Chromatic Aberration Correction and Spectral Reconstruction from Colour Images

by

Bernard Llanos

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

We present an algorithm for simultaneously demosaicing digital images, and correcting chromatic aberration, that operates in a latent space of spectral bands. Light refraction by a camera lens system depends on the wavelength of the light, causing relative shifting, and blurring, between intensity patterns in different wavelengths on the image sensor. The effect on the image is called chromatic aberration, and appears as colour fringes around edges in the image, and blur.

Chromatic aberration depends not only on the camera's optical system, but also on the spectral characteristics of the light entering the camera. Previous works on calibrating chromatic aberration produce models of chromatic aberration that assume fixed discrepancies between image channels, an assumption that is only valid when the image channels capture narrow regions of the electromagnetic spectrum. When the camera has wideband channels, as is the case for conventional trichromatic (RGB) cameras, the aberration observed both within and between channels can only be accurately predicted given the spectral irradiance of the theoretical, aberration-free image.

We develop a physically-correct chromatic aberration calibration procedure for RGB cameras. Using bandpass-filtered light, we calibrate a model of chromatic aberration as an image distortion that is parameterized by both image position, and light wavelength. To correct chromatic aberration, we estimate a spectral image that corresponds to the RGB image by solving a global numerical optimization problem. We include our model of chromatic aberration

in the data-fitting term of the optimization problem that models the transformation from the spectral image to the captured RGB image. We also include regularization terms in our optimization problem to enforce smoothness in the output image. Whereas the captured RGB image is mosaiced, meaning that each pixel senses only one colour channel, our algorithm does not require a demosaicing preprocessing step to recover all colour channel intensities at each pixel. Therefore, we avoid introducing bias from demosaicing algorithms, which is important because chromatic aberration and demosaicing are known to interact.

Since we model within-channel chromatic aberration, our reconstructed images are sharper than those obtained by previous works on calibrated warping of colour channels. In contrast to explicit deblurring algorithms, our algorithm leaves defocus blur intact, separating it from chromatic aberration. We also avoid introducing artifacts, such as ringing, that are commonly produced by deblurring algorithms. Nevertheless, recovering spectral images from RGB images is an ill-posed problem, and this ill-posedness is the major limitation of our approach. We determined that our spectral images have higher accuracy than measurements made using a consumer-grade spectrometer. Still, we recommend further research on RGB-to-spectral reconstruction, especially in relation to chromatic aberration, which may serve as useful constraint.

Preface

This thesis is an original work by Bernard Llanos. Research for this thesis was supervised by Dr. Herb Yang at the University of Alberta. No part of this thesis has been previously published.

Acknowledgements

The research described in this thesis involved many people, and I am grateful for their assistance. I would like to thank my supervisor, Dr. Herb Yang, who patiently advised me even when I did not have a clear research direction, and who gave me freedom to find a project I was interested in. They helped me find the joy of discovery. I also thank Dr. Martin Jagersand, for providing me with similar freedom as their teaching assistant, and for teaching me many of the concepts and techniques used in my research. Dr. Pierre Boulanger has also benefitted my work by serving on my examining committee with Dr. Yang and Dr. Jagersand.

I am especially grateful to Steve Sutphen, from the Department of Computing Science, who built the physical apparatus for my experiments. They translated my requirements into a design that exceeded my expectations. Additionally, I am grateful to them for sharing their experience with me through interesting conversations.

Similarly, I have benefitted from the experience of numerous members of the Department of Computing Science. Dr. Yiming Qian introduced me to computer vision research and gave me practical advice for success. The other members of the research group were a continual source of ideas through their presentations of literature. They include: G M Mashrur E Elahi, Juehui Fan, Dong Huo, Dr. Zhao Pei, H M Ata-E-Rabbi, Katie Tran, and Dr. Shibai Yin.

Outside the Department, I am grateful to Dr. Jilu Feng and Dr. Benoit Rivard for providing reference spectrometry data for my experiments. I am grateful to the Faculty of Science, the Faculty of Graduate Studies and Research, and the Graduate Students' Association of the University of Alberta for financial support in the form of scholarships and travel grants.

Alberta Innovates, and the Natural Sciences and Engineering Research Council of Canada granted significant funding for my studies.

Contents

List of Tables

List of Figures

Glossary

Blind Deconvolution

The process of reversing the effect of a point spread function, while simultaneously estimating the point spread function. Blind deconvolution attempts to correct images for optical aberrations without any prior calibration of the point spread function.

Chromatic Aberration

The image artifacts produced by spectral dispersion. To correct chromatic aberration means to produce an image that would be formed if light of all wavelengths propagated through the optical system in the same way as light of a reference wavelength.

Chromatic aberration is often decomposed into two non-physically based components: Lateral (transverse) and longitudinal (axial) chromatic aberration. Lateral chromatic aberration is modelled as image distortion, whereas longitudinal chromatic aberration is the chromatic aberration that remains after lateral chromatic aberration has been corrected. Longitudinal chromatic aberration is often modelled as a change in blur with the wavelength of light.

Chromatic aberration is discussed in detail in Section 1.1.2.

Colour-Filter Array

The pattern of colour filters covering the sensor of a single-sensor colour camera. Colour-filter arrays allow an image sensor to imitate human colour vision by producing a multi-dimensional response to incident light, instead of a one-dimensional intensity measurement. The Bayer pattern [8] is a common colour-filter array pattern (Section 2.1).

Crosstalk

(Pixel/Sensor) crosstalk, also called Sensor Interpixel Correlation, encompasses all physical effects that cause neighbouring pixels to respond to a stimulus that should have been registered only at the current pixel [69]. Crosstalk appears as colour desaturation and blur in images [126].

Deconvolution

The process of reversing the effect of a point spread function. Deconvolution produces an image which is more similar to the ideal pinhole camera image of a scene than the input image produced by a real optical system. All real imaging systems have optical aberrations, which authors usually model as point spread functions.

Demosaicing

The process of converting a single-channel colour-filter array image to a full-colour image (Section 2.1)

Demosaicing Criterion (DMC)

Our heuristic criterion for selecting regularization weights, computed as the mean squared error with respect to the Green channel of the input image, demosaiced by bilinear interpolation (3.24)

Filter mosaic

The pattern of colour filters covering the sensor of a single-sensor colour camera.

Goodness of Fit coefficient (GOF)

The cosine of the angle between two vectors (4.6) used as a measure of error between spectral quantities such as spectral irradiances

Hyperspectral Image

An image recording intensities as a function of wavelengths of light — specifically an image sampled at finely-spaced wavelengths. Hyperspectral images are spectral images with high spectral resolution.

