Single Image Spectral Reconstruction for Multimedia Applications

Antonio Robles-Kelly

NICTA, Locked Bag 8001, Canberra ACT 2601, Australia

antonio.robles-kelly@nicta.com.au

ABSTRACT

In this paper, we present a method which can perform spectral reconstruction and illuminant recovery from a single colour image making use of an unlabelled training set of hyperspectral images. Our method employs colour and appearance information to drive the reconstruction process subject to the material properties of the objects in the scene. The idea is to reconstruct the image spectral irradiance making use of a set of prototypes extracted from the training set. These spectra, together with a set of convolutional features are hence obtained using sparse coding so as to reconstruct the image irradiance. With the reconstructed spectra in hand, we proceed to compute the illuminant power spectrum using a quadratic optimisation approach. We provide a quantitative analysis for our method and compare to a number of alternatives. We also show sample results on illuminant substitution and transfer, film simulation and image recolouring using mood board colour schemes.

Categories and Subject Descriptors

I.4 [**Computing Methodologies**]: Image Processing and Computer Vision; I.5 [**Computing Methodologies**]: Pattern Recognition

General Terms

Algorithms, Experimentation, Performance

Keywords

Spectral reconstruction, Sparse coding, Image reproduction and enhacement

1. INTRODUCTION

In multimedia, computer vision, video and graphics, we rely upon cameras and rendering contexts to capture and reproduce colour image data. Furthermore, the accurate reproduction and capture of the scene colour across different devices is an important and active area of research spanning camera simulation [19], sensor design [5] and white balancing [14].

The manner in which colour data is presented to the user is central to multimedia applications since the imagery is often delivered to the user based upon a rendering intent, *i.e.* colorimetric, perceptual, etc., which can determine the processing to be undertaken or the display medium. To this end, several methods have been proposed whereby making use of reflectance and illuminant measurements, colorimetric simulations and comparisons can be effected using quadratic programming [6], monochromators [31], spectrophotometers [21] or fluorescence [10]. Other camera radiometric calibration methods such as that in [14] require the scene irradiance to be in hand or make assumptions regarding the scene.

Note that, even when the camera has been radiometrically calibrated, the image raw colour values are sensor specific [24]. Moroever, raw-to-raw colour mappings between cameras are generally limited to linear transformations [24]. The problem here stems from the fact that, in practice, cameras often do not abide to the Luther condition, *i.e.* the camera spectral sensitivity functions are a linear transformation of the CIE colour matching functions [22]. This induces a non-linear transformation between camera colour spaces which depends on both, the spectral sensitivity functions and the image irradiance [12].

Thus, and despite being a challenging task, the spectral reconstruction of the spectral irradiance from a single colour image opens up the possibility of exploiting the relationship between the scene spectral reflectance, illuminant power spectrum and sensor sensitivity functions for applications such as material-specific colour rendition [8], accurate colour reproduction [28] and material reflectance substitution [9].

It is important to note that existing spectral reconstruction approaches are eminently pixel-based, disregarding the object material properties and reconstructing the image irradiance solely based upon colour information. For instance, Nguyen et al. [25] have attempted to reconstruct the scene reflectance by mapping the RGB values yielded by a trichromatic camera to the scene reflectance spectra. Maloney [20] and Agahian *et al.* [2] have used a PCA basis to model of the surface reflectance. Romero *et al.* [26] also used a linear basis to represent the illuminant in the scene. In [1], the authors use look-up tables to reconstruct reflectance data using linear interpolation. Zhao and Berns [36] have used a Karhunen-Loeve transformation, *i.e.* the matrix **R** method, to perform spectral reflectance reconstruction using a calibration target.

2. CONTRIBUTION

Here, we present a method for spectral reconstruction that allows images to be reproduced and presented with a look and feel in better accordance to the rendering intent. For instance, a warm or cooler feeling can be conveyed by substituting the illuminant (Figures 2 and 3) or more complex emotions through mood boards (Figure 5). Moreover, our method is also suitable for emulation of other cameras and films (Figure 4), which can be used to give a sense of realism, authenticity or artistic choice in multimedia settings.

We note that, by using the material properties of the objects in the image, the effects of metamers (surfaces with different spectral reflectance but identical colour) can be reduced and, hence, a better spectral reconstruction can be obtained. This hinges in the notion that the image reflectance is a photometric invariant which is intrinsic to the object material. Thus, here we employ features often used for classification and recognition in computer vision in combination with a set of training images to reconstruct the spectral irradiance. This contrasts with other methods for spectral reconstruction in the sense that they do not attempt to model the object material properties but rather reconstruct the reflectance from colour information alone.

The method presented here is quite general, making the following assumptions

- The spectral sensitivity function of the camera used to acquire the colour image is known.

- The colour image can be white balanced using existing methods.

- The scene in the colour image is illuminated by a single illuminant.

- The illuminant power spectra for the training data set is available.

We would like to stress these are not overly restrictive, being akin to those in [25].

In Figure 1, we show a diagram of the training and testing phases of our method. At training, the dataset of hyperspectral images is used to obtain spectral and convolutional features. We then use these features to recover the prototype set using a constrained sparse coding approach. Since these prototypes correspond to spectral samples in the training dataset and their corresponding material properties, they can be employed at testing to reconstruct the image spectra. Finally, once the spectral reconstruction is effected, we compute the illuminant power spectrum making use of a quadratic optimisation approach.

The paper is organised as follows. In Section 3, we provide background material on the relationship between the image irradiance and colour. In Section 4, we present the constrained sparse coding approach used at training and elaborate upon the reconstruction of the image irradiance. We also introduce our illuminant power spectrum recovery step. In Section 5, we provide implementation details and further discussion on our method. We show experiments in Section 6 and conclude in Section 7.

3. IMAGE IRRADIANCE AND COLOUR

Throughout the paper, we employ the relationships between spectral irradiance, illuminant power spectrum, spectral sensitivity functions, colour and chromaticity. Thus, we
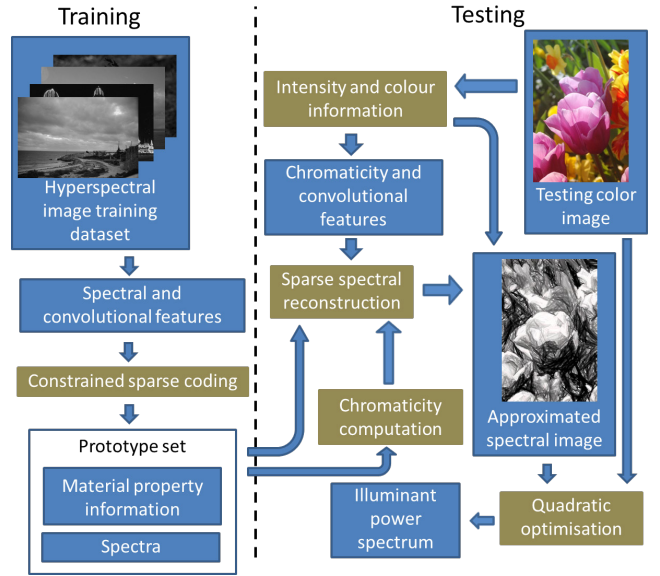
Figure 1: Our method takes at input a single colour image and recovers the spectra and illuminant making use of a training dataset.

commence by providing some useful background on these and express the image irradiance at wavelength λ and pixel u as follows

$$I(u, \lambda) = L(\lambda)R(u, \lambda) \qquad (1)$$

where $L(\lambda)$ is the power spectrum of the light impinging on the object surface and $R(u, \cdot)$ is the product of the mean scattered power in the direction of the viewer and the surface reflectance spectrum.

