS mount lens, and a Samsung S10 smartphone's main camera at $f/2.4$ aperture. For the collection of the training set, we used 15 discs with 18 color samples in each disc, for a total of 270 color samples. For the testing set, we used 2 discs with 18 color samples, for a total of 36 color samples. The discs were custom made, and the color samples were paint samples from three different paint manufacturers such that samples from two manufacturers (Tikkurila and Värisilmä) were used for the training set and one manufacturer (Teknos) was used for the test set. Each of the color samples were imaged using different illuminants in a darkened room. For the training set, three illuminants were used: a full-spectrum LED, halogen lamps, and a white LED, whose spectra are shown in Figure 5a. For the test set, two illuminants were used: halogen lamps and cold white LED, whose spectra are shown in Figure 5b. The test set thus included images captured using an illuminant outside of the training set to test the different models' generalizability for illuminants not in the training set. Furthermore, all of the actual samples were different.

4.1. Diffraction Image

To capture diffraction images using the three cameras, each of the cameras was equipped with 2D transmissive regular diffraction grating and a field stop. Each camera was equipped with the same diffraction grating with a grating constant of approximately 191 grooves per millimeter. The diffraction gratings were placed on the optical path of the camera–lens system before the field stop. For each of the cameras, custom field stop assemblies were constructed. For the Canon M3, seen in Figure 6, a tube of approximately 12 cm in length was mounted in front of the lens with a square aperture with each side measuring 8mm at the other end of the tube. The Raspberry Pi HQ camera was placed in a custom-made rectangular box measuring 4.5 cm by 5.2 cm by 10 cm, with a square aperture each side measuring 1cm at the end opposite to the camera. For the Samsung S10 a custom-made tube 3.5 cm in length was attached to a 17 mm screw thread onto a camera case attached to the phone. The field stop in the tube for the Samsung S10 was also square with each side measuring 6 mm. For all of the cameras, we attempted to center the square field stop at the central optical path.

The masking on the color discs, as seen in Figure 6, functioned effectively as a secondary field stop at the plane where the color samples were located. The masking was somewhat poor at absorbing incident light, so a significant amount of light was still scattered and reflected from the surface of the masking, producing a low contrast for paint samples with relatively low total reflectance. However, the masking solved the problem in which different diffraction components effectively see different parts of the scene when the field stop is placed between the camera and imaged plane i.e. the imaginary plane from which the image is formed at the camera sensor. Another diffraction imaging construction was presented in [20], where a collimating lens was used to overcome the problem of scene inconsistency between diffracted orders, which for the purpose of imaging plane paint samples would introduce unnecessary optical components to the construction.

4.2. Imaging Setup

The imaging setup for diffraction photos consisted of the camera setup as described in Section 4.1 for each camera, a tripod mount, illuminants, color sample discs, a rotating platform, and a remote trigger mechanism. Figure 6 shows the imaging setup with the Canon M3 camera, the full-spectrum LED illuminant, and one of the color sample discs. The cameras were mounted on a tripod such that each image was captured from the same location for each camera. The cameras were placed directly over the color samples at appropriate heights so that color samples were entirely visible in the captured images. Illuminants were placed to one side in relation to the color discs and at an angle to avoid possible specular reflections from the color samples. The color sample discs were placed on a rotating platform that was accurately rotated using a stepper motor via a timing belt. The Canon M3 was remote controlled using the *gphoto2* [44] software. The Raspberry

Pi HQ camera was attached to a Raspberry Pi 4 computer using the CSI interface and controlled using custom software. The Samsung S10 was remote controlled via Bluetooth using custom software on an ESP32 SOC device. In all cases, the purpose of the remote control was to trigger the photo capture without physical contact with the cameras and for the automation of image capture. The rotating platform also enabled the automation of image capturing with manual labor required only for changing the color sample discs, illuminants, and cameras.

For the Canon M3, an aperture size of $f/10$ was used for all images. The Raspberry Pi HQ camera was equipped with a nonadjustable aperture lens. For the Samsung S10, we used the main camera with $f/2.4$, based on the principle of favoring smaller apertures. All images were captured in RAW Bayer format, to maintain a linear dependence between the recorded sensor values and the intensity of light, with manual settings and manual focus. The exposure times were manually maximized such that for the first color sample, which was white, it did not produce overexposed images. All possible noise reduction and image enhancement options were switched off for image capture and recording.

For the purpose of extracting ground truth spectra, the Specim IQ hyperspectral camera was used to capture hyperspectral images of the color sample discs. The color sample discs were placed into a custom jig, and the Specim IQ camera was mounted stationary on a tripod. The illuminants were placed approximately in the same geometry, i.e., the same angle w.r.t. to the color samples. Each color sample disc and illuminant combination was imaged using the Specim IQ hyperspectral camera producing a total of 45 hyperspectral images for the training set and 4 for the testing set. The ground truth spectra were obtained from the hyperspectral images by computing the mean value in 5×5 neighborhoods, separately for each channel. That is, for each color patch, we computed:

$$y(\Lambda) = \frac{\sum_{(x,y) \in W} HSI(x,y,\Lambda)}{25}$$

where $HSI(x,y,\Lambda)$ is the value of the hyperspectral image at image coordinates (x,y) and spectral channel Λ and W is a 5×5 pixel subregion.

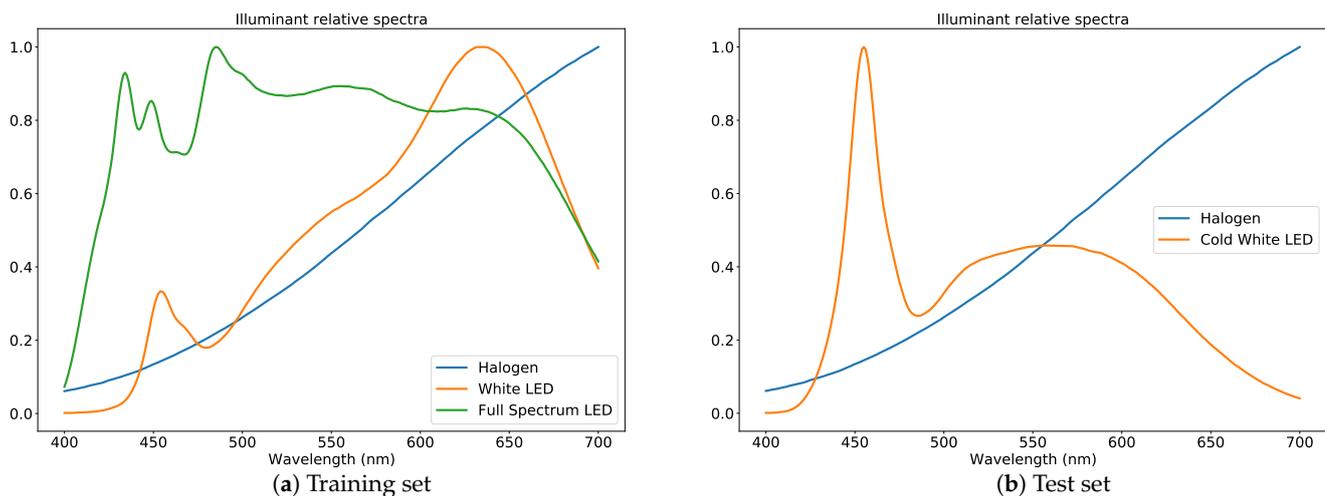


Figure 5. Relative spectra of the illuminants used in capturing diffraction images for the training and test set, respectively.