Mean Relative Absolute Error (MRAE)

Mean of the absolute deviations between true and estimated values, where each deviation is expressed as relative to the true value (Equations (4.3) and (4.10))

Mean Squared Error Criterion (MSEC)

Our reference criterion for selecting regularization weights, computed as the mean squared error with respect to the true image (3.23)

Metamer

A spectral irradiance which produces the same response in a set of colour channels of a camera, or the same colour perception in human colour vision, as a different spectral irradiance.

Our definition of metamers follows the definition of *sensor metamers* by Prasad and Wenhe [95]. Prasad and Wenhe [95] discuss other uses of the term.

Metamerism

The undesirable property of a light sensing system wherein it produces the same responses to different spectral irradiances

Minimum Distance Criterion (MDC)

A heuristic criterion proposed by Song *et al.* [108] for selecting regularization weights. Our version is provided in (3.22)

Monochromatic

The property of having approximately a single degree of freedom of spectral information. A monochromatic image is an image captured by a sensor having a single spectral sensitivity function. Monochromatic light is light with a narrow distribution about a single mean wavelength.

Multispectral Image

An image recording intensities as a function of wavelengths of light — specifically an image sampled at few wavelengths, or at a relatively low spectral resolution that is still higher than the spectral resolution of conventional colour (RGB) images.

Point Spread Function

The impulse response of an optical system. The image which would be formed by a single light source of infinitesimal size [112]. Point spread functions vary with position on the image plane [37, 58], and with the distance between the light source and the lens [112], among other factors.

RGB Image

An image formed by a trichromatic camera (after post-processing by demosaicing, if necessary). The acronym "RGB" stands for "Red, Green, and Blue". Red, green, and blue represent the three spectral response functions of the camera. In most trichromatic cameras, these are functions that have global maxima in the long (red), medium (green) and short (blue) wavelengths of visible light, respectively.

Root Mean Squared Error (RMSE)

Square root of the mean of the squared Euclidean distances between true and estimated values (Equations (4.4), (4.5), or (4.11), depending on the interpretation of a "value")

Spectral Reflectance

The reflectance, or bidirectional reflectance distribution function [92], of an object, measured as a function of the wavelength of light. Spectral reflectance is the constant of proportionality between the irradiance received by a surface from a particular incoming direction, and the reflected radiance emitted in a particular outgoing direction, as a function of wavelength. The dependency of spectral reflectance on the incoming and outgoing directions is usually ignored, because the context is usually that of a single light source illuminating a scene captured by a camera with a fixed viewpoint.

Spectral Irradiance

Light irradiance measured as a function of the wavelength of light. Iradiance is the energy per unit time received by an infinitesimal area from all directions, accounting for foreshortening [92]. Irradiance is the quantity measured by an image sensor, and is proportional to the radiance of the scene being imaged [48].

Spectral Dispersion

A dependency of light propagation on wavelength. Under the paradigm of geometric optics, spectral dispersion is the deviation between the paths of light rays having different wavelengths.

A dispersion region, in the context of an image, refers to the finite area that may receive light from the same point in the scene because of spectral dispersion.

We may refer to dispersion as a displacement between images formed by light of different wavelengths, using terms such as "dispersion vector".

We model dispersion in this work as displacements, although its effect is more general, and is only approximated in terms of displacement vectors. For futher discussion, refer to Section 6.2.

Spectral Image

An image recording intensities as a function of wavelengths of light, with a higher spectral resolution than colour images (*e.g.* RGB images). Spectral images may contain either spectral reflectance or spectral irradiance information.

Spectral Radiance

Light radiance measured as a function of the wavelength of light. Radiance is the energy per unit time passing through an infinitesimal area perpendicular to the propagation direction of interest, per unit solid angle [92].

Spectral Response

A function describing the variation in the constant of proportionality between the irradiance incident on the camera lens and the output of the image sensor (in a given colour channel) with the wavelength of light

Spectral Sensitivity

A function describing the variation in the constant of proportionality between the irradiance received by an image sensor and the output of the image sensor with the wavelength of light. The term "spectral sensitivity" is used to characterize the sensor itself, whereas the term "spectral response" describes the relationship between light incident on the camera and the sensor's output. The spectral response of a camera therefore depends on the lens, and all components along the optical path preceding the image sensor [72]. In contrast with Manakov [72], we consider the colour filters in a colour-filter array to be part of the sensor, therefore determining the spectral sensitivity functions of the sensor, not only the spectral response functions of the camera.

Spectral Reconstruction

The estimation of spectral irradiances or other high-spectral resolution data corresponding to the signals recorded by an imaging device. Spectral reconstruction attempts to recover the spectral characteristics of the light incident on the imaging device, despite the many-to-one mapping of spectral irradiances to device responses. For example, RGB-to-spectral reconstruction refers to estimating either spectral irradiances corresponding to the colour vectors of an RGB image, or to estimating the spectral reflectances of the objects in the scene being imaged.

Structural Similarity index (SSIM)

A similarity metric between two intensity images (monochromatic images), proposed by Wang *et al.* [123]

Trichromatic

The property of having three degrees of freedom of spectral information. A trichromatic image is an image captured by a camera having three spectral response functions, such as camera having a sensor with a colour-filter array composed of three different types of filters. (Normal)

human colour vision is trichromatic because the eye has three types of photoreceptor cells, and each type has a different spectral sensitivity.

Chapter 1

Introduction

With higher image sensor pixel densities, optical aberrations become more noticeable [107]. Of these, we argue that chromatic aberration is the most objectionable. The eye is accustomed to seeing objects at varying degrees of defocus as part of accommodation. Therefore, the eye can interpret image blur, the visual manifestation of most optical aberrations, as sub-optimal accommodation. Unless the blur is severe, the viewer perceives the same scene in the image as if the blur was absent. They assume that the blur is part of the imaging process, not the scene. Chromatic aberration produces false colours around image edges, something to which the visual system is not accustomed. The false colours in an image give the impression that either the image is not a direct perception, but has been recorded by a camera, or that the scene contains the false colours. Yet the eye has its own chromatic aberration [127], which must be perceived at a level beneath conscious awareness. Whereas chromatic aberration is usually considered to be a nuisance in lens design, in the visual system, in contrast, it may provide information about the spectral properties of the scene, as researched in the context of colour-blind animals [109].

We develop our thesis around the connection between chromatic aberration and spectral information. Chromatic aberration arises because light propagation through an imaging system depends on the wavelength of the light. Suppose the chromatic aberration of an imaging system is calibrated in terms of wavelengths. Then the captured image can be corrected for chromatic aber-

ration only if the intensity of the light as a function of wavelength, its *spectral radiance*, is known.