Equation 1 is important since it permits expressing the value of the colour channel $c = \{R, G, B\}$ at pixel u as

$$I_c(u) = \int_\Lambda L(\lambda)Q_c(\lambda)R(u, \lambda)d\lambda \qquad (2)$$

where $Q_c(\lambda)$ is the spectral sensitivity function of the imager for the colour channel c and Λ is the wavelength interval spanning the visible spectrum. Note that, in practice, spectra is sampled in a set of n discrete wavelength indexed values λ_k. As a result, the integral above becomes a sum given by

$$I_c(u) = g(u) \sum_{\lambda_k \in \Lambda} L(\lambda_k)Q_c(\lambda_k)r(u, \lambda_k) \qquad (3)$$

where $g(u) = |R(u, \cdot)|_2$ is the L-2 norm of $R(u, \cdot)$ across the wavelength domain, *i.e.* the radiosity at pixel u[1], such that $R(u, \lambda) = g(u)r(u, \lambda_k)$.

With the notation above, it becomes straightforward to define the colour pixel values in the rg-chromaticity space

---

[1] Note that radiosity is often called intensity in some areas of physics. This is also in good accordance with the intuitive use of the term in computer vision and image processing and, hence, we use it throughout the paper

as follows

$$K_c(u) = \frac{I_c(u)}{\sum_{q=\{R,G,B\}} I_q(u)} \qquad (4)$$

$$= \frac{\sum_{\lambda_k \in \Lambda} L(\lambda_k) Q_c(\lambda_k) r(u, \lambda_k)}{\sum_{q=\{R,G,B\}} \sum_{\lambda_k \in \Lambda} L(\lambda_k) Q_q(\lambda_k) r(u, \lambda_k)}$$

where $c = \{R, G, B\}$.

4. SPECTRAL RECONSTRUCTION

Note that, if a set $\mathcal{I}$ of training images is in hand and both, the spectral sensitivity functions and illuminant power spectra are known, for each pixel u, it becomes straightforward to compute both, the intensity $g(u)$ and its corresponding rg-chromaticity values. This is important since intensity values are traditionally used in computer vision to obtain textons [17]. We follow the intuition that these textons are indicative of local object surface properties [33] and, hence, apply the filter banks in [33] to recover a convolutional feature for each pixel u. Our choice of a convolutional feature vector with local support follows the notion that, in this manner, object specific information is ignored so as to capture material properties devoid of the object itself [27].

4.1 Spectra and Convolutional Features

The use of convolutional features allows for a treatment where the image irradiance can be used to recover a set of prototypes for spectral reconstruction. To see this more clearly, recall that the convolutional features are computed using a set of filters. If the ith filter in our set is applied to the kth wavelength indexed band across the image lattice, we can write

$$f_{i,\lambda_k} = I_{\lambda_k} * h_i = L(\lambda_k) r_{\lambda_k} \circ (g * h_i) \qquad (5)$$

where the right-hand side of the equation follows the distributive property of the convolution. In Equation 5, we have omitted the pixel indices as a matter of convenience ($I_{\lambda_k} \equiv I(\cdot, \lambda_k)$, $g(\cdot) \equiv g$ and $r_{\lambda_k} \equiv r(\cdot, \lambda_k)$), h_i is the filter mask and, as usual, $*$ and $\circ$ indicate the convolution and Hadamard products, respectively.

As a result, for each wavelength-indexed band in our training set, the spectra becomes a multiplicative constant as applied to the convolutional features computed from the image intensities. Thus, for our spectral reconstruction approach we use the vectors given by

$$\gamma(u) = [r(u, \lambda_1)\mathbf{v} \,||\, r(u, \lambda_2)\mathbf{v} \,||\, \ldots \,||\, r(u, \lambda_N)\mathbf{v}] \qquad (6)$$

where $\mathbf{v}$ is a row vector whose i^{th} entry corresponds to the filter response $g*h_i$ at pixel u, N is the number of wavelength-indexed bands in our training imagery and we have written $||$ to imply vector concatenation.

The use of the vectors in Equation 6 for our reconstruction approach has two main advantages. Firstly, it permits a consistent treatment of both, our hyperspectral image training set and our colour testing image. Secondly, it allows for a constrained sparse coding scheme to be used to recover prototypes that can then be employed to perform the spectral reconstruction. This is since we can always convert the vectors $\gamma(u)$ to their rg-chromaticity analogues by extracting the values of $r(u, \cdot)$ so as to apply Equation 4. This can be done in a straightforward manner by noting that the value of $r(u, \lambda_1)$ repeats over the length of the convolutional features across the vector $\gamma(u)$.

4.2 Constrained Sparse Coding

Here, we view the problem in hand as that of representing the vectors $\gamma(u)$ using a set Φ of prototypes such that

$$\gamma(u) = \sum_{\phi \in \Phi} \alpha_\phi(u)\phi \qquad (7)$$

where $\alpha_\phi(u)$ can be seen as the "abundance" of the prototype ϕ in $\gamma(u)$.

This treatment responds to a twofold motivation. Firstly, we exploit the fact that, in this manner, the set of prototypes can be obtained using sparse coding. Secondly, Equation 7 implies that the prototype ϕ should have a structure akin to that of $\gamma(u)$. This is important since the spectral and convolutional feature components in ϕ can then be used to reconstruct the testing image making use of its chromaticity information.

We will elaborate upon this later on. For now, we focus on the recovery of the prototype set Φ. To this end, we aim at solving the optimization problem given by

$$\min_{\Phi, A} \left\{ \sum_{u \in \mathcal{I}} |\gamma(u) - \alpha(u)^T \Phi|_2 + \tau_1 \sum_{u \in \mathcal{I}} |\alpha(u)|_1 \right\} \qquad (8)$$

subject to

$$|\alpha(u)|_2 \leq \kappa \qquad \forall \quad u \in \mathcal{I}$$
$$\alpha_\phi(u) \geq 0 \qquad \forall \quad \alpha_\phi(u) \in \alpha(u)$$

where $|\cdot|_p$ denotes the p-norm, A is the set of column weight vectors $\alpha(u)$ for all the pixels in our training data, κ is a bounding positive constant and τ_1 is a scalar that accounts for the contribution of the second term in Equation 8 to the minimisation in hand. Note that we have added the positivity constraint on the vector $\alpha(u)$ following the notion that the contributions of the primitives to the pixel spectrum correspond to an additive process. This is consistent with the notion that the weight vector entry $\alpha_\phi(u)$ can be seen as the "abundance" of the prototype ϕ in $\gamma(u)$. Also, we have imposed a norm constraint in order to avoid the possibility of degenerate cases involving linear transformations of $\alpha(u)$. Note that this is akin to the approach taken in [16].