Figure 6. Imaging setup for the Canon M3 camera and the full-spectrum LED illuminant. Other cameras were setup in a similar way, adapting to the camera. The color targets were on a paper disc with a dark mask over each color patch exposing a 1cm by 1cm area of the color patch. The light was mounted to the side of the camera at an angle to reduce the effect of reflection from the surfaces of the color patches. The paper discs were mounted on to a acrylic disc and secured by three screws. The acrylic disc was rotated by a stepper motor via a timing belt.

4.3. Data

The data for training and testing were collected at separate times. The data consisted of diffraction images and ground truth hyperspectral images acquired with the Specim IQ hyperspectral camera. For the training dataset, 10 diffraction images for each combination of the 3 cameras, the 3 illuminants (whose spectra are shown in Figure 5a), and 270 color patches were collected. For the deconvolution methods, the entire diffraction image was used. For the purpose of the cepstral methods, a cepstrum profile tensor was computed for each color patch as an average cepstrum profile for each of the 4 major cepstrum directions using all of the diffraction images for a color patch. The cepstrum profiles were computed from raw Bayer images without any demosaicing, but instead stacking appropriate values to their own color channels, reducing the image resolution by a factor of four. The cepstrum profile tensor for each color patch was of size (number of directions, color channels, n_λ) = (4, 3, 301), where the number of wavelength channels n_λ was linearly interpolated to the inclusive range $\lambda_{min} = 400$ nm and $\lambda_{max} = 700$ nm at a 1 nm sampling interval. The cepstrum profile range was calibrated to coincide with the correct wavelength range by following the calibration procedure described in Section 5.1. Similarly, the ground truth spectrum data for each color patch were linearly interpolated to the same inclusive range, resulting in a spectrum vector of size 301. We were only concerned with relative spectra, so each ground truth spectrum \mathbf{y} was normalized using:

$$\mathbf{y}_{\text{normalized}} = \mathbf{y}w(\mathbf{y}), \text{ where } w(\mathbf{y}) = \frac{\sum_{\lambda=400nm}^{700nm} y(\lambda)c(\lambda)}{\sum_{\lambda=400nm}^{700nm} c(\lambda)}, \text{ and } y(\lambda) = \mathbf{y}_{\lambda-\lambda_{min}},$$

where $c(\lambda)$ is the mean of the absolute CIE RGB color matching function [45] values at wavelength λ , as shown in Figure 7b. The purpose of the normalization was to weigh the areas of the visible spectrum that were more significant to color perception as defined by the CIE RGB color matching functions. For all the following calculations, the ground truth spectra \mathbf{y} refer to the normalized ground truth spectra $\mathbf{y}_{\text{normalized}}$.

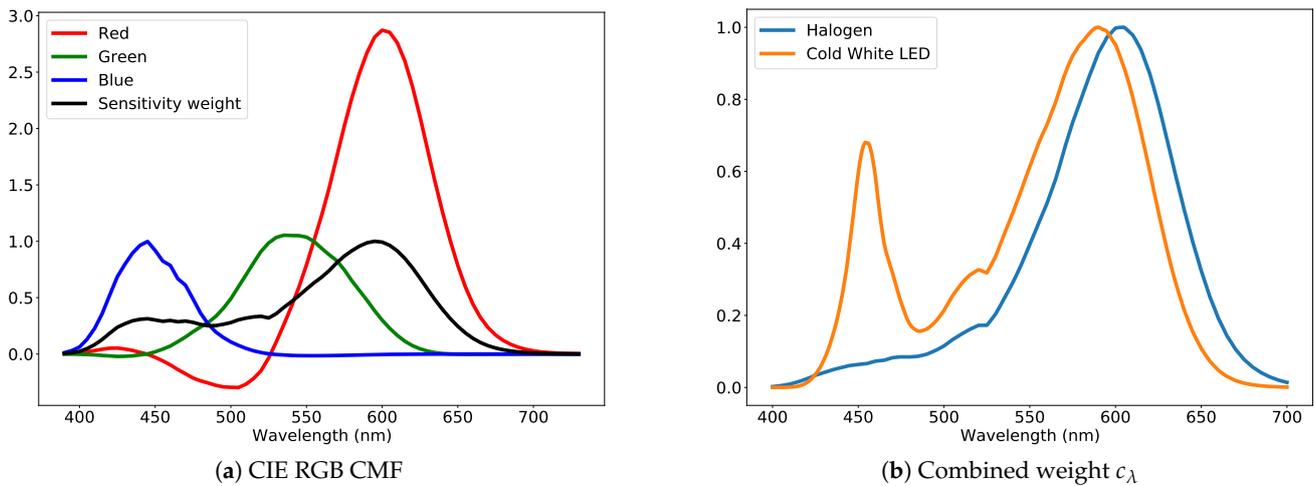


Figure 7. (a) CIE RGB color matching functions used as a proxy for camera sensor spectral sensitivity. The mean of the absolute values (in black) was used as a single proxy for sensor sensitivity. (b) Combined weight c_λ calculated from the sensitivity weight and illuminant spectrum.

The test dataset was collected at a later time, rather than simultaneously with the training set, in order to simulate realistic use cases. The cameras, lights, and color patches were reset with a reasonable attempt to place them in a similar setting to how the training data were collected. In practice, the test data collection setup still differed from the training data setup, for example with differences in illuminant angles and imaging distances from the color patches. Furthermore, the color patches were selected from a different manufacturer to those in the training set, and one of the illuminants, the cold white LED, had a distinctly different spectrum from the spectra of the illuminants in the training set. Again, 10 diffraction images for each combination of the 3 cameras, the 2 illuminants (whose spectra are shown in Figure 5b), and 36 color patches were collected. Cepstrum profile tensors and ground truth spectrum data were computed in a manner identical to the training data.

The reader may be interested in the fact that the spectra of the collected color patches were smooth. Each spectrum was sampled over 102 wavelengths over the visible wavelength range of 400–700 nm by the Specim IQ hyperspectral camera used to collect the ground truth data. The full-width at half-maximum (FWHM) bandwidth, a measure of the spectral resolution, of the Specim IQ hyperspectral camera was 7 nm [46]. Furthermore, the reflectance spectra of many materials, including paints, are smooth [34], leading to the discovery that most visible reflectance spectra can be encoded by between 13 and 23 principle component parameters. Parkkinen et al. [11] found that only 8 principle components are enough to explain 99.9% of the variance in reflectance spectra. Note that this only indicates the dimensionality of the data and does not directly provide information on its smoothness, since the components may be nonsmooth. To quantify the smoothness of the spectra, we calculated the mean of the sum of absolute first and second differences of the normalized spectra, i.e., the means of $\sum_{\lambda=401nm}^{\lambda=700nm} |\hat{s}_\lambda - \hat{s}_{\lambda-1}|$ and $\sum_{\lambda=402nm}^{\lambda=700nm} |\hat{s}_\lambda - 2\hat{s}_{\lambda-1} + \hat{s}_{\lambda-2}|$, where $\hat{s} = \frac{s}{\max(s)}$ and s are spectra, which for the test dataset were 1.84 and 0.149, respectively. The maximum possible sum of the absolute first-order differences for a single spectrum was 150 for a comb-like spectrum, so the spectra were indeed relatively smooth. While the ground truth radiance spectra were observed to be smooth, our computational methods did not exploit this property, other than in the form of regularization, for better generalizability of recovered spectra. This was in contrast to many previous methods that explicitly assumed smooth spectra [7,8].