In conventional digital cameras, however, the spectral radiance of light reaching the image sensor is unknown. Pixels respond to light regardless of its wavelength, such that image sensors are only capable of generating monochromatic images. Colour images are made possible by placing filters that have non-uniform spectral transmittance over pixels. The Red, Green, and Blue colour channels of consumer colour cameras are implemented using filters with three different spectral transmittances. These filters are not selective in the wavelengths of light that they transmit, meaning that multiple spectral radiances will produce the same colour [86]. Such collections of spectral radiances are called *metamers*. Furthermore, in most colour cameras, each pixel is under only one type of filter, but neighbouring pixels are under different types of filters [8], producing a *filter mosaic*, or *colour-filter array*. Therefore, determining the colour of the image —the set of filtered light intensities at each pixel —is a challenging process, known as *demosaicing*.

We hypothesize that the ambiguity in the spectral radiances corresponding to the colour image is small enough to allow for adequate correction of chromatic aberration upon recovering an approximate spectral image. We note that there are constraints on the spectral image beyond the spectral sensitivities of the colour channels, such as the physical constraint that spectral intensities are non-negative, and the statistical constraint that neighbouring locations in natural scenes have similar spectral radiances. In addition, chromatic aberration itself may serve as a cue for spectral image recovery, much like its possible role in biological vision systems mentioned above. Therefore, we develop a new method for correcting chromatic aberration in colour images, in which we recover a spectral image that projects to the captured image according to a calibrated model of chromatic aberration, colour channel spectral sensitivity, and mosaicing. In other words, we perform simultaneous demosaicing and chromatic aberration correction through spectral reconstruction.

1.1 Image Formation

In this section, we review key concepts in the study of imaging systems, and explain the physical nature of chromatic aberration.

1.1.1 The Point spread function

In computer vision, cameras are most commonly modelled as pinhole cameras [112]. Assuming we want to study the scene being imaged, not the image itself, a pinhole camera is the ideal camera, because the image it produces can be modelled as a simple projection of the scene from three to two dimensions. From the image, we can directly measure geometric and radiometric properties of the scene.

Real cameras do not conform to the pinhole imaging model, because each point on the image sensor receives light from multiple points in the scene, and because the pixels of the image sensor have finite area. The relationship between a real image, and the image that an ideal pinhole camera would have produced, can be expressed using a *point spread function*. A point spread function is the image which would be formed of a dark scene containing a single light source of infinitesimal size [112]. The point spread function is often approximated by a two-dimensional Gaussian distribution of intensity, but for large amounts of blur, a more general model must be used [73]. Aside from their dependence on the camera's optical design, point spread functions vary with the angular position of the light source relative to the camera's optical axis [37, 58], and with the distance along the optical axis between the camera and the light source [112]. Calibrated point spread functions can be used to transform the image into an image which is more similar to the image from a pinhole camera. This process is called *deconvolution*, because each local region of the captured image is modelled as the convolution of the pinhole camera image with the point spread function for that region. Alternatively, without calibration, both the point spread function and the ideal image can be estimated simultaneously in *blind deconvolution* [112].

1.1.2 Chromatic Aberration

Spectral dispersion is the dependence of light propagation on its wavelength. In imaging systems, spectral dispersion causes chromatic aberration. Formally, *chromatic aberration* is the difference between an image and the image that would have been formed in the absence of spectral dispersion. This definition is problematic because dispersion relates different wavelengths of light. If an image was taken under monochromatic light, would the image exhibit chromatic aberration? We think of chromatic aberration as depending only on the imaging system, not on the lighting conditions, nor on the scene. Therefore, we can instead define chromatic aberration as the difference between an image and the image that would have been formed if light of all wavelengths had the same propagation as light of a chosen reference wavelength. According to this new definition of chromatic aberration, a monochromatic image formed by light with a wavelength other than the reference wavelength would exhibit chromatic aberration.

Chromatic aberration is usually classified as longitudinal (axial) or lateral (transverse): The former produces a wavelength-dependent defocus effect, whereas the latter shifts image position with wavelength [45]. This distinction is relevant to methods for correcting chromatic aberration, which usually target one effect more than the other. There is no physical basis for the distinction, because spectral dispersion gives rise to both lateral and longitudinal chromatic aberration. At first glance, correcting lateral chromatic aberration appears simpler, because it appears to require only warping the images formed under different wavelengths relative to each other. In contrast, correcting longitudinal chromatic aberration requires deblurring. In camera with wideband colour filters, however, lateral chromatic aberration also manifests as blur, because the sensor measures intensities across broad ranges of wavelengths. Only if the image sensor responded to narrowband portions of the spectrum could lateral chromatic aberration be corrected by image warping. For wideband colour filters, one can correct lateral chromatic aberration after calibrating per-colour channel point spread functions that incorporate both warping and

blurring [101]. Unfortunately, these point spread functions are only valid for the spectral radiance of the light used during calibration.

To develop some intuition about chromatic aberration, consider Figure 1.1, which shows light from a point light source passing through a lens. Spectral dispersion causes the angle of refraction of the light rays to depend on wavelength. Consequently, light focuses at different distances behind the lens (longitudinal chromatic aberration), and forms images with different magnifications (lateral chromatic aberration), as a function of its wavelength. The diagram represents a geometric optics idealization, and neglects optical effects such as diffraction. Furthermore, the optical designs of real camera lenses are normally much more complex. Regardless, the diagram correctly predicts that, for radially-symmetric lens systems, lateral chromatic aberration is a phenomenon with a strong radial dependence on image position. Lateral chromatic aberration is minimal near the principal point (the optical centre of the image) [75]. Light focused near the principal point originates from light sources along the optical axis of the lens, and propagates in a radially-symmetric pattern about the optical axis that makes the position of highest intensity in the image independent of wavelength. As a consequence of the spatial non-uniformity of chromatic aberration, chromatic aberration correction methods must tolerate aberration of varying severity within the same image.

In refractive imaging systems, spectral dispersion arises from a change in refractive index with wavelength. The dependence of refractive index on wavelength can be described by the Sellmeier equation [11]:

$$n\left(\lambda\right)^2 - 1 = a + \sum_k \frac{b_k}{\lambda^2 - \lambda_k^2} \tag{1.1}$$

where n is the refractive index, and a and b_k are empirical constants.