4.3 Reconstructing the Image Irradiance

With the prototypes in hand, we now turn our attention to the spectral reconstruction of a testing image I^*. To reconstruct the spectra, we commence by computing the intensity of the testing image. With the intensity in hand, we proceed to compute the chromaticity values $K_c(v)$, $c = \{R, G, B\}$ and the convolutional feature $\mathbf{w}$ for each pixel v in the testing image.

Recall that each of the prototypes $\phi \in \Phi$ shares its structure with that of $\gamma(u)$ as defined in Equation 6. This is as a result of the linear relationship in Equation 7 and the positivity constraint introduced in our sparse coding approach. This is important since we can write

$$\phi = [s(\lambda_1)\mathbf{x} \,||\, s(\lambda_2)\mathbf{x} \,||\, \ldots \,||\, s(\lambda_N)\mathbf{x}] \qquad (9)$$

where $s(\cdot)$ can be viewed as the spectral component of ϕ whereas $\mathbf{x}$ arises from the convolutional features.

With these ingredients, we aim at recovering the weight vector $\rho(v)$ such that the spectral reconstruction of the testing pixel v becomes

$$R(v, \cdot) = g(v)\rho(v)^T \mathbf{S} \qquad (10)$$

such that $\mathbf{w} = \rho(v)^T\mathbf{X}$, where $\mathbf{X}$ and $\mathbf{S}$ are matrices whose rows correspond to the vectors $\mathbf{x}$ and $s(\cdot)$ as defined in Equation 9 and $g(v)$ is the intensity at pixel v.

Thus, both, the convolutional feature $\mathbf{w}$ and the spectral reconstruction of the pixel v are given by the linear combination of the spectral and convolutional components of the prototypes $\phi \in \Phi$. To determine the weight vector $\rho(v)$, we note that the relationship in Equation 10 applies also to the chromaticity values. That is, the chromaticity $K_c(v)$ of the testing pixel is given by

$$K_c(v) = \rho(v)^T\mathbf{K}_c \qquad (11)$$

where $\mathbf{K}_c$ is a matrix whose rows correspond to the chromaticities $K_c(s)$, $c = \{R, G, B\}$ of the vectors $s(\cdot)$.

Thus, we construct the vectors

$$\beta(v) = [K_R(v)\mathbf{w} \,||\, K_G(v)\mathbf{w} \,||\, K_B(v)\mathbf{w}] \qquad (12)$$

for each of the pixels v in the testing image and solve the optimisation

$$\min_{\rho(v)}\left\{|\beta(v) - \rho(v)^T\Omega|_2 + \tau_2|\rho(v)|_1\right\} \qquad (13)$$

subject to

$$\rho(v)^T\Omega \geq 0 \qquad \forall \quad \rho(v)_\omega \in \rho(v)$$

where τ_2 is a scalar that controls the degree of sparsity in the resulting weight vector, Ω is a matrix whose rows are given by the vectors

$$\omega = [K_R(s)\mathbf{x} \,||\, K_G(s)\mathbf{x} \,||\, [K_B(s)\mathbf{x}] \qquad (14)$$

and $\rho_\omega(v)$ is the entry of $\rho(v)$ corresponding to ω.

Here, we have added the positivity constraint on the weight column vector $\rho(v)$ following the notion that the spectra cannot exhibit negative values. The vectors $\omega \in \Omega$ are analogue to the prototypes $\phi \in \Phi$ presented in Section 4.1. The main difference, however, is that these are defined in the rg-chromaticity space rather than the image spectral irradiance. This is not a problem, however, since the rg-chromaticity values for $s(\cdot)$ can be computed in a straightforward manner using Equation 4 if the illuminant power spectrum for the training imagery and spectral sensitivity functions for the testing image are known.

4.4 Illuminant Power Spectrum Recovery

With the spectra in hand, we can recover the illuminant power spectrum $L(\cdot)$ for the testing image I^* making use of the colour information for each of the colour channels $I_c^*(\cdot)$, $c = \{R, G, B\}$. Making use of Equation 3, and since the spectral sensitivity function $Q_c(\cdot)$ is known, we can solve the following non-negative optimization problem

$$\min_{L(\cdot)}\left\{\sum_{\substack{v \in I^* \\ c \in \{R,G,B\}}} |I_c^*(v) - \sum_{\lambda_k \in \Lambda} L(\lambda_k)Q_c(\lambda_k)R(v, \lambda_k)|_2\right\}$$

$$(15)$$

subject to $L(\lambda_k) \geq 0$ for all $\lambda_k \in \Lambda$.

5. IMPLEMENTATION AND DISCUSSION

In Algorithm 1 we show the pseudocode for the training stage of our approach. There are a number of traits that merit further discussion. Note that the algorithm takes at input a dataset of training hyperspectral images with the illuminant removed. This is, for each of the images in the training data set, we use

$$R(u, \lambda_k) = \frac{I(u, \lambda_k)}{L(\lambda_k)} \qquad (16)$$

as an alternative to the image irradiance $I(u, \lambda_k)$ at input. This is reflected in Line 3, of Algorithm 1. We have done this since the illuminant power spectrum $L(\cdot)$ for the training images is assumed to be known and, hence, $R(u, \lambda_k)$ can be computed in a straightforward manner.

For the convolutional features, we have used the three filter banks in [33]. It is also worth noting that the constrained sparse coding problem in Equation 8 can be solved in a number of ways. Here, we have opted for a basis pursuit [30] where the non-negative constraint is enforced explicitly. For initialisation, we have used the mode seeking method in [34]. This follows the intuition that the prototypes are expected to correspond to the modes of the distribution of the vectors $\gamma(u)$ over the training images.

In Line 12 of the the algorithm we recover, for each prototype, the corresponding spectral and convolutional components. This can be done via a simple division following by a normalisation step. The reason being that, for convolutional features of length M, the first M elements of ϕ will account for $s(\lambda_1)\mathbf{x}$. As a result, by concatenating these N times, i.e. the number of wavelength-indexed bands in our training images, we can employ the Hadamard product and Equation 9 so as to write

$$\phi \circ \zeta^{-1} = \left[\mathbf{1}||\, \frac{s(\lambda_2)}{s(\lambda_1)}\mathbf{1} \,||\, \dots \,||\, \frac{s(\lambda_N)}{s(\lambda_1)}\mathbf{1}\right] \qquad (17)$$

where $\mathbf{1}$ is a vector whose dimensionality is the same as the convolutional features and ζ is a vector comprised of the first M elements of ϕ concatenated N times.