5. Computational Pipeline

This section provides the details of the full computational pipeline used for reconstructing the spectrum with the algorithms described in Section 3 from images acquired

using the protocol described in Section 4. We first explain a common calibration scheme and then discuss additional elements required for applying the general algorithms for our imaging setup. Finally, we explain how the models were trained using the ground truth spectra and how the accuracy of the reconstructed spectra was evaluated.

5.1. Calibration

Calibration is necessary to express spectra as a function of wavelength λ . The images consisted of pixels and spectra produced by the deconvolution and cepstral method as a function of pixel dislocation d_p , which is the distance from the source of diffraction to the diffracted component in pixels. For a monochromatic point-like source, d_p is easily measured between the zeroth-order and first-order diffraction components. We assumed that there exists a linear mapping between pixel dislocations and wavelengths. The strategy we employed to solve this linear mapping was to capture diffraction images of an illuminant through a diffuser with a known spectrum and that had multiple peaks. For the illuminant, we used an RGB LED light producing light that was in appearance white, but consisted of red, green, and blue LED light sources. Capturing one diffraction image of the RGB LED is equivalent to imaging the red, green, and blue LED separately. Other good choices for the calibration illuminant would be light sources that have multiple distinct peaks, such as some metal halide and fluorescent lamps. The spectrum of the RGB LED illuminant was measured using a Hopocolor OHSP-350C spectral illuminance meter. From the spectrum, maxima were extracted for the red, green, and blue parts of the spectrum. As a result, we had (λ_i, p_i) pairs for $i \in \{red, green, blue\}$, and we solved for $\lambda = ap + c$, where a and c are parameters to be solved for each camera.

5.2. Deconvolution

To process the diffraction images for the deconvolution approach described in Section 3.1, we first needed to split each image into diffraction components. We did this using a semi-automatic method: First, we found the center image by finding the brightest point in a blurred version of the image and using a bounding area of predefined size and shape around it. Then, we obtained an estimate for the diffraction pattern using 2D deconvolution (4); this resulted in a noisy estimate similar to Figure 3. By finding the angles with the highest total brightness values in the estimated diffraction pattern, we determined the diffraction directions of the grating. By combining this with the previously calibrated pixel dislocation information, we found the clipping bounds for the diffraction components. After this, we chose one pair of diametrically opposite first-order diffraction components, rotated the image to make the diffraction horizontal, and used 1D deconvolution (6) in the rotated image to obtain an estimated 1D diffraction pattern p'_c for each color channel c . Finally, we mapped the pixel dislocations to wavelengths and averaged the two sides by setting $r'_c(\lambda) = (p'_c(x_\lambda) + p'_c(-x_\lambda))/2$, where x_λ is the pixel dislocation corresponding to wavelength λ .

Each resulting function $r'_c : [\lambda_{min}, \lambda_{max}] \rightarrow \mathbb{R}$ estimates the *raw spectrum* r_c , which is the light spectrum f multiplied pointwise by a response function that depends only on the spectral response s_c of the camera and the diffraction kernel h , and thus staying constant in all images taken with the same imaging setup. Thus, it makes sense to train wavelength-specific linear models to obtain the estimate f' for the spectrum f from the raw spectrum estimates r'_c . This means that for each imaging setup, we trained the weights $w_c(\lambda) \in \mathbb{R}$ for all color channels c and wavelengths $\lambda \in [\lambda_{min}, \lambda_{max}]$ that yielded the estimate:

$$f'(\lambda) = \sum_{c \in \{red, green, blue\}} w_c(\lambda) r'_c(\lambda).$$

For training the weights, we computed the raw spectrum estimates r'_c for each training image and optimized the weights to minimize the sum of squared differences between the estimates f' and the ground truth spectra f by solving a separate linear least squares

problem for each wavelength λ . We also trained an alternative model where we constrained the weights to be non-negative using constrained linear least squares optimization.

Both deconvolutions (2D and 1D) were carried out using discretized images and discrete Fourier transforms, resulting in discretized raw spectrum estimates r'_c and spectrum estimate f' . For the preliminary 2D deconvolution, we found that using the squared L2 norm as the regularization term, that is a one-hot regularization kernel k , worked sufficiently well for the purpose of detecting diffraction directions. For the final 1D deconvolution, we wanted to enforce continuity in the raw spectrum estimates, which is why we used the sum of squared differences of adjacent elements in r'_c as the regularization term. This means that the regularization kernel k was zero except for two adjacent elements R and $-R$, where R is a regularization weight hyperparameter, which controls the smoothness of the raw spectra obtained from the 1D deconvolution.

5.3. Cepstrum

As discussed in Section 3.2, to model the relationship between the cepstra and the spectra, we assumed a linear relationship. Two different models were tested: a simpler one that assumed a one-to-one mapping between elements and a more complex one where each element in the input cepstrum affected each output element. The first one is given simply by:

$$\hat{y} = w \circ x + b,$$

where \circ is an elementwise product and w and b are l -dimensional vectors. The other one is given by:

$$\hat{y} = Wx + b,$$

where W is an $l \times l$ matrix and b is a vector of size l , i.e., standard linear regression.

5.4. Training

To measure accuracy, we used the Canberra distance, $d_{\text{Can}}(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{\lambda} \frac{|\hat{y}_{\lambda} - \mathbf{y}_{\lambda}|}{\hat{y}_{\lambda} + \mathbf{y}_{\lambda}}$ between the spectrum extracted from diffraction images $\hat{\mathbf{y}}$ and the ground truth spectrum \mathbf{y} . The Canberra distance is chosen based on a comprehensive evaluation of spectral differences [47]. However, the Canberra distance does not factor in the spectral sensitivity of a camera's sensor: a sensor's ability to record a response for a particular wavelength range is proportional to the sensor's spectral sensitivity. That is, if a sensor has a low sensitivity to a particular wavelength range, commonly below 400 nm or above 700 nm, one cannot expect any kind of accurate result for that range. Such reasoning also applies to the illumination spectrum as well: if there is no light at a particular wavelength range, no camera sensor response can be expected. For the aforementioned reasons, we modified the Canberra distance with a wavelength-specific weighting factor c_{λ} and define a weighted Canberra distances as:

$$d_{\text{WCan}}(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{\lambda} c_{\lambda} \frac{|\hat{y}_{\lambda} - \mathbf{y}_{\lambda}|}{\hat{y}_{\lambda} + \mathbf{y}_{\lambda}}.$$