For SCHOTT N-BK7 glass, a common lens material, the Sellmeier equation is[1]

$$n\left(\lambda\right)^2 - 1 = \frac{1.03961212\lambda^2}{\lambda^2 - 0.00600069867} + \frac{0.231792344\lambda^2}{\lambda^2 - 0.0200179144} + \frac{1.01046945\lambda^2}{\lambda^2 - 103.560653} \tag{1.2}$$

[1] https://refractiveindex.info/?shelf=glass&book=BK7&page=SCHOTT

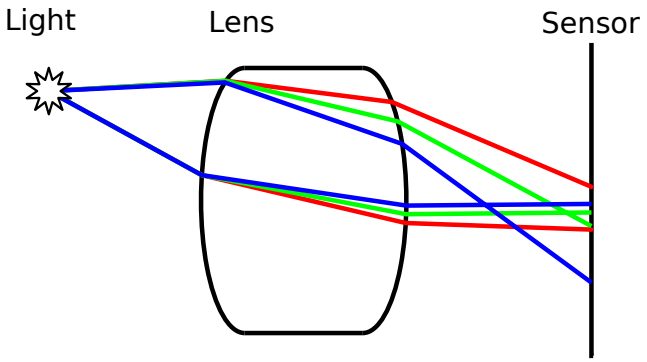


Figure 1.1: A simplified visualization of chromatic aberration from a single biconvex lens. A white point light source produces an image with a wavelength-dependent blur (longitudinal chromatic aberration) and a wavelength-dependent position (lateral chromatic aberration). The Red, Green, and Blue rays in the diagram represent long, medium, and short-wavelength components of the light's radiation. They do not represent colour channels, which do not correspond to single wavelengths of light.

We plot the refractive index of SCHOTT N-BK7 glass, according to Equation (1.2), in Figure 1.2. Figure 1.2 demonstrates that chromatic aberration likely has a smooth dependence on wavelength, but that this dependence is nonlinear. Whereas Figure 1.2 shows a monotonic function of wavelength, dispersion in compound lens systems is not necessarily monotonic with respect to wavelength [11]. We observed non-monotonicity of lateral chromatic aberration in our experiments (Figure B.4).

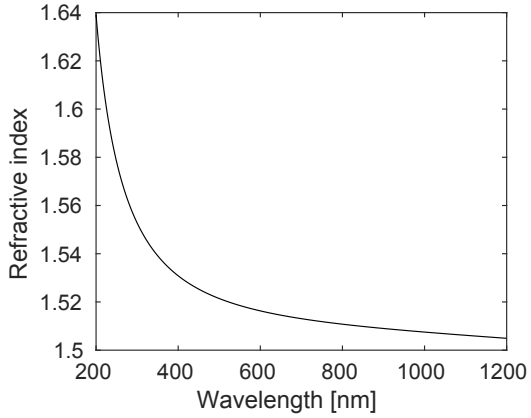


Figure 1.2: A plot of the refractive index of SCHOTT N-BK7 glass as a function of wavelength, based on the Sellmeier equation (1.2).

1.2 Contributions

Historically, lens designers were responsible for minimizing chromatic aberration, but performing the correction in software has the advantage of maintaining image quality while allowing for more compact, inexpensive lenses [42]. Furthermore, such an approach lets chromatic aberration be used for interesting applications such as depth from defocus [118, 117] or extended depth of field [21].

Yet lens designers work in terms of wavelengths of light, whereas colour image processing algorithms have been formulated in terms of colour channels. Each channel has internal chromatic aberration that varies with the spectra of light arriving from the scene. In this work, we assess whether working in the spectral domain can improve chromatic aberration correction in colour images, even though colour images represent highly lossy compressions of spectral information. We attempt to create, in software, a process analogous to the more physically-principled approach taken by lens designers.

We make the following contributions:

(i) We present the first study of RGB-to-spectral reconstruction directly from raw colour-filter array images for correcting chromatic aberration.

(ii) We make our method easy to use by basing it on physical constraints, such that it has a small number of interpretable parameters, and a simple calibration procedure. We do not require that the user collects large volumes of training data. Furthermore, we present criteria for automatically choosing parameter values for our method.

(iii) We demonstate improved correction of chromatic aberration relative to methods which operate in three-dimensional colour spaces, such as RGB. Our results show reduced colour fringes, and correction of within-colour channel blurring from lateral chromatic aberration. In contrast, monochromatic aberrations, such as defocus blur, which sometimes are deliberately used for visual effect, are preserved.

In the following chapters, we first review related works on image processing (Chapter 2), and then present our approach (Chapter 3). Subsequently, we discuss how we evaluated our approach in Chapter 4. We present the results of our evaluation in Chapter 5, and make some global remarks about our results in Chapter 6, which leads us to discuss possible directions for future work (Chapter 7). Finally, we summarize our findings in Chapter 8.

Chapter 2

Related Work

We draw inspiration from works in several areas, including demosaicing, spectral reconstruction, chromatic aberration correction, and image restoration. In this section, we survey research in these areas. We discuss works that we evaluated as part of our experiments in additional detail in Section 4.6.

2.1 Demosaicing

Demosaicing computes missing intensities in the colour channels of a colour-filter array image. In the camera we used for our experiments, the sensor has the pattern of colour filters shown in Figure 2.1.

This pattern (Figure 2.1) is known as a Bayer pattern, after its inventor [8]. In a Bayer pattern, there are twice as many pixels with Green colour filters as there are with Red or Blue colour filters. The increased sampling rate of the Green colour filters is justified by the human visual system's greater sensitivity to wavelengths in the middle of the visible spectrum [65]. The Green colour

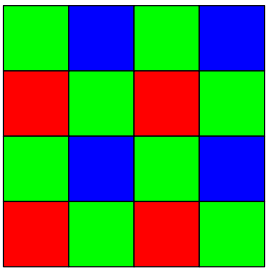


Figure 2.1: The Bayer colour-filter array pattern [8] used in our BlackFly Flea3 colour camera.

filters always cover the diagonal pixels of the 2×2 repeating pattern of colour filters. There are four variations of the pattern, distinguished by the colour filters that cover the top left and second from top left pixels of the image. If the pixels in the the 2×2 pattern units are ordered from left to right, top to bottom, the variations are: `GBRG`, `BGGR`, `RGGB`, and `GRBG`.

A colour-filter array is a more convenient solution for colour sensing, from a hardware and optics perspective, than the primary alternative, which was to have a separate image sensor for each colour channel [65]. Unfortunately, by using a colour-filter array, the image captured by the sensor is no longer directly interpretable by human viewers. Demosaicing is required to recover an interpretable, full-colour image.

At first, demosaicing was treated as an interpolation problem. Bilinear or bicubic interpolation were selected for efficiency [125]. Later solutions interpolated transformations of the colour channels, such as the differences between colour channels, or the ratios of different colour channels [59]. More recently, Kiku *et al.* [59] introduced residual interpolation, in which the other colour channels are initially demosaiced by guided upsampling [40], using a preliminary interpolation of the Green channel. The residuals of the filtering result are then interpolated, and are added to the filtering result to produce the new demosaiced channel.

The key insight motivating residual interpolation is that interpolation methods perform best when the samples to be interpreted have minimum Laplacian energy [59]. The Laplacian energy of an image is defined as:

$$E\left(\mathbf{I}\right) = \frac{1}{N} \sum_{(x,y)} \left(\Delta I_{(x,y)}\right)^2 \tag{2.1}$$

where $\Delta I_{(x,y)}$ is the image Laplacian evaluated at position (x, y), and N is the number of pixels in the image. Interpolation therefore performs well when the image is smooth, as interpolation enforces smoothness on the result.