From Equation 17 is straightforward to note that the ith unique value in $\phi \circ \zeta^{-1}$ will account for the quantity $\frac{s(\lambda_i)}{s(\lambda_1)}$. That is, these values are proportional to the spectral components $s(\lambda_i)$ up to the inverse of $s(\lambda_1)$. At this point, we also note that, by definition, $r(u, \cdot)$ is normalised to unity. Hence, it follows that, once the values $\frac{s(\lambda_i)}{s(\lambda_1)}$ are in hand, they can be normalised such that $|s(\cdot)|_2 = 1$ so as to recover the spectral component of ϕ. Once $s(\cdot)$ is computed, the convolutional component $\mathbf{x}$ can be retrieved using the equation

$$\frac{1}{s(\lambda_1)}\zeta = [\mathbf{x}||\mathbf{x}||\dots||\mathbf{x}] \qquad (18)$$

and noting that $\mathbf{x}$ repeats N times on the right-hand side of the expression above.

In Algorithm 2, we show the step sequence for the testing phase of our algorithm. In our code, we have also used basis pursuit for the solution of the optimisation in Equation 13. Also, in our implementation, the intensity of the testing image is given by the lightness dimension, i.e. L, of the image in the CIELab colour space. For the optimisations in Equations 8 and 13 we have set $\tau_1 = \tau_2 = 1/N$, i.e. the inverse of the number of wavelength indexed bands in our training imagery, and $\kappa = 1$.

Regarding our illuminant recovery step, it is worth noting in passing that the optimisation in Equation 15 is somewhat similar to the expression used in [25]. The main difference stems from the lack of a regularisation term which imposes a smoothness constraint on the illuminant power spectrum.

Algorithm 1 Training
──────────────────────────────────────
Require: $\mathcal{I}$: Dataset of training hyperspectral images with the illuminant removed
1: **for** Every image in $\mathcal{I}$ **do**
2:    **for** Every pixel u in the image **do**
3:       Compute the image intensity $g(u) = |I(u, \cdot)|_2$
4:    **end for**
5:    Compute the convolutional features using the image intensities
6:    **for** Every pixel u in the image **do**
7:       Construct $\gamma(u)$ making use of Equation 6.
8:    **end for**
9: **end for**
10: Compute Φ by solving Equation 8
11: **for** Every prototype ϕ in Φ **do**
12:    Recover the spectral and convolutional components so as to satisfy Equation 9.
13: **end for**
14: **return** Set of spectral components of Φ and set X of convolutional components of Φ.
──────────────────────────────────────

Algorithm 2 Testing
──────────────────────────────────────
Require: Input colour image, set of spectral components of Φ, set X of convolutional components of Φ, spectral sensitivity functions $Q_c(\cdot)$ for the sensor used to acquire the testing image
1: White balance the input image
2: **for** Every $\phi \in \Phi$ in the prototype set **do**
3:    Compute the chromaticity values using Equation 4.
4:    Construct the vector ω using Equation 14.
5: **end for**
6: **for** Every pixel v in the input image **do**
7:    Compute the chromaticity values using Equation 4.
8:    Compute the image intensity $g(v)$
9: **end for**
10: Compute the convolutional features using the image intensities
11: **for** Every pixel v in the input image **do**
12:    Construct the vector $\beta(v)$ as per Equation 12.
13:    Recover the weight vector $\rho(v)$ by solving the optimisation problem in Equation 13.
14:    Compute the spectrum $R(v, \cdot)$ using Equation 10.
15: **end for**
16: Compute the illuminant power spectrum by solving the optimisation in Equation 15
17: **return** The reconstructed spectra and the illuminant power spectrum for the testing image
──────────────────────────────────────

Note that the number of variables in $L(\lambda)$ is given by the number of wavelength indexed bands comprising the spectra $R(v, \cdot)$. For N discrete wavelengths $\lambda_1, \ldots, \lambda_N$, the recovered image spectra has a dimension of $P \times N$, where P is the number of pixels in the testing image. The optimisation problem in Equation 15 is generally well-formed if there are more pixels in the image than the number of wavelength indexed channels, *i.e.* $P \geq N$. As a result, the minimisation in hand can be solved using standard non-negative least squares methods.

6. EXPERIMENTS

6.1 Datasets and Settings

In this section, we commence by elaborating on the datasets used in our experiments. Later on, we present a quantitative analysis for our approach and illustrate its utility for illuminant substitution and transfer, film simulation and mood board colour transfer.

For the experiments presented in this section, we use a number of hypersepctral image datasets of rural and urban environments for both, training and testing. For our experiments, we also use the spectral sensitivity functions for commercial camera models presented in [18][2], spectral sensitivity functions for Fuji and Kodak film, real-world imagery acquired RAW with a Nikon D80 and Nikon D5100 cameras and reflectance and illuminant power spectra captured in house using a spectrometer.

On the hyperspectral datasets used here, the first of these is comprised by the reflectance images of 16 scenes captured by Foster *et al.* [7][3]. These images were acquired with a Peltier-cooled digital camera and a fast VariSpec tunable liquid-crystal filter producing a spatial resolution of 1344×1024 pixels and a spectral resolution of 10 nm over the wavelength range $[400nm, 720nm]$. The intensity response was recorded with 12-bit precision and was linear

to the image irradiance. To compute image reflectance, the illumination spectrum was obtained from a grey Munsell surface placed in the scene.

The second of our datasets is that presented in [25][4]. The dataset is comprised of 64 images acquired using a Specim camera with a spectral resolution of 10 nm in the visible spectrum. It is worth noting that the dataset has been divided into testing and training sets. Here, all our experiments have been effected using the latter of these for training our method.

Finally, we have also employed a subset of the Scyllarus Series A dataset of spectral images[5]. Our subset is given by the 52 images in the full dataset which contain a Spectralon calibration target in the scene and, hence, the image reflectance can be computed from the irradiance. These images were acquired with a Liquid Crystal Tunable Filter (LCTF) tuned at intervals of 10 nm in the visible spectrum. The intensity response was recorded with a low distortion intensified 12-bit precision camera.

For our quantitative experiments and presentation of results, unless otherwise noted, we have employed the CIE colour sensitivity functions proposed by Judd [13]. All the spectral reconstructions performed herein cover the range $[400nm, 700nm]$ in $10nm$ steps. Also, note that using the full set of pixels from the 40 training images is, in practice, infeasible. As a result, for our method and the alternatives in [2] and [1] we have used the Markov chain Monte Carlo (MCMC) sampling strategy in [4] and set the number of prototypes delivered by our sparse coding approach to $11,800$. For the method in [25], we have followed the authors and used k-means to obtain $16,000$ reflectance samples.

---

[2]Accessible at http://www.cis.rit.edu/jwgu/research/camspec/

[3]The imagery can be accessed at: http://personalpages.manchester.ac.uk/staff/d.h.foster/

[4]The dataset can be downloaded from: http://www.comp.nus.edu.sg/~whitebal/spectral_reconstruction/

[5]Downloadable at: http://www.scyllarus.com

		Foster	Scyllarus	NUS
Abed et al. [1]	GW	28.96	26.85	18.69
	WP	26.98	23.93	16.82
	General GW	27.94	25.92	17.96
	1st GE	27.17	24.96	17.20
	2nd GE	27.03	24.48	16.91
Agahian et al. [2]	GW	58.98	44.24	38.19
	WP	49.08	39.42	33.65
	General GW	53.79	42.91	37.53
	1st GE	50.67	41.92	35.81
	2nd GE	50.75	41.87	35.98
Nguyen et al. [25]	GW	25.67	23.98	18.85
	WP	21.62	18.76	14.91
	General GW	23.71	20.75	16.46
	1st GE	21.95	19.73	14.87
	2nd GE	21.59	19.36	15.07
Our approach	GW	17.81	14.23	13.21
	WP	14.32	**12.66**	11.61
	General GW	15.07	13.99	12.81
	1st GE	**14.14**	13.34	11.31
	2nd GE	14.45	13.19	**11.23**

Table 1: Average per-pixel spectral reconstruction errors (in degrees) yielded by our approach and the three alternatives for the datasets under study and a number of white balancing methods. The absolute lowest error per dataset is in bold font. We have underlined the lowest error for each of the alternatives as applied to each dataset.