The weighting factor c_{λ} takes into consideration the camera spectral sensitivity, the sensitivity weight, and the spectrum distribution of the illuminant, the illuminant weight, used to illuminate the color patches. Instead of measuring each camera's spectral sensitivity, we used the mean of absolute CIE RGB color matching function [45] values, as show in Figure 7a. Absolute values were used, because the CIE color matching function for red also has negative values. The reason to use the CIE RGB color matching function values as a proxy for camera sensor sensitivity is that camera sensor spectral responses are designed to be close to these color matching functions. Indeed, a visual comparison of measured camera sensor spectral sensitivity in [48,49] to CIE RGB color matching functions reveals a close match. Any previous measurement for the camera sensor spectral sensitivity would not apply, due to the diffraction grating placed in the optical path. The final weighting factor c_{λ} is the product of the illuminant weight and the sensitivity weight normalized

to a maximum of one. For the test set illuminants, the weighting factors c_λ are shown in Figure 7b, which are the product of the sensitivity weight in Figure 7a and illuminant relative spectra shown in Figure 5b. To eliminate the effect of the total power of the weighting factor, i.e., the area under the weighting factor curve, the final weighted Canberra distance is divided by the sum of c_λ , so the measurement for accuracy that we used becomes:

$$d_{WCan}(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{\lambda} c_{\lambda} \frac{|\hat{\mathbf{y}}_{\lambda} - \mathbf{y}_{\lambda}|}{\hat{\mathbf{y}}_{\lambda} + \mathbf{y}_{\lambda}} \sum_{\lambda} c_{\lambda}, \tag{7}$$

For the deconvolution method, we needed to train two things for each camera: the weights $w_c(\lambda)$ for the linear transformation from the raw spectrum estimates r'_c to the spectrum estimate f' and the regularization weight R . In our case, the training data contained each ground truth spectrum f only in a normalized form, and thus, we also normalized our estimate spectrum f' as a postprocessing step. Due to the nonlinearity caused by the normalization, we needed to slightly alter the method described in Section 5.2 for training the weights $w_c(\lambda)$ for fixed R : we also trained a scaling coefficient for each estimate spectrum f' and alternated between minimizing the loss with respect to the weights $w_c(\lambda)$ and minimizing the same loss with respect to the scaling coefficients. Both of these minimization problems can be solved individually using (constrained) linear least squares, and we repeated these minimization steps until the result converged. After this, the scaling coefficients may be discarded as the spectrum estimate f' is normalized anyway.

We picked the value for the regularization weight hyperparameter R by optimizing the mean weighted Canberra distance in a 5-fold cross-validation. In our preliminary tests, the cross-validation score appeared to be approximately unimodal as a function of the weight R , and thus, we chose ternary search as the optimization algorithm.

In training for the cepstrum-based methods, the expression for the weighted Canberra distance (Equation (7)) contains absolute values and is therefore tricky to optimize. Instead, to model the relationship between the cepstra and the spectra, we optimized a simpler objective:

$$d_{WMSE}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{\sum_{\lambda} c_{\lambda} (\hat{\mathbf{y}} - \mathbf{y})^2}{\sum_{\lambda} c_{\lambda}} \tag{8}$$

which is the weighted mean-squared error. Both models presented in Section 5.3 were trained by minimizing Equation (8) using full-batch gradient descent with an adaptive learning rate. Note that the predicted spectra $\hat{\mathbf{y}}$ need to sum to one after weighting. This was accomplished by simply dividing them by their sum at each iteration of the gradient descent. We trained each camera separately, but mixed all light sources together per camera. This allowed the model to generalize to new light sources. The results for the two presented cepstrum-to-spectrum models are given in the next section.

5.5. Testing Procedure

To evaluate a generally applicable method, we needed a testing procedure that evaluates the accuracy in unseen conditions. As described in Section 4.3, the test dataset consisted of color patches from a different manufacturer than the color patches in the training set. The motivation to use another paint manufacturer for the test set was to minimize the chance that the reflectance spectra of the color patches would be identical to the color patches in the training set. Two illuminants were used in the test set: halogen and cold white LED. The halogen illuminant served to test the generalizability over unseen color patches, whereas the cold white LED served to test the generalizability over both illumination conditions and unseen color patches.

The testing procedure involved three agents: a tester and two modelers (one for the deconvolution method and one for the cepstrum method). The tester did not directly partake in the modeling work, but was responsible for calculating the accuracy results over the test dataset and keeping the ground truth values secret from the modelers. During

training, the modelers had access to the diffraction images, cepstrum profile tensors, and ground truth spectra for the training data, which were further split into the data used for training and validating the models. For testing, the modelers received only the diffraction images and the cepstrum profile tensors. They also knew which samples related to which camera and which illuminant, but the illuminants were identified only by Codes A and B and the illuminant spectra were not provided for them. After having selected the best models, the modelers applied them on the test images and submitted the predicted spectra for the tester. Only one additional submission was necessary for the cepstrum method due to a scaling issue, but even in this instance, the tester only revealed information for correcting the scaling.

The main goal of this strict testing protocol was to prevent even accidental fine-tuning of the models for the test conditions. The results reported in Tables 1 and 2 were based on the submitted inferred spectra, and the models were not modified or retrained in any way after seeing the final results.

Table 1. Mean of weighted Canberra distances between estimated and ground truth spectra for camera, illuminant, method, and model combinations. Smaller numbers mean better accuracy w.r.t. ground truth spectra. The best result is indicated in boldface.

Mean of Weighted Canberra Distances							
Illuminant		Halogen			Cold White LED		
	Model	Canon M3	RPI HQ	Samsung S10	Canon M3	RPI HQ	Samsung S10
Deconvolution	Free	0.0603	0.0686	0.0359	0.1316	0.1134	0.0586
	Non-negative	0.0552	0.0697	0.0356	0.1274	0.1227	0.0592
Cepstrum	Fully connected	0.1089	0.1271	0.1364	0.2663	0.3080	0.2045
	Simple	0.1041	0.1240	0.1272	0.2392	0.2637	0.1995

Table 2. Summary results of the means of weighted Canberra distances between estimated and ground truth spectra for illuminant, method, and model combinations averaged over cameras. The best result is indicated in boldface.

Summary Results: Mean of Weighted Canberra Distances				
Illuminant		Halogen	Cold White LED	
	Model			Mean
Deconvolution	Free	0.0549	0.1010	0.0779
	Non-negative	0.0535	0.1029	0.0781
Cepstrum	Fully connected	0.1241	0.2591	0.1913
	Simple	0.1184	0.2339	0.1759

6. Results

The results for the weighted Canberra distances for the test set are reported in Table 1, in which the results are tabulated as illuminant, camera, method type, and model. A summary of the results is reported in Table 2, where the metrics are averaged over all cameras to provide an overall perspective. Finally, a random selection of the spectra estimated using the non-negative deconvolution model is shown in Figure 8 for a visual comparison of the estimates in comparison to the ground truth spectra. The results clearly indicated that the deconvolution method performed significantly better than the cepstrum methods. All models performed significantly worse for the cold white LED illuminant, which was not present in the training data, indicating that generalization to illuminants not present in

the training set was much weaker. The differences between the camera models were fairly small. For better-performing deconvolution models, the smartphone camera, Samsung S10, performed the best, with a mean weighted Canberra distance of 0.0356 for the non-negative model under halogen illumination, although performance differences between the cameras were negligible. For comparison, the mean weighted Canberra distance was approximately 0.056 between the ground truth spectra and ground truth spectra added with zero mean normally distributed noise with a standard deviation of 10% of the mean of each spectrum.