Demosaicing can also use an explicit smoothness constraint, as done by Menon and Calvagno [77], who formulated a global optimization problem with a quadratic data-fitting term, and quadratic penalties on high-frequencies of each colour channel, and on the differences between the gradients of the colour

channels. While the cross-channel penalty terms are only reliable when colour channels have similar intensities, they have the advantage of allowing for a least-squares solution. Still, quadratic penalties tend to oversmooth strong edges, so Menon and Calvagno use the gradient of their initial solution to downweight their smoothing terms, in a subsequent round of global optimization that, for efficiency, recovers only the luminance of the image. Specifically, they lighten the smoothing penalties on image edges having a similar magnitude and direction as the edges in their initial solution, therefore performing anisotropic, spatially-adaptive regularization [77].

Tan *et al.* [115] present a different approach to adaptive regularization, by incorporating many different prior terms into their global optimization problem. The flexible Alternating Direction Method of Multipliers method [12], which we also use in our work, even allows them to incorporate priors that do not have closed-form expressions [115]. Obtaining a good result is then a matter of selecting appropriate global weights on the different prior terms, as opposed to local, adaptive weights. In their article, they do not explain how to select the optimal weights [115].

Unfortunately, all demosaicing methods that use constraints between colour channels assume that values in different colour channels are correlated. Correlation cannot be assumed for highly saturated colours, as highlighted by Wu and Zhang [126], who use a set of training images to learn a Bayesian-type classifier to distinguish saturated colours from more "pastoral" colours. Their demosaicing algorithm then adapts to the classification result at each pixel.

The approach of Wu and Zhang [126] combines ideas from both signal processing-based demosaicing methods, and fully data-driven demosaicing methods. The latter category of methods learn to produce demosaicing results directly based on training data. For example, Wu *et al.* [125] learn a regression model, based on the A+ superresolution method [116], that can improve the output of an initial demosaicing algorithm. More recently, other authors, such as Tan *et al.* [114], have used neural networks as their regression models. Data-driven methods are difficult to evaluate, unfortunately, because they cannot be trained on images from cameras with colour-filter arrays, yet it with these

cameras that they must be used.

While data-driven approaches to demosaicing can be adapted to arbitrary images, they cannot overcome artifacts which are present in the training images. One such artifact is crosstalk, wherein, for example, light passes through the colour filter covering one pixel to irradiate adjacent pixels for different colour channels [46, 96]. More generally, crosstalk, or *sensor interpixel correlation*, encompasses any physical effect which causes neighbouring pixels to respond to a stimulus that should have been registered exclusively at the current pixel [69]. Most demosaicing methods ignore crosstalk, including our own. Those which do account for crosstalk during demosaicing may be useful references for further improving the performance of our method on high-resolution cameras [126], and on cameras with multispectral colour-filter arrays [24].

Unfortunately, to our knowledge, Dijkstra *et al.* [24] presented the only straight-forward method for calibrating crosstalk. They had the advantage of using a multispectral camera, with known spectral sensitivity in each channel. In a multispectral camera, one can assume that crosstalk manifests as a deviation of the actual spectral sensitivities from idealized spectral sensitivities that are Gaussian functions with the same peak locations in the spectrum [24]. Demosaicing methods for RGB cameras have instead relied on known point spread functions representing crosstalk [46, 126], rather than attempting to calibrate crosstalk.

2.2 Chromatic aberration correction

Like crosstalk, chromatic aberration is also present in any set of training images for demosaicing methods. Demosaicing and chromatic aberration correction are intimately related problems, as shown in simulation by Wang *et al.* [122], such that there are works on solving the two problems jointly.

Schuler *et al.* [101] presented one of the earlier works on simultaneous demosaicing and chromatic aberration correction. Their method treats chromatic aberration correction as a deblurring problem, where the point spread function is both spatially-varying, and colour channel-specific. They express

the corrected image as the solution to a global optimization problem, which allows them to perform demosaicing implicitly, by including mosaicing in the data-fitting term of their optimization problem. While recovering the image takes hours, their optimization-based spatially-varying deconvolution algorithm gives better results than a faster Fourier-domain approximation [101]. A few later works have built on their approach by testing different regularization terms. Whereas Schuler *et al.* [101] used the YUV colour space to regularize chrominance more strongly than luminance, Heide *et al.* [42] developed a cross-channel prior to better model the relationships between the original colour channels:

$$\frac{\nabla_{xy} I_{j,(x,y)}}{I_{j,(x,y)}} \approx \frac{\nabla_{xy} I_{k,(x,y)}}{I_{k,(x,y)}} \tag{2.2}$$

where j and k are two different colour channels. Their prior has the advantage that it can be applied not only to colour channels in RGB images, but also to the spectral bands of multispectral images. On the other hand, (2.2) complicates the optimization algorithm because of its nonlinearity [42].

More efficient methods for performing both demosaicing and chromatic aberration calibrate forward transformations, and do not attempt to satisfy optimality criteria. Usually, they treat the two tasks as sequential operations. Lluis-Gomez and Edirisinghe [70] use block-matching to calibrate displacement vectors between the three colour channels for individual image patches. They then warp the raw colour channels of the image prior to demosaicing, noting that attempting to correct chromatic aberration after demosaicing will produce suboptimal results, because demosaicing has been affected by chromatic aberration [70]. To our knowledge, the state-of-the-art in calibrated chromatic aberration correction by image warping is the method of Rudakova and Monasse [99]. First, Rudakova and Monasse use a calibration pattern of black disks on a white background to obtain highly robust and accurate measurements of relative displacements between colour channels. We repurposed their calibration technique for spectral dispersion measurement, as described in Section 3.2.1. Next, Rudakova and Monasse fit bivariate polynomials to the disk displacements to model very general patterns of aberration across the

image [99]. Their global model is in contrast to the method of Lluis-Gomez and Edirisinghe [70], who lack a mechanism to reduce noise in their image block warp vectors. Many works preceding Rudakova and Monasse [99] did leverage global models of colour channel displacements, but typically relied on simple radial [75], or radial and tangential models [71], which Rudakova and Monasse demonstrate are suboptimal.

In our view, warping is equivalent to demosaicing, because warping requires interpolating image pixels, and interpolation reconstructs a continuous version of the image, where each colour channel is defined at all locations, not only at locations corresponding to its colour filters. Therefore, methods which correct chromatic aberration by warping raw image data are performing sub-optimal demosaicing, by failing to leverage correlations between colour channels. A promising approach that does not have this limitation was developed by Korneliussen and Hirakawa [60], who propose a chromatic aberration-tolerant demosaicing algorithm, and then correct the demosaiced image for chromatic aberration by warping its low-frequency components. Next, they update the high-frequency components of the Red and Blue channels based on those of the Green channel. Their method is derived from a Fourier domain analysis of chromatic aberration, and uses a luminance-chrominance colour space where the effect of chromatic aberration can be better isolated [60]. Unfortunately, none of the above efficient methods account for within-channel blur, or between-channel relative blur, caused by chromatic aberration. They use the Green channel as a reference for the Red and Blue channels, assuming that it is not affected by chromatic aberration.