		Foster	Scyllarus	NUS
Nguyen et al. [25]	GW	27.86	23.98	19.18
	WP	26.21	21.66	16.88
	General GW	25.17	23.52	17.99
	1st GE	22.53	22.06	16.71
	2nd GE	22.32	21.73	17.28
Our approach	GW	18.56	15.11	14.06
	WP	15.67	**13.35**	12.47
	General GW	15.22	15.12	13.47
	1st GE	**14.23**	14.62	12.67
	2nd GE	15.01	14.01	**12.25**

Table 2: Illuminant angular error (in degrees) yielded by our approach and the alternative in [25] for the datasets under study. The absolute lowest error per dataset is in bold font. We have underlined the lowest error for each of the two methods as applied to each dataset.

6.2 Quantitative Evaluation

We commence by showing quantitative results on both, the spectral reconstruction and the illuminant recovery for our method and a number of alternatives. These are the method presented in [25], which is based on a Radian Basis Function (RBF) network, the weighted PCA algorithm in [2] and the linear interpolation approach of Abed *et al.* [1].

For all the methods, we have used the training images provided in the NUS [25] dataset and quantified the spectral reconstruction error using the Euclidean angle in degrees between the ground truth and the reconstructed values of $R(u, \cdot)$. We opt for this error measure as it is widely used in previous works [32]. Note that the other error measure used elsewhere is the RMS error [25]. It is worth noting, however, that the Euclidean angle and the RMS error are correlated when the spectra is normalised to unit L2-norm.

In Table 1, we show the angular errors for the spectral reconstruction on the three datasets and the methods under study when the grey world (GW) assumption [3], white patch (WP) algorithm [15] and the 1st and 2nd order grey edge (GE) methods [32] are used to perform white balancing at testing. Note that the absolute best (in bold), is always yielded by our method, followed by the approach in [25].

In Table 2 we show the angular error for the recovered illuminant for both, our method and the alternative in [25]. This is since the other alternatives do not aim at recovering the illuminant. Note that, again, our method consistently delivers the lowest angular error. Moreover, the results in the table are quite consistent with those in Table 1.

Here, we also quantify the difference in pseudocolor, *i.e.* the reconstructed image converted into colour using the colour sensitivity functions in [13]. between the reconstructed image spectra and the input testing imagery. We do this follow-

ing the intuition that a good spectral reconstruction should yield a pseudocolor image whose colour difference with respect to the ground truth is small. To quantify the colour difference, we have used the CIEDE2000 colour difference [29]. In Table 3, we show the colour difference per dataset for each of the alternatives and white balancing strategies. Note that the colour difference for our method is the lowest, consistently being about 1 unit, which is within the just noticeable chromatic difference.

Finally, we use the dataset in [18] to evaluate the effect of different spectral sensitivity functions in our method and the alternatives. To this end, we have tested the methods using pseudocolour images, *i.e.* colour imagery generated using the hyperspectral images and spectral sensitivities for the 28 camera models in [18] for each of the datasets. In Table 4, we show the mean and standard deviation for the

		Foster	Scyllarus	NUS
Abed et al. [1]	GW	9.11	7.85	5.89
	WP	7.99	6.99	5.12
	General GW	8.54	7.44	5.20
	1st GE	8.18	7.11	5.37
	2nd GE	8.05	7.23	5.62
Agahian et al. [2]	GW	15.82	12.78	9.99
	WP	12.85	9.56	8.47
	General GW	14.78	11.43	9.32
	1st GE	13.67	9.87	9.19
	2nd GE	12.99	10.76	8.92
Nguyen et al. [25]	GW	5.69	6.18	3.93
	WP	6.54	5.58	3.59
	General GW	7.48	5.73	3.77
	1st GE	5.63	4.68	3.32
	2nd GE	4.43	4.52	3.29
Our approach	GW	2.57	2.32	2.14
	WP	1.42	**1.03**	**1.02**
	General GW	2.21	2.43	1.02
	1st GE	1.32	1.21	1.09
	2nd GE	**1.08**	1.34	1.04

Table 3: Average per-pixel CIEDE2000 colour difference for the three datasets. The absolute lowest difference per dataset is in bold font. We have underlined the lowest error for each of the alternatives as applied to each dataset.

	Abed et al. [1]	Agahian et al. [2]
	Spectra	Spectra
Foster	27.04±3.56 (WP)	49.12±2.68 (WP)
	29.24±4.06 (GW)	58.95±3.63 (GW)
Scyllarus	24.24±3.25 (WP)	39.35±3.96 (WP)
	26.94±3.79 (GW)	44.20±4.29 (GW)
NUS	16.14±3.27 (WP)	33.37±3.47 (WP)
	19.48±3.72 (GW)	39.09±3.45 (GW)
	Nguyen et al. [25]	
	Spectra	Illuminant
Foster	21.14±2.32 (2nd GE)	22.18±1.32 (2nd GE)
	26.36±2.98 (GW)	28.42±3.31 (GW)
Scyllarus	19.05±1.29 (WP)	21.95±1.68 (WP)
	23.82±2.16 (GW)	23.82±2.16 (GW)
NUS	14.48±1.98 (1st GE)	16.75±1.07 (1st GE)
	18.12±1.28 (GW)	19.94±2.58 (GW)
	Our approach	
	Spectra	Illuminant
Foster	**14.21±1.28 (1st GE)**	**14.08±1.43 (1st GE)**
	18.45±2.84 (GW)	19.45±3.43 (GW)
Scyllarus	**12.82±1.69 (WP)**	**13.88±1.95 (WP)**
	14.29±2.31 (GW)	15.11±2.01 (General GW)
NUS	**11.17±1.29 (2nd GE)**	**12.19±1.09 (2nd GE)**
	13.43±1.29 (GW)	14.16±1.18 (GW)

Table 4: Per-pixel angular difference for the three datasets when the spectral sensitivity functions of commercial cameras are used to generate the testing imagery. The white balancing methods (in parenthesis) shown are those that previously delivered the best and worst performance. Absolute best per dataset are in bold fonts.

per-pixel angular error for each of the alternatives yielded by the best and worst performing white balancing algorithm. In the table, the corresponding white balancing method is in parenthesis. We only show the illuminant recovery results for our method and the approach in [25], which do deliver the illuminant at output.