The plots in Figure 9 show the error dependence of the test set spectra with respect to the wavelength. The logarithmic mean error is calculated for each wavelength λ by:

$$\log \left(\frac{\sum \frac{|\mathbf{y}(\lambda) - \hat{\mathbf{y}}(\lambda)|}{\mathbf{y}(\lambda)}}{n} \right)$$

where $\mathbf{y}(\lambda)$ is the ground truth spectrum at λ , $\hat{\mathbf{y}}(\lambda)$ the estimated spectrum at λ , and n the number of test spectra, i.e., $n = 36$. The maximum is similarly calculated for each wavelength λ by:

$$\log \left(\max_{(\mathbf{y}, \hat{\mathbf{y}}) \in \text{Test set}} \left\{ \frac{|\mathbf{y}(\lambda) - \hat{\mathbf{y}}(\lambda)|}{\mathbf{y}(\lambda)} \right\} \right)$$

For the cepstrum models, the errors were more concentrated than for the deconvolution models, but at a higher level. For the deconvolution models, the mean absolute error was relatively stable across the entire spectrum, with slightly better performance in the 400–450 nm range. The error of the cepstrum models, although on average higher, was relatively more stable over the spectrum than the error of deconvolution models. Within the range of 550–600 nm, in which most cameras had high spectral sensitivity, the cepstrum models performed slightly better. On average, the 0.95 quantile error was approximately 6.25-times higher for the deconvolution models and approximately 4.8-times higher for the cepstrum models than the mean error, indicating a relatively large worst-case deviation from mean performance for all models and methods.

To complement the direct evaluation of spectral reconstruction, we also report accuracy in terms of color specification. Table 3 shows the median color difference values ΔE_{00} in the CIEL*a*b* space [50] for the different model, camera, and illuminant combinations, and Table 4 summarizes these by each model and illuminant. The metric was designed so that a value of one indicates a color difference that is just perceivable for a typical human observer. We report the medians over the test spectra as a more reliable overall estimate, to avoid giving excess weight for a small number of outlier samples. For ΔE_{00} , the CIEL*a*b* values were calculated from the CIE XYZ tristimulus values, which were calculated from the spectra using the CIE 1931 2° standard observer color matching functions without an illuminant reference because the spectra already included the effect of the illuminant. The ΔE_{00} values indicated that the cepstrum method was poor at estimating the CIEL*a*b* values, while the deconvolution methods performed significantly better, especially under halogen illumination.

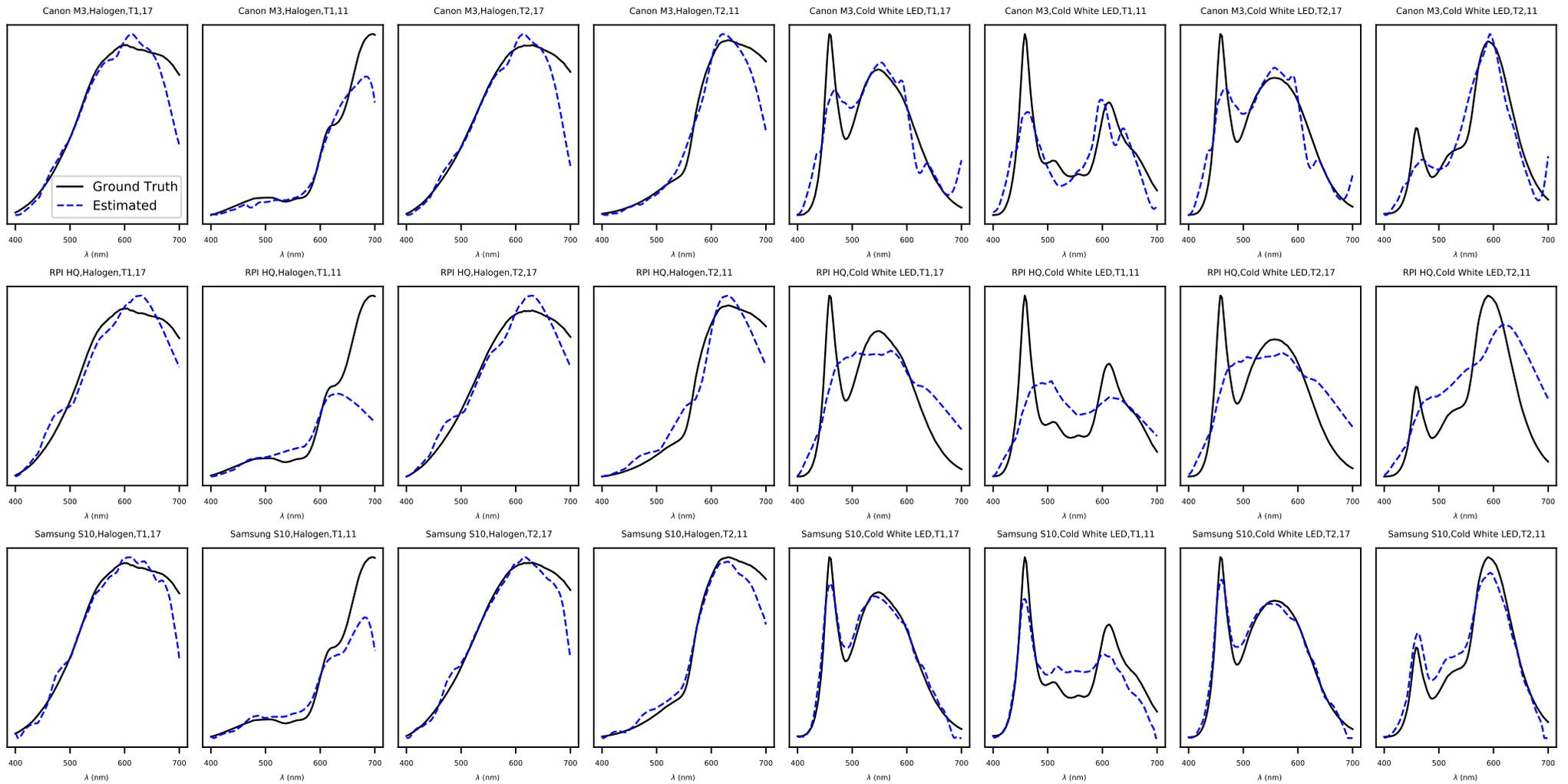


Figure 8. A random selection of relative spectra estimated using the best presented method, the non-negative deconvolution model (blue dashed line) compared against the relative ground truth spectra (black solid line). Each row shows spectra for a single camera with the first half of columns imaged under halogen illumination and the second half under cold white LED illumination.