In many applications, the input image for chromatic aberration correction is a full colour image, as the original raw image is no longer available. Moreover, the properties of the camera are often unknown, motivating researchers to propose methods which restore images solely based on prior assumptions of ideal image properties. For instance, Sun *et al.* [111] model images as having ideal colour channels related by second-order Taylor series expansions. They also assume that colour channels are related by spatially-varying blur, therefore accounting for longitudinal chromatic aberration. They formulate an alternat-

ing optimization algorithm that enforces their Taylor series approximation, and corrects relative blur between colour channels [111]. In comparison, most earlier works tended to optimize simple criteria, such as encouraging matching gradients between colour channels, and sharp edges (*e.g.* [18, 93]). Sing [107], in contrast, derived a more rigorous approach from optics and digital signal processing principles. They used a small number of parameters to model magnification change and (spatially-invariant) blur from chromatic aberration, and estimated values for these parameters from sharp edges in the image. Furthermore, they also corrected in-camera sharpening, which can amplify noise, and produce ringing artifacts in the image [107]. Still, their method is limited by its need for sharp edges in the image, and by its reliance on a model of chromatic aberration with few parameters. More generally, uncalibrated chromatic aberration correction methods cannot adapt to the peculiarities of individual cameras and lenses, and may produce spurious results if the scene being imaged contains colour features which resemble chromatic aberration artifacts.

While Korneliussen and Hirakawa [60], and other works, acknowledge that chromatic aberration is a spectral phenomenon which cannot be perfectly corrected in the RGB domain, it seems that chromatic aberration correction in the spectral domain has only been attempted in systems which capture spectral images directly. In a series of articles, J. Brauers and their collaborators studied filter wheel spectral cameras, which capture images through different narrowband optical filters to produce multispectral images. They first corrected distortion between spectral bands using an affine transformation calibrated using patch-wise image registration with mutual information as the image similarity criterion [15]. Concurrently, they also proposed correcting for relative blurring, using point spread functions calibrated between spectral bands [13]. Finally, in their later work, they improved on their mathematical models for image distortion, such as showing how conventional geometric camera calibration procedures, which return pinhole camera models and lens radial and tangential distortion parameters, are sufficient for correcting the distortion induced by the different filters in a filter wheel [14].

Among methods for correcting chromatic aberration, our work on spectral correction of chromatic aberration in colour images is related to that of Cossairt and Nayar [21], who used a lens with severe longitudinal chromatic aberration to obtain extended depth-of-field RGB (or monochromatic) images. While they examined point spread functions for light with different spectral characteristics, they did not attempt to use spectral information to correct chromatic aberration. Instead, they assumed white light when deblurring images [21]. We also calibrate chromatic aberration as a spectral phenomenon, but then we account for spectral variation throughout the image when correcting chromatic aberration.

2.3 Spectral Reconstruction

To use spectral information for correcting chromatic aberration, we must perform spectral reconstruction. A spectral image measures intensities at fine resolution in the spectral domain, whereas a conventional image sensor captures images at low spectral resolution, because it responds to light across a broad range of wavelengths. We define *spectral reconstruction* as the process of recovering high-resolution spectral information from the responses of conventional image sensors.

2.3.1 Physically-Based Approaches

Post-processing can recover high-spectral resolution images if optical elements preceding the image sensor convert spectral information into spatial or temporal information. The simplest spectral imaging systems use filters to isolate different portions of the electromagnetic spectrum. Filter wheel systems, as mentioned in Brauers and Aach [14], for example, capture series of images through different optical bandpass filters, therefore encoding spectral information using temporal information. Alternatively, spectral information can be encoded as spatial information, as when a multispectral filter array covers the sensor (*e.g.* the sixteen-channel camera mentioned in Dijkstra *et al.* [24]). Assuming the filters have narrow passbands, there is little ambiguity in the

wavelengths of the light recorded by the sensor in either design. Unfortunately, the spectral resolution of these systems is fixed, and is generally coarse. Furthermore, the limited temporal or spatial resolution of filter wheels or filter arrays, respectively, reduces image quality. Note that wavelength selection can also be applied to the light source (*e.g.* as in Goel *et al.* [36]), instead of the camera, but controlling the light source is only possible in lab environments.

More sophisticated spectral imaging systems rely on dispersion to provide a flexible tradeoff between spatial and spectral resolution. Coded aperture snapshot spectral imaging (CASSI) systems, for example, create a patchwork of spectral and spatial information on the sensor by masking light after it passes through a dispersive optical element [35, 66, 119, 121]. In optimization-based approaches to snapshot spectral imaging, a *modulation matrix*, relating the spectral image and the captured image, must be calibrated before spectral reconstruction can be performed [66]. The modulation matrix serves as the data-fitting term of the optimization problem, and represents imaging as a rank-deficient linear mapping. To compensate for the rank-deficient data-fitting term, Lin *et al.* [66] used a learned sparse representation of spectral images to constrain the output images. In contrast, Choi *et al.* [19] used a convolutional autoencoder model of spectral images, in addition to regularizing the gradient of the spectral image. As an alternative to optimization, which is computationally expensive [66], Wang *et al.* [121] moved the computation to an offline step by training a convolutional neural network for spectral reconstruction.

Of greater interest to us are spectral reconstruction methods also based on dispersion, but in a form more similar to chromatic aberration. Cao *et al.* [16] placed a triangular prism in front of a camera, and masked the light incident on the prism to create separated dispersed spectra on the image sensor. Their reconstruction process is therefore very simple, but they cannot achieve high spatial resolution. Subsequently, Baek *et al.* [7] omitted the mask, such that the spectra from adjacent points overlap on the camera sensor. They solved an optimization problem to recover the image, relying on isolated sharp edges in the image to provide good cues for spectral reconstruction. We adapt their

method by omitting the prism, such that chromatic dispersion in the image arises only from the camera's lens. After describing our method and its empirical behaviour, we elaborate on the differences between our work and theirs in Section 6.4.

2.3.2 Data-Driven

The primary disadvantage of the methods described in Section 2.3.1 is their reliance on specialized equipment. Therefore, a separate class of spectral reconstruction methods operates on the images taken by ordinary RGB cameras. At first glance, the problem seems unsolvable, because the space of spectral radiance is infinite-dimensional, whereas colour cameras have three dimensions of spectral sensitivity. Yet metamers, which are different spectra that produce the same RGB values, seem to be rare in natural images [33, 95].