It is worth noting in passing that, as expected, in the tables, all the alternatives do better in the NUS dataset. This is since we have used the NUS subset for training. Nonetheless, our method is able to deliver good results on the other datasets regardless. This hints that our approach can generalise well, being trained in a particular dataset and then used to process imagery from other sources.

6.3 Illuminant Substitution and Transfer

Now, we turn our attention to a sample application relevant to image editing. In this section, we show how the reconstructed spectra and recovered illuminant power spectrum can be used to substitute or transfer the illuminant from one scene to another. This stems from the observation that, once the spectra $R(u, \cdot)$ and the illuminant power spectrum are in hand, we can recover the new image in a straighforward manner by evaluating Equation 3.

For our experiments, we have used two widely available commercial cameras. These are a Nikon D80 and a Nikon D5100. The imagery presented here has all been acquired in RAW format and, for our method, we have performed white balancing using the 2nd order grey edge method [32]. The spectral sensitivity functions are those in [18] and the training set is, again, the subset of spectral imagery taken from the NUS dataset.

In Figure 2, we show two sets of images where the illuminant has been transferred from one on to the other. In the figure, the top row shows the images used for extracting the illuminant transferred to the imagery in the second row.

The third row shows the white balanced imagery. We have included these since it can be interpreted as true "white" light. The bottom row shows the illuminant substitution results when the power spectrum of the light in the images on the top row is used to compute the colour for the ones in the second row. Note that, in the figure, the changes in the imagery on the left are quite subtle whereas those in the right-hand side are much more evident. Nonetheless, we can appreciate that the tones on the stainless steel escalator are in good accordance with the metallic surfaces in the top image and the green walls appear more yellow.

Finally, in Figure 3, we show a set of images where the illuminant has been substituted by that of a tungsten (middle column) and fluorescent lamp (right-hand column). We have acquired the power spectrum of these light sources in house using a spectrometer equipped with an integrating sphere. Here, we also see the expected effects due to the changes in the illuminant. Note that the middle column imagery is "warmer", which is a common effect of incandescent and tungsten lamps. The right-hand panels, in the other hand, show shifts of the colour balance towards the green channel.
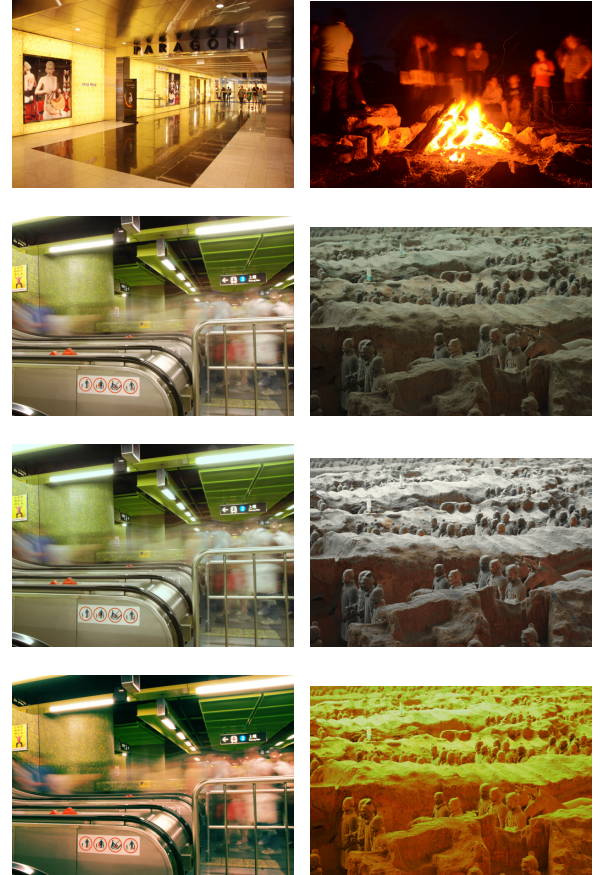
Figure 2: Illuminant transfer results. We have transferred the illuminant from the imagery in the top row to that in the second rows The third row shows the images in the second row after white balancing. The bottom two rows show the images in the second row lit by the illuminant in the top images.
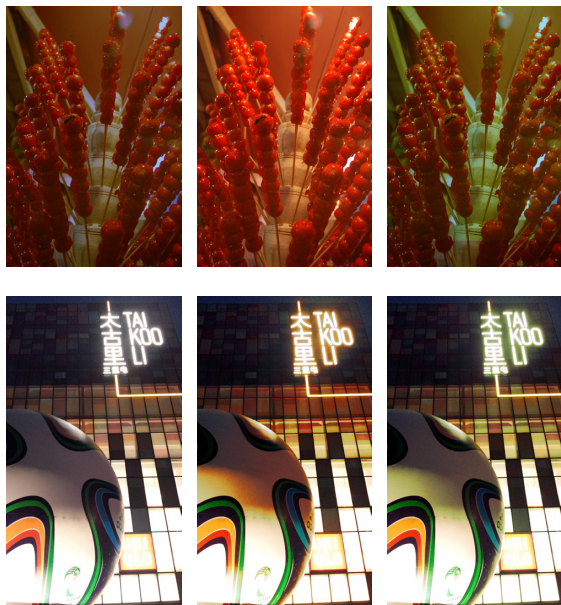
Figure 3: Illuminant substitution results. Left-hand column: Input images; Middle column: Images when the illuminant has been subsituted by an incandescent light; Right-hand column: Images with the illuminant substituted by a fluorescent lamp.

6.4 Film Simulation

We now turn our attention to the use of spectral sensitivity functions other than that of the sensor used to acquire the image to present the scene to the user. This can be done in a straightforward manner by setting $Q_c(\cdot)$ in Equation 3 to an alternative. This, in effect, allows for the presentation of the image to the user with the colour rendition of a completely different medium, *i.e.* film as compared to digital.

To illustrate the effects of doing this, in Figure 4 we have taken two sample pictures acquired with a Nikon D80 and recovered their spectral reconstruction using our method. As done in the previous section, for all our images, we have performed white balancing using the 2nd order grey edge method [32]. For all the imagery, we have produced colour imagery using the spectral sensitivity function corresponding to the Fujichrome 64T Type II Professional[6] and the Kodak HD 400[7] and T-MAX 100 films[8].

Note that, in the figure, the images produced using the Fujichrome 64T Type II Professional show a very faint greenish colour tint whereas the ones produced using the sensitivity function of the Kodak HD 400 appear to be better colour balanced. This is consistent with the Fujichrome 64T film being a tungsten film as compared to the Kodak HD 400, which is a daylight film. This is also consistent with the notion that different types of film will deliver distinctive colours with particular look and feel. In the other hand,

---

[6]Available at http://www.fujifilmusa.com/support/ServiceSupportProduct.do?prodcat=238599
[7]Accessible at http://www.kodak.com/global/en/consumer/products/techInfo/e7013/e7013.shtml
[8]Downloadable from http://www.kodak.com/global/en/professional/support/techPubs/f32/f32c.jhtml

the imagery produced using the sensitivity functions for the Kodak T-MAX 100 has a very different character as compared to the colour panels in the figure.