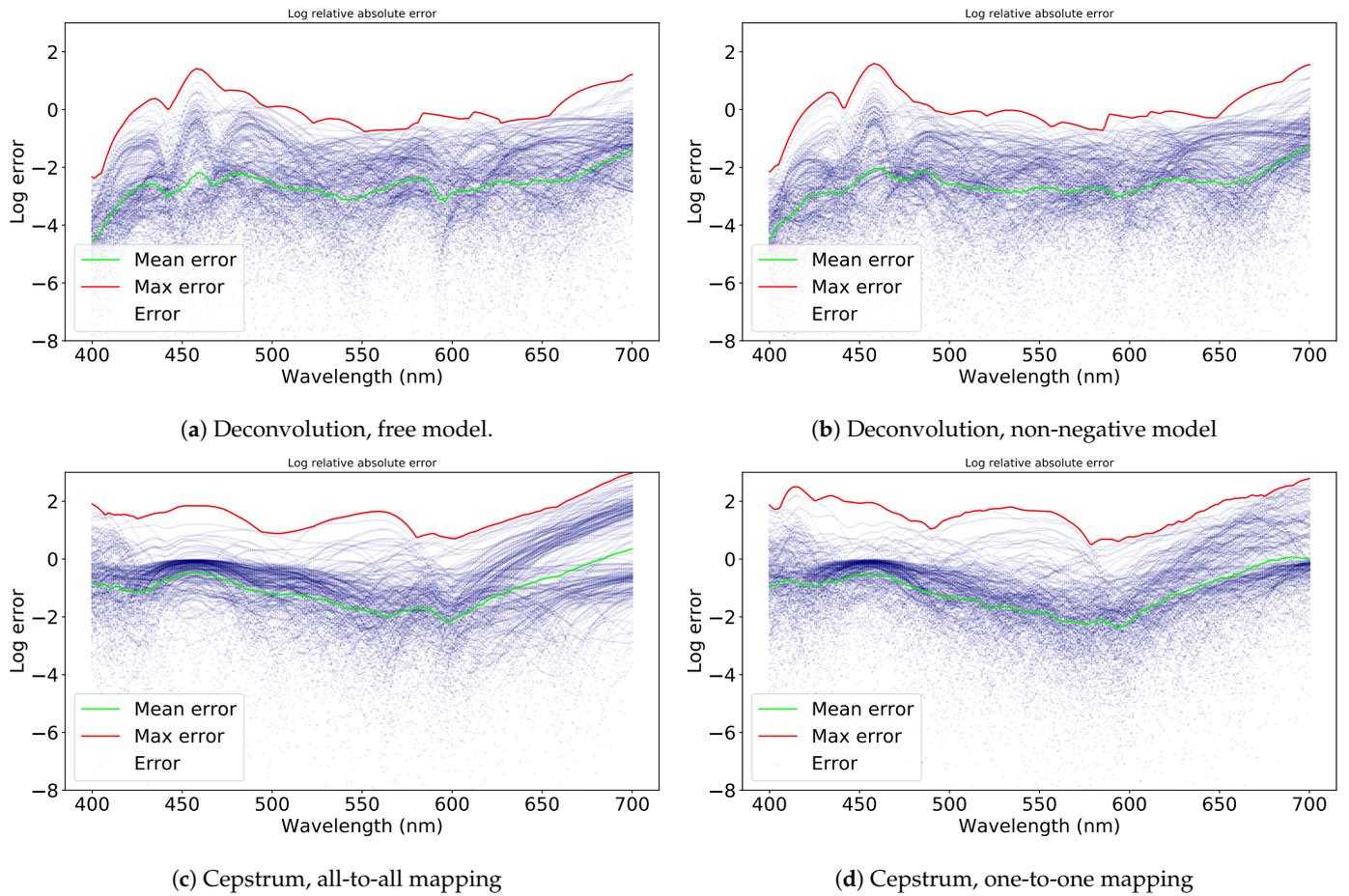


Figure 9. Log relative absolute error of the test set spectra in relation to ground truth spectra. While the mean error for each wavelength is generally low, especially for the deconvolution models, the maximum error can be quite high.

Table 3. Median of ΔE_{00} between estimated and ground truth spectra for camera, illuminant, method, and model combinations. Smaller numbers mean better accuracy w.r.t. ground truth spectra. The best result is indicated in boldface.

		Median of ΔE_{00}					
Illuminant		Halogen			Cold White LED		
	Model	Canon M3	RPI HQ	Samsung S10	Canon M3	RPI HQ	Samsung S10
Deconvolution	Free	2.34	3.65	1.70	5.46	4.24	3.55
	Non-negative	2.40	6.94	1.63	6.20	10.99	3.38
Cepstrum	Fully connected	12.79	8.57	14.40	28.28	29.65	23.60
	Simple	9.94	11.59	13.05	23.80	26.03	21.97

Table 4. Summary results of the median of ΔE_{00} between estimated and ground truth spectra for illuminant, method, and model combinations, median over cameras.

Summary Results: Median of ΔE_{00}				
Illuminant		Halogen	Cold White LED	
	Model			Median
Deconvolution	Free	2.44	4.37	3.25
	Non-negative	2.72	6.85	5.08
Cepstrum	Fully connected	12.56	28.56	19.61
	Simple	11.37	23.81	17.49

7. Discussion and Conclusions

The results indicated that we can extract the spectra of color patches in different illumination conditions using diffraction imaging and algorithmic reconstruction of the spectrum. The estimates were not perfect, and the accuracy depended on the lighting and camera model; however, the important observation was that the general-purpose approach only requiring an optically simple add-on device attached to a regular camera sensor worked for a wide range of conditions with satisfactory accuracy. The method based on the deconvolution operation resulted in more accurate spectrum extraction, compared to the method based on the cepstrum operation, but the cepstrum method may still have value for calibration purposes. Whilst the weighted Canberra distance results for the cepstrum models were somewhat poor, the cepstrum operation itself was merited with its simplicity and partial robustness, as discussed in Section 3.2. Indeed, the original idea for using the cepstrum came from the work in [17] and from diffraction imaging experiments conducted by the authors of metal halide and fluorescent illuminants, characterized by strong spectral peaks, which were immediately evident in the extracted cepstrum profiles. Notable in these experiments was the good agreement of peak location and relative intensity of the cepstrum profile with spectra measured using a calibrated spectrophotometer, after having accounted for the camera's spectrum response.

The test set data were collected independently of the training data and consisted of paint samples from a different manufacturer, and we used also an illuminant that was not used in the training data. In other words, the testing procedure detailed in Section 5.5 was specifically designed to measure the out-of-domain generalizability of the methods. The results for color patches under halogen illumination indicated how well the models generalized to unseen color patches under an illuminant included in the training set, yet whose spectrum was unknown. The results for color patches under the cold white LED illumination indicated how well the models generalized to unseen color patches under an illuminant not included in the training set, thus representing the more challenging case for out-of-domain generalizability (an even more challenging case would be to adapt a trained model to a new camera from a small number of training samples and test generalizability to unseen color patches and illuminants). For both of these test cases, the deconvolution models provided good results, especially when considering that the ground truth spectra are unlikely to be perfect. Due to the good performance of the deconvolution models for illuminants also found in the training set, we suspect that a pretrained model could be easily adapted for better accuracy for new illuminants with a relatively small number of samples.

When inspecting the results from the perspective of determining the color, the smallest error was obtained with the non-negative deconvolution model using the Samsung S10 camera under halogen illumination, reaching $\Delta E_{00} = 1.63$. This almost matched the just-noticeable difference of $\Delta E_{00} = 1.0$ and is hence likely to be accurate enough for typical applications in need of low-cost imaging devices, but for, e.g., scientific imaging, the results may not be accurate enough. The performance of the other cameras using the deconvolution models, as seen in Table 3, was also fairly good under halogen illumination.