Therefore, as investigated empirically by Jia *et al.* [53], one can reasonably assume that spectra lie on a three-dimensional manifold. Alternatively, as done by Arad and Ben-Shahar [4] and Wu *et al.* [124], one can learn a sparse dictionary of spectral signatures. Aside from explicit constraints on the mapping from RGB to spectral information, researchers have also implicitly learned well-behaved mappings for RGB-to-spectral reconstruction. A variety of learned representations have been tested, including radial basis function networks [84], Gaussian processes [44], and neural networks [3, 85, 104].

Data-driven spectral reconstruction methods are usually computationally-efficient, and cost-effective. The may not even require the spectral response functions of the camera as input (*e.g.* Shi *et al.* [104]). On the other hand, metamerism remains an obstacle of unknown importance, especially given that the performance of data-driven methods outside of spectral image datasets is poorly-characterized. We revisit these concerns later in Section 3.3.2.

Chapter 3

Image Reconstruction

3.1 Overview

We strive to find a spectral image which corresponds to an observed colour-filter array (raw) image. Using a *maximum a posteriori* approach to image estimation, we seek to minimize an objective function of the form:

$$\mathbf{I}^* = \underset{\mathbf{I}}{\operatorname{argmin}} \|\mathbf{M}\left(\mathbf{\Omega}\left(\mathbf{\Phi}\left(\mathbf{B}\left(\mathbf{I}\right)\right)\right)\right) - \mathbf{I}_{\text{raw}}\|_2^2 + \sum_{j=1}^{N} w_j \mathbf{S}_j\left(\mathbf{I}\right), \quad \mathbf{I} \geq 0 \tag{3.1}$$

Graphically, problem (3.1) is summarized in Figure 3.1. Our method is conceptually simple, as it consists of a single optimization problem. Each term in problem (3.1) encapsulates a substantial amount of information, as we will describe in detail in this chapter.

The latent spectral image, $\mathbf{B}\left(\mathbf{I}\right)$, is a hypothetical image that would be formed by a spectral camera with a lens free from the lateral chromatic aberration of the real camera's lens that we have calibrated. $\mathbf{B}$ represents a change of basis, allowing us to estimate the spectral image in a finite-dimensional space of control bands, as opposed to the infinite dimensional space of wavelengths. From the spectral image, we can obtain a colour image, also corrected for lateral chromatic aberration.

The first term of (3.1) approximates the image formation process that results in the observed raw image: $\mathbf{\Phi}$ is an operator describing the lateral chromatic aberration of the real camera lens. $\mathbf{\Omega}$ is an operator that converts the spectral image to intensities in the colour channels of the camera, according

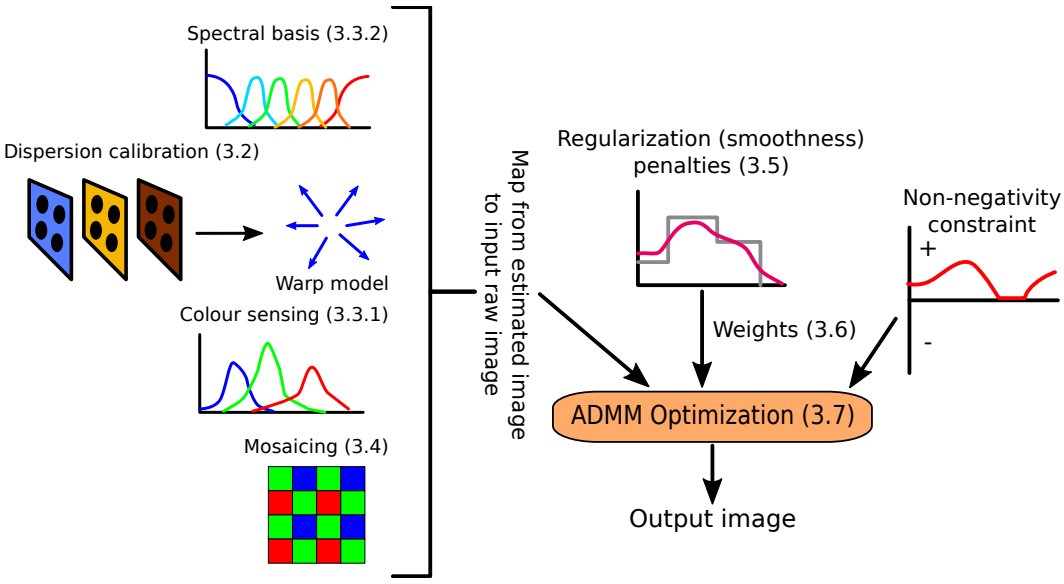


Figure 3.1: A visual overview of our image reconstruction method, showing the elements of our image reconstruction optimization problem (3.1). Numbers in parentheses refer to section numbers in this chapter.

to the camera's known spectral response functions. Lastly, $\mathbf{M}$ is an operator which forms the raw image by subsampling the full colour image according to the camera sensor's colour-filter array.

The summation in (3.1) represents a weighted sum of prior terms, which we also refer to as regularization operators, because we have limited our experiments to smoothness priors. In contrast to our model of image formation, which is derived from principles of optics and digital imaging, we selected our prior terms by empirical validation. To do so, we experimented with a range of priors, as described in Chapter 5. We tested different gradient operators, different norms applied to the gradients, and different methods for selecting the weights (w_j) multiplying the prior terms.

The final element of the optimization problem (3.1) is a non-negativity constraint, which follows from the nature of light as power. Adding a constraint makes solving the optimization problem more challenging, however, and we evaluate in Section 5.2.2 whether it is warranted.

3.1.1 Terminology

We use the term "spectral" to refer to quantities which vary as a function of the wavelength of light, and the term "spectral domain" to refer to the space of visible wavelengths of light. In contrast, we will use the term "frequency" to refer to the rate of oscillation of a signal, regardless of whether we are considering the signal as a function of space (*e.g.* in the image plane) or as a function of wavelength. We will avoid using the term "frequency" to refer to the rate of oscillation of light intensity with respect to time, and will avoid using the term "spectral" in relation to Fourier analysis.

3.2 Modelling Dispersion

There are two key challenges we face in modelling how image formation changes with wavelength. First, calibrating a model may require specialized equipment, and existing datasets normally do not provide data that could be used for calibration. Second, we need to choose a model that can be calibrated with sufficient sensitivity to capture small differences between images without being sensitive to noise. We based our method on the work of Rudakova and Monasse [99], and therefore model only lateral chromatic aberration —spatial warping between the images formed under different wavelengths of light. We describe our method for modelling lateral chromatic aberration in this section, but later elaborate on our rationale for selecting lateral chromatic aberration as the form of wavelength dependence to measure, in Section 6.2.