6.5 Mood Board Colour Transfer

Finally, we explore the use of a mood board for recolouring the image accordingly. Mood boards are collections of colours, filters or images that convey a mood or feeling. These are used by arts and media professionals as part of the creative process. Here, we have used a set of tiles from an XRite ColorChecker Digital SG to create a mood board in a manner akin to that used for Lee filters[9].

For each of our two mood boards, we have selected a set of tiles from the colour chart. We have then acquired the reflectance of these tiles in house with a StellarNet spectrometer and used these reflectance spectra to "unmix" the reconstructed image irradiance for three sample scenes captured with a Nikon D5100. Spectral unmixing is widely used in remote sensing so as to express the relative abundance of primordial materials, *i.e.* end members, in multispectral or hyperspectral imagery. Thus, here, after performing our spectral reconstruction, we have applied the method in [11] so as to recover the spectral reflectance. With the reflectance in hand, we have then used the unmixing method in [23] so as to recover the abundances of the mood board spectra that account for the best linear approximation of the image reflectance, *i.e.* the abundances of the XRite ColorChecker tiles whose reflectance best mixes into the image spectra.

In Figure 5, we show the results of applying the two mood boards to our input images. As in the previous section, we have used, for our approach, the 2nd order grey edge white balancing method [32]. Note that, in principle, the same unmixing approach can be applied in a straightforward manner to the image colour in the rg-chromaticity space. As a result, in the figure, we have also included the results yielded by the unmixing in the chromaticity space rather than in the spectral domain.

Note that the colour-based results are much more similar to one another than those yielded by our spectral reconstruction. This is due to the fact that, as a result of Grassmann's law [35], any value inside the simplex formed by the mood board colours on the CIE XY chromaticity diagram will be reproduced by the unmixing process. This contrasts with the spectral domain, where each of the pixel colours arises from the application of Equation 3 to the mixed spectra and, hence, it does not abide to Grassmann's law. This can be clearly appreciated in the figure, where the results yielded by our spectral reconstruction have clearly acquired the colour scheme of the corresponding mood boards.

7. CONCLUSIONS

In this paper, we have presented a method to perform spectral reconstruction from a single colour image making use of training data. The method presented here is based upon a set of spectral and convolutional features. This is an important trait of our approach, leveraging material appearance information to better capture the reflectance properties of objects in the training set. We have also shown how the illuminant power spectrum can be recovered via a quadratic optimisation step. We have performed a qualitative analysis and compared against a number of alternative methods. We

---

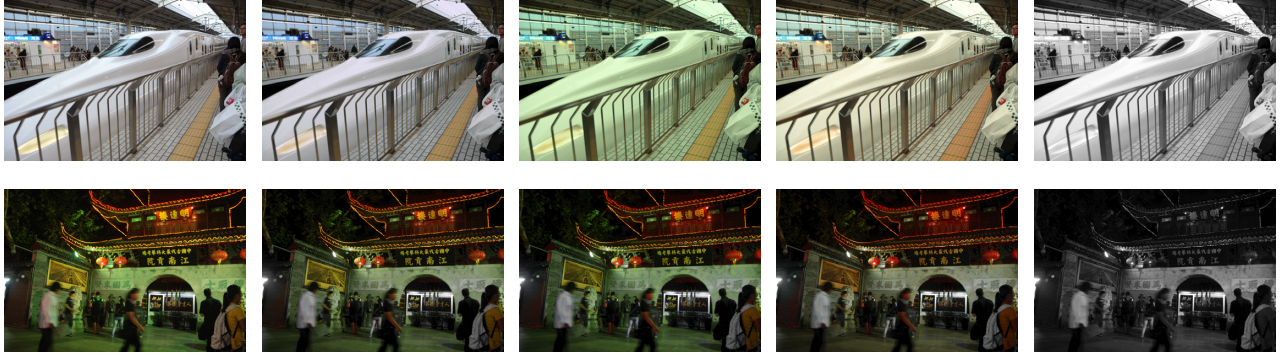[9]See http://www.leefilters.com/lighting/moods.html

Figure 4: Film simulation results. We have performed spectral reconstruction on the images in the right-hand column and simulated three commercial films. The second column shows the white balanced images. The remaining columns show the result of applying a uniform irradiance illuminant and the spectral sensitivity functions for Fujichrome 64T Type II Professional, Kodak HD 400 and Kodak T-MAX 100, respectively.
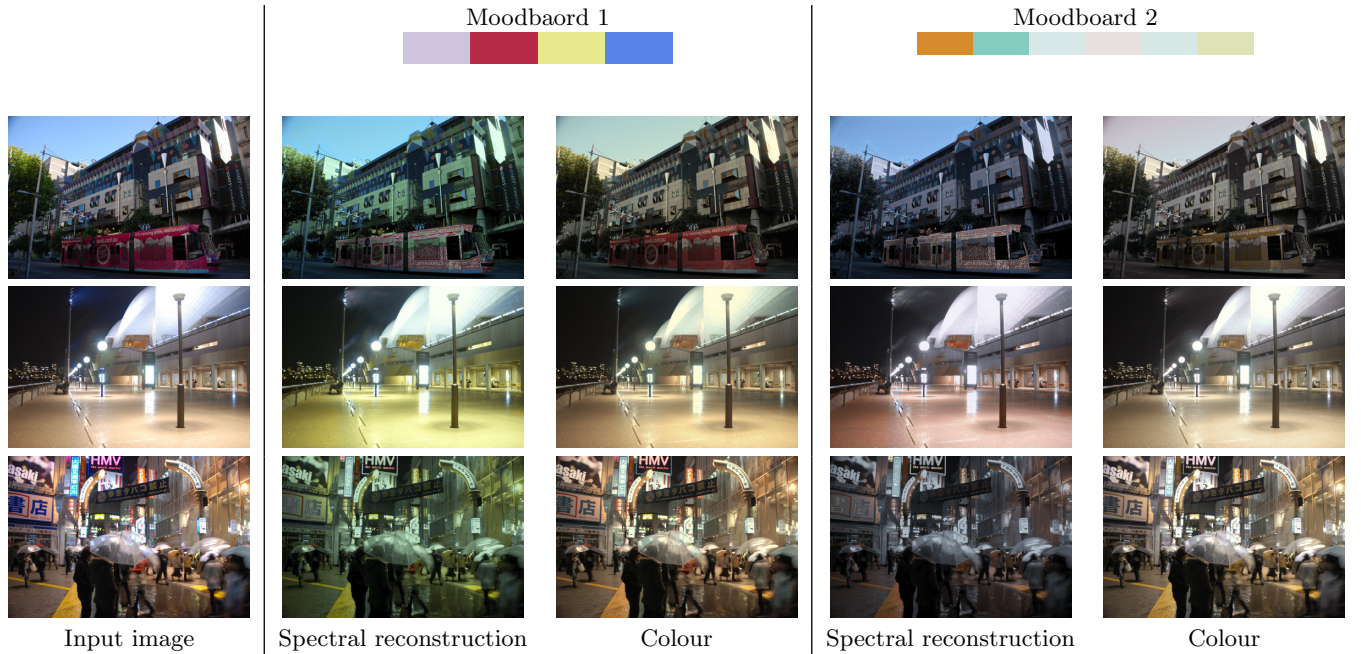
Moodbaord 1   Moodboard 2

Input image   Spectral reconstruction   Colour   Spectral reconstruction   Colour

Figure 5: Mood board colour transfer results. We have performed spectral reconstruction on the images in the right-hand column and used the spectral reflectance of the mood board tiles to further decompose the scene reflectance. In the figure, we show results for two mood board tile sets. We also show results when the unmixing is effected in the rg-chromaticity space ("Colour" labelled columns) rather than on the spectral reconstruction.

have also illustrated the utility of our approach for illuminant transfer and substitution, film look and feel simulation and the application of mood board colour schemes to novel imagery.