A closer inspection of the estimated spectra (Figure 8) and errors (Figure 9) revealed that the errors were larger towards the 700nm end of the spectrum. This was likely due to the vanishing spectral sensitivity of the cameras toward the ends of the visible spectrum. One potential strategy for improving the accuracy would be to remove the infrared filter present in all cameras used in our experiments to increase the spectral sensitivity for larger wavelengths, but investigating the practical value of this would require replicating all of the data collection and training of the models.

The mathematical model used here assumed noise-free images and could be extended to explicitly model electronic and shot noise produced by the imaging device, possibly in a similar fashion as [9]. The computational models themselves introduced some form of noise resilience due to the redundancy of the large spatial area of the diffraction pattern, so that individual noisy pixels did not directly influence the result, but explicitly modeling the noise could still improve the accuracy especially for the cepstrum method.

Better modeling of the diffraction phenomenon could also improve the accuracy, for instance by relaxing the linearity and isoplane requirements imposed on the diffraction kernel h_{λ} , combined with a more flexible deconvolution model. Furthermore, better modeling of spatial variability in the illumination intensity in combination with the imaging aperture size, exposure time, and signal amplification could help. A potential improvement could also be found in imposing stronger priors on the spectra, similar to Cheung et al. [7], Parkkinen et al. [11], although the lack of explicit requirements can also be considered an advantage of the proposed method. The priors could also include information on the space of trichromatic camera spectral sensitivity functions [31]. Furthermore, improvements can be made by illuminant estimation [51,52] from the diffraction images, which would allow for direct reflectance spectra estimation without illuminant spectra measurement.

Author Contributions: Conceptualization, M.E.T., T.T., and C.R.; methodology, M.E.T.; software, M.E.T., T.T., and C.R.; validation, M.E.T., T.T., and C.R.; formal analysis, M.E.T., T.T., and C.R.; investigation, M.E.T.; resources, M.E.T., T.T., and C.R.; data curation, M.E.T.; writing—original draft preparation, M.E.T., T.T., C.R., and A.K.; writing—review and editing, M.E.T., T.T., C.R., and A.K.; visualization, M.E.T.; supervision, A.K.; project administration, A.K.; funding acquisition, A.K., M.E.T., and C.R. All authors read and agreed to the published version of the manuscript.

Funding: This research was funded by Business Finland Grant 1389/31/2019. Open access funding provided by the University of Helsinki.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; nor in the decision to publish the results.

References

1. Ohno, Y. Spectral color measurement. *Colorimetry: Understanding the CIE System*; In CIE Central Bureau: Vienna, Austria, 2007; pp. 101–134
2. Vane, G.; Goetz, A.F.; Wellman, J.B. Airborne imaging spectrometer: A new tool for remote sensing. *IEEE Trans. Geosci. Remote Sens.* **1984**, *6*, 546–549. [[CrossRef](#)]
3. Goetz, A.F.; Herring, M. The high resolution imaging spectrometer (HIRIS) for EOS. *IEEE Trans. Geosci. Remote Sens.* **1989**, *27*, 136–144. [[CrossRef](#)]
4. Herrala, E.; Okkonen, J.T.; Hyvarinen, T.S.; Aikio, M.; Lammasniemi, J. Imaging spectrometer for process industry applications. *Optical Measurements and Sensors for the Process Industries. Int. Soc. Opt. Photon.* **1994**, *2248*, 33–40.
5. Oke, J.; Cohen, J.; Carr, M.; Cromer, J.; Dingizian, A.; Harris, F.; Labrecque, S.; Lucinio, R.; Schaal, W.; Epps, H.; et al. The Keck low-resolution imaging spectrometer. *Publ. Astron. Soc. Pac.* **1995**, *107*, 375. [[CrossRef](#)]
6. Pearlman, J.S.; Barry, P.S.; Segal, C.C.; Shepanski, J.; Beiso, D.; Carman, S.L. Hyperion, a space-based imaging spectrometer. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 1160–1173. [[CrossRef](#)]
7. Cheung, V.; Westland, S.; Li, C.; Hardeberg, J.; Connah, D. Characterization of trichromatic color cameras by using a new multispectral imaging technique. *JOSA A* **2005**, *22*, 1231–1240. [[CrossRef](#)]
8. Heikkinen, V.; Lenz, R.; Jetsu, T.; Parkkinen, J.; Hauta-Kasari, M.; Jääskeläinen, T. Evaluation and unification of some methods for estimating reflectance spectra from RGB images. *JOSA A* **2008**, *25*, 2444–2458. [[CrossRef](#)] [[PubMed](#)]

9. Shimano, N.; Hironaga, M. Recovery of spectral reflectances of imaged objects by the use of features of spectral reflectances. *JOSA A* **2010**, *27*, 251–258. [[CrossRef](#)]
10. Peyvandi, S.; Amirshahi, S.H.; Hernandez-Andres, J.; Nieves, J.L.; Romero, J. Generalized inverse-approach model for spectral-signal recovery. *IEEE Trans. Image Process.* **2012**, *22*, 501–510. [[CrossRef](#)]
11. Parkkinen, J.P.; Hallikainen, J.; Jaaskelainen, T. Characteristic spectra of Munsell colors. *JOSA A* **1989**, *6*, 318–322. [[CrossRef](#)]
12. Imai, F.H.; Berns, R.S.; Tzeng, D.Y. A comparative analysis of spectral reflectance estimated in various spaces using a trichromatic camera system. *J. Imaging Sci. Technol.* **2000**, *44*, 280–287.
13. Attewell, D.; Baddeley, R.J. The distribution of reflectances within the visual environment. *Vis. Res.* **2007**, *47*, 548–554. [[CrossRef](#)]
14. Imai, F.H.; Berns, R.S. Spectral estimation using trichromatic digital cameras. In Proceedings of the International Symposium on Multispectral Imaging and Color Reproduction for Digital Archives, Chiba University, Chiba, Japan, 21–22 October 1999; Volume 42, pp. 1–8
15. Foster, D.H.; Amano, K. Hyperspectral imaging in color vision research: Tutorial. *J. Opt. Soc. Am. A* **2019**, *36*, 606–627. [[CrossRef](#)]
16. Eckhard, J.; Eckhard, T.; Valero, E.M.; Nieves, J.L.; Contreras, E.G. Outdoor scene reflectance measurements using a Bragg-grating-based hyperspectral imager. *Appl. Opt.* **2015**, *54*, D15–D24. [[CrossRef](#)]
17. Toivonen, M.E.; Rajani, C.; Klami, A. Snapshot hyperspectral imaging using wide dilation networks. *Mach. Vis. Appl.* **2021**, *32*, 1–11. [[CrossRef](#)]
18. Okamoto, T.; Takahashi, A.; Yamaguchi, I. Simultaneous acquisition of spectral and spatial intensity distribution. *Appl. Spectrosc.* **1993**, *47*, 1198–1202. [[CrossRef](#)]
19. Johnson, W.R.; Wilson, D.W.; Fink, W.; Humayun, M.S.; Bearman, G.H. Snapshot hyperspectral imaging in ophthalmology. *J. Biomed. Opt.* **2007**, *12*, 014036. [[CrossRef](#)] [[PubMed](#)]
20. Habel, R.; Kudenov, M.; Wimmer, M. Practical spectral photography. In *Computer Graphics Forum*; Blackwell Publishing Ltd: Oxford, UK, 2012; Volume 31, pp. 449–458
21. Toivonen, M.E.; Klami, A. Practical Camera Sensor Spectral Response and Uncertainty Estimation. *J. Imaging* **2020**, *6*, 79. [[CrossRef](#)] [[PubMed](#)]
22. Markvart, A.; Liokumovich, L.; Medvedev, I.; Ushakov, N. Continuous Hue-Based Self-Calibration of a Smartphone Spectrometer Applied to Optical Fiber Fabry-Perot Sensor Interrogation. *Sensors* **2020**, *20*, 6304. [[CrossRef](#)]
23. Blanchard, P.M.; Greenaway, A.H. Simultaneous multiplane imaging with a distorted diffraction grating. *Appl. Opt.* **1999**, *38*, 6692–6699. [[CrossRef](#)] [[PubMed](#)]
24. Pan, B.; Wang, Q. Single-camera microscopic stereo digital image correlation using a diffraction grating. *Opt. Express* **2013**, *21*, 25056–25068. [[CrossRef](#)] [[PubMed](#)]
25. Pan, B.; Yu, L.; Zhang, Q. Review of single-camera stereo-digital image correlation techniques for full-field 3D shape and deformation measurement. *Sci. China Technol. Sci.* **2018**, *61*, 2–20. [[CrossRef](#)]
26. Javoršek, D.; Jerman, T.; Rat, B.; Hladnik, A. Assessing the performance of a spectral reflectance estimation method based on a diffraction grating and a spectroradiometer. *Color. Technol.* **2014**, *130*, 288–295. [[CrossRef](#)]
27. Rom, R. On the cepstrum of two-dimensional functions (Corresp.). *IEEE Trans. Inf. Theory* **1975**, *21*, 214–217. [[CrossRef](#)]
28. Okamoto, T.; Yamaguchi, I. Simultaneous acquisition of spectral image information. *Opt. Lett.* **1991**, *16*, 1277–1279. [[CrossRef](#)]
29. Descour, M.; Dereniak, E. Computed-tomography imaging spectrometer: Experimental calibration and reconstruction results. *Appl. Opt.* **1995**, *34*, 4817–4826. [[CrossRef](#)] [[PubMed](#)]
30. Aikio, M. Hyperspectral prism-grating-prism imaging spectrograph. Ph.D. Thesis, VTT Technical Research Centre of Finland, Espoo, Finland, 16 June 2001
31. Jiang, J.; Liu, D.; Gu, J.; Süsstrunk, S. What is the space of spectral sensitivity functions for digital color cameras? In Proceedings of the 2013 IEEE Workshop on Applications of Computer Vision (WACV), Clearwater Beach, FL, USA, 15–17 January 2013; pp. 168–179.
32. Vrhel, M.J.; Gershon, R.; Iwan, L.S. Measurement and analysis of object reflectance spectra. *Color Res. Appl.* **1994**, *19*, 4–9. [[CrossRef](#)]
33. Durmus, D. CIELAB color space boundaries under theoretical spectra and 99 test color samples. *Color Res. Appl.* **2020**, *45*, 796–802. [[CrossRef](#)]
34. Hardeberg, J.Y. On the spectral dimensionality of object colors. In Proceedings of the Conference on Colour in Graphics, Imaging, and Vision. Society for Imaging Science and Technology, Poitiers, France, 2–5 April 2002; Volume 2002, pp. 480–485
35. Noll, A.M. Cepstrum pitch determination. *J. Acoust. Soc. Am.* **1967**, *41*, 293–309. [[CrossRef](#)]
36. Kemerait, R.; Childers, D. Signal detection and extraction by cepstrum techniques. *IEEE Trans. Inf. Theory* **1972**, *18*, 745–759. [[CrossRef](#)]
37. Lee, D.J.; Krile, T.F.; Mitra, S. Power cepstrum and spectrum techniques applied to image registration. *Appl. Opt.* **1988**, *27*, 1099–1106. [[CrossRef](#)]
38. Gonzalez, R. Robust image registration via cepstral analysis. In Proceedings of the 2011 International Conference on Digital Image Computing: Techniques and Applications, Noosa, QLD, Australia, 6–8 December 2011; pp. 45–50.
39. Hassab, J.; Boucher, R. Analysis of signal extraction, echo detection and removal by complex cepstrum in presence of distortion and noise. *J. Sound Vib.* **1975**, *40*, 321–335. [[CrossRef](#)]

40. Mandava, M. Fast and Robust Cepstral Techniques for Digital Image Registration. Ph.D. Thesis, Texas Tech University, Lubbock, TX, USA, 1992.
41. Prasad, K.; Unbehauen, R. Analytical expressions of 2-D complex cepstrum. *Electron. Lett.* **1983**, *19*, 831–832. [[CrossRef](#)]
42. Van der Walt, S.; Schönberger, J.L.; Nunez-Iglesias, J.; Boulogne, F.; Warner, J.D.; Yager, N.; Gouillart, E.; Yu, T. Scikit-image: Image processing in Python. *PeerJ* **2014**, *2*, e453. [[CrossRef](#)] [[PubMed](#)]
43. Nogueira, F. Bayesian Optimization: Open Source Constrained Global Optimization Tool for Python. 2014. Available online: <https://github.com/fmfn/BayesianOptimization> (accessed on 15 January 2021)
44. Figuière, H.; Niedermann, H.U. Gphoto. 2021. Available online: <https://github.com/gphoto> (accessed on 11 January 2021)
45. Smith, T.; Guild, J. The CIE colorimetric standards and their use. *Trans. Opt. Soc.* **1931**, *33*, 73. [[CrossRef](#)]
46. Behmann, J.; Acebron, K.; Emin, D.; Bennertz, S.; Matsubara, S.; Thomas, S.; Bohnenkamp, D.; Kuska, M.T.; Jussila, J.; Salo, H.; others. Specim IQ: Evaluation of a new, miniaturized handheld hyperspectral camera and its application for plant phenotyping and disease detection. *Sensors* **2018**, *18*, 441. [[CrossRef](#)] [[PubMed](#)]
47. Deborah, H.; Richard, N.; Hardeberg, J.Y. A comprehensive evaluation of spectral distance functions and metrics for hyperspectral image processing. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* **2015**, *8*, 3224–3234. [[CrossRef](#)]
48. Camera Spectral Sensitivity Database. Available online: <https://www.gujinwei.org/research/camspec/db.html> (accessed on 8 April 2020).
49. Spectral Sensitivity Database. Available online: <https://nae-lab.org/~rei/research/cs/zhao/database.html> (accessed on 8 April 2020).
50. Sharma, G.; Wu, W.; Dalal, E.N. The CIEDE2000 Color-Difference Formula: Implementation Notes, Supplementary Test Data, and Mathematical Observations. In *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*. Available online: www.interscience.wiley.com (accessed on 28 August 2021).
51. Shi, W.; Loy, C.C.; Tang, X. Deep specialized network for illuminant estimation. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 11–14 October 2016; pp. 371–387.
52. Bianco, S.; Cusano, C.; Schettini, R. Single and multiple illuminant estimation using convolutional neural networks. *IEEE Trans. Image Process.* **2017**, *26*, 4347–4362. [[CrossRef](#)]