Acknowledgements

8. REFERENCES

[1] F. M. Abed, S. H. Amirshahi, and M. R. M. Abed. Reconstruction of reflectance data using an interpolation technique. *Journal of the Optical Society of America A*, 26(3):613–624, 2009.

[2] F. Agahian, S. A. Amirshahi, and S. H. Amirshahi. Reconstruction of reflectance spectra using weighted principal component analysis. *Color Research & Application*, 33:360–371, 2008.

[3] G. Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin Institute*, 310(1):337–350, 1980.

[4] J. Domke and X. Liu. Projecting ising model parameters for fast mixing. In *Neural Information Processing Systems*, pages 665–673, 2013.

[5] T. Ejaz, T. Horiuchi, G. Ohashi, and Y. Shimodaira. Development of a camera system for the acquisition of high-fidelity colors. *IEICE Trans. on Electronics*, E89–C(10):1441–1447, 2006.

[6] G. Finlayson, S. Hordley, and P. Hubel. Recovering device sensitivities with quadratic programming. In *Proceedings of the IS&T/SID Color Imaging Conference*, pages 90–Ű95, 1998.

[7] D. H. Foster, K. Amano, S. M. C. Nascimento, and M. J. Foster. Frequency of metamerism in natural scenes. *Journal of the Optical Society of America A*, 23(10):2359–2372, 2006.

[8] L. Gu, C. P. Huynh, and A. Robles-Kelly. Material-specific user colour profiles from imaging spectroscopy data. In *IEEE International Conference on Computer Vision*, 2011.

[9] L. Gu, A. Robles-Kelly, and J. Zhou. Efficient estimation of reflectance parameters from imaging spectroscopy. *IEEE Trans. on Image Processing*, (99):1–1, 2013.

[10] S. Han, Y. Matsushita, I. Sato, T. Okabe, and Y. Sato. Camera spectral sensitivity estimation from a single image under unknown illumination by using fluorescence. In *Computer Vision and Pattern Recognition*, 2012.

[11] C. P. Huynh and A. Robles-Kelly. A solution of the dichromatic model for multispectral photometric invariance. *International Journal of Computer Vision*, 90(1):1–27, 2010.

[12] J. Jiang, D. Liu, J. Gu, and S. Süsstrunk. What is the space of spectral sensitivity functions for digital color cameras? In *Workshop on Applications of Computer Vision*, pages 168–179, 2013.

[13] D. B. Judd. Report of U.S. secretariat committee on colorimetry and artificial daylight. page 11, 1951.

[14] R. Kawakami, H. Zhao, R. Tan, and K. Ikeuchi. Camera spectral sensitivity and white balance estimation from sky images. *International Journal of Computer Vision*, 105(3):187–204, 2013.

[15] E. H. Land and J. J. Mccann. Lightness and retinex theory. *Journal of the Optical Society of America*, 61:1–11, 1971.

[16] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *Neural Information Processing Systems (NIPS)*, pages 801–808, 2006.

[17] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three dimensional textons. *International Journal of Computer Vision*, 43(1):29–44, 2001.

[18] H. T. Lin, S. J. Kim, S. Süsstrunk, and M. S. Brown. Revisiting radiometric calibration for color computer vision. In *International Conference on Computer Vision*, pages 129–136, 2011.

[19] P. Longere and D. H. Brainard. Simulation of digital camera images from hyperspectral input. In *Vision Models and Applications to Image and Video Processing*.

[20] L. Maloney. Evaluation of linear models of surface spectral reflectance with small numbers of parameters. *Journal of the Optical Society of America A*, 50(1):1673–1683, 1983.

[21] R. Martin, Z. Arno, and K. Reinhard. Practical spectral characterization of trichromatic cameras. In *SIGGRAPH Asia*, 2011.

[22] J. Nakamura. *Image Sensors and Signal Processing for Digital Still Cameras*. CRC Press, 2006.

[23] J. M. P. Nascimento and J. M. B. Dias. Vertex component analysis: A fast algorithm to unmix hyperspectral data. *IEEE Trans. on Geoscience and Remote Sensing*, 43(4):898–910, 2005.

[24] R. M. H. Nguyen, D. K. Prasad, and M. S. Brown. Raw-to-raw: Mapping between image sensor color responses. In *Computer Vision and Pattern Recognition*, 2014.

[25] R. M. H. Nguyen, D. K. Prasad, and M. S. Brown. Training-based spectral reconstruction from a single RGB image. In *European Conference on Conputer Vision*, pages 186–201, 2014.

[26] J. Romero, A. García-Beltrán, and J. Hernández-Andrés. Linear basis representation of natural and artificial illuminants. *Journal of the Optical Society of America*, 15:2913–2920, 1997.

[27] G. Schwartz and K. Nishino. Visual material traits: Recognizing per-pixel material context. In *Proc. of Color and Photometry in Computer Vision*, 2013.

[28] G. Sharma, M. J. Vrhel, and H. J. Trussell. Color imaging for multimedia. *Proceedings of the IEEE*, 86(6):1088–1108, 1998.

[29] G. Sharma, W. Wu, and E. N. Dalal. The CIEDE2000 color-difference formula: implementation notes, supplementary test data, and mathematical observations. *Color Research & Application*, 30(1):21–30, 2005.

[30] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[31] P. Urban, M. Desch, K. Happel, and D. Spiehl. Recovering camera sensitivities using target-based reflectances captured under multiple led-illuminations. In *Proceedings of the Workshop on Color Image Processing*, page 916, 2010.

[32] J. van de Weijer, T. Gevers, and A. Gijsenij. Edge-based color constancy. *IEEE Trans. on Image Processing*, 16(9):2207–2214, 2007.

[33] M. Varma and A. Zisserman. Classifying images of materials: Achieving viewpoint and illumination independence. In *European Conf. on Computer Vision*, volume 3, pages 255–271, 2002.

[34] A. Vedaldi and S. Soatto. Quick shift and kernel methods for mode seeking. In *European Conference on Computer Vision*, pages 705–718, 2008.

[35] G. Wyszecki and W. Stiles. *Color Science: Concepts and Methods, Quantitative Data and Formulae*. Wiley, 2000.

[36] Y. Zhao and R. S. Berns. Image-based spectral reflectance reconstruction using the matrix R method. *Color Research & Application*, 32(5):343–351, 2007.