3.2.1 Keypoint Localization

To calibrate the slight shifts between the images in different colour channels, Rudakova and Monasse measure the displacements of black disks in a printed calibration pattern of black disks on a white background. They use the Levenberg-Marquardt algorithm to find disk positions in each colour channel with high precision, and robustness to noise and blur. Next, they pair the disk positions in the Green channel with those in the Red and Blue channels based on proximity, to obtain a sparse map of lateral chromatic aberration

vectors [99].

We follow their approach, but as we are interested in lateral chromatic aberration as a spectral phenomenon, we find displacements between disk keypoints in images taken under different bandpass-filtered illuminations. Since we can assume there is no aberration between colour channels in these images, we can therefore use all channels together to find each disk's position. We compute keypoint displacements between colour channels, in images taken under unfiltered illumination, only for comparison with our models of spectral dispersion.

We detect disk keypoints in an image taken of a disk pattern by first computing grey level thresholds with Otsu's method [88], applied separately to each colour channel. By fusing the thresholded versions of the colour channels, we obtain a binary image, which, after postprocessing with morphological opening and closing operations, provides connected components of white pixels which are candidate disks. We filter the candidate disks to those which are within two standard deviations from the mean area, to remove spurious detected disks.

To refine a disk keypoint's position, we minimize the sum of squared differences in intensity between a synthetic disk image, and the image pixels in the neighbourhood of the keypoint. Our synthetic disk image is constructed from the following parameters, which we optimize using the Levenberg-Marquardt algorithm:

Lightnesses: A set of intensity values for the interior and exterior of the disk, for each colour channel. We initialize the interior and exterior lightnesses from the modes of the image intensities in the detected disk and non-disk regions, respectively.

Axis lengths: The lengths of the major and minor semi-axes of the ellipse, initialized by fitting an initial ellipse to the connected component in the binary image

Orientation: The angle of the ellipse's major semi-axis with respect to the image x-axis

Centre: The coordinates of the ellipse centre

Edge width: The slope of the assumed linear intensity transition between the interior and exterior of the ellipse. We use an initial value determined empirically by observing the sharpness of edges in the image.

After obtaining refined disk centres, we build a sparse map of spectral dispersion by finding the displacements between mutually nearest-neighbour centres, paired between the image for the reference bandpass-filtered illumination, and the images for each of the other illuminations. Alternatively, to create a map of lateral chromatic aberration between colour channels, we fit refined disk keypoints to individual colour channels within an image, and then pair keypoints between the Green channel and the other colour channels simply by associating the keypoints corresponding to the same disk. Note that disk keypoints are more robustly estimated in bandpass-filtered images than between colour channels in images taken under unfiltered illumination, because smaller numbers of pixels are used to refine each keypoint in the latter case.

3.2.2 Vignetting correction

The disk keypoint model described in Section 3.2.1 assumes that each disk has a uniform intensity within its perimeter, and that the intensity outside of its perimeter is also uniform. Vignetting in the camera lens system, and non-uniformity in the illumination, may violate this assumption. To test whether these phenomenon invalidate our disk keypoints, we corrected images for intensity variation using bivariate polynomial models of intensity, and observed a slight improvement in the symmetry of the resulting dispersion pattern (Figure B.1). Therefore, we correct intensity variation as a preprocessing step before fitting disk keypoints to the image, although doing so is usually not necessary.

Our vignetting correction method takes as input a binary image indicating a region of the image which should have uniform intensity. We extract pixels from the Green channel of the raw image within this region, and use them to fit a bivariate polynomial model of image intensity, I, in image x and y

coordinates:

$$\hat{I}_{(x,y)} = \sum_{i=0}^{p} \sum_{j=0}^{p} a_{ij} x^i y^j \tag{3.2}$$

To find the polynomial coefficients, $\{a_{ij}\}$, we minimize the squared error,

$$\{a_{ij}\} = \underset{\{a_{ij}\}}{\operatorname{argmin}} \sum_{(x,y)} \left\| I_{\text{raw, }(x,y)} - \sum_{i=0}^{p} \sum_{j=0}^{p} a_{ij} x^i y^j \right\|_2^2 \tag{3.3}$$

which is an overdetermined linear least squares problem. We select the degree of the polynomial, p, from among the set of values $\{0, 1, \ldots, 5\}$, by tenfold cross-validation [39].

For numerical stability, we must normalize the image coordinates [99]. Given the set of image pixel coordinates from the user-marked region, $\{(x_k, y_k)\}$, we compute an affine homography, T_{xy}, a 3×3 transformation matrix, which centers the coordinates such that they have zero mean, and gives them variances of unity along each of their principal component directions. We also normalize the image intensities $\left\{I_{\text{raw, }(x,y)}\right\}$ to have zero mean and unit variance, using a 2×2 affine transformation matrix T_I. As we solve for the polynomial coefficients given the normalized image intensities and normalized spatial coordinates, our polynomial model must be evaluated on image coordinates subject to the transformation T_{xy}, and then its output must be transformed by T_I^{-1}. For simpler presentation, we have omitted these transformations from the equations in this section.

To correct the image for vignetting and non-uniform illumination, we divide the captured image by its polynomial approximation:

$$I_{\text{corrected, }(x,y)} = \frac{I_{\text{raw, }(x,y)}}{\sum_{i=0}^{p} \sum_{j=0}^{p} a_{ij} x^i y^j} \tag{3.4}$$

Note that we apply the same correction model to all channels of the raw image. For images captured under bandpass-filtered illumination, we can assume that the colour channels are approximately related by scaling factors given by the relative sensitivities of the colour channels to wavelengths within the filter passband, as discussed further in Section 4.2.4. Consequently, the same intensity correction can be applied to all colour channels, and our per-channel

disk lightness parameters described in Section 3.2.1 will accommodate for the scaling factors between colour channels. In contrast, for images captured under unfiltered illumination, our use of a single correction model for all colour channels rests on the additional assumption that the intensity non-uniformity is spectrally-invariant.

3.2.3 Image registration

Rudakova and Monasse proposed their disk keypoints as a method to find shifts between the colour channels of an image with high precision [99]. The disk keypoints are described by a parametric model that accommodates affine distortion between the calibration pattern and the captured images, therefore allowing for ellipses in the image, instead of circles. Affine distortion is only an approximation to the geometric distortion induced by the lens, and lens aberrations that cause spatially-varying blur might further degrade the accuracy of keypoint localization under the affine distortion model. Therefore, we sought to evaluate disk keypoint detection. Rudakova and Monasse did so only in simulated images, as they could compare their results with the synthetic ground truth keypoint positions. We choose to perform the assessment in real images, where ground truth positions are not available. Consequently, we performed a relative assessment, by comparing the disk keypoints with an alternative method for calibrating displacements between image channels, or between images captured under unfiltered illumination. The results of our assessment are provided in Section B.2.

Our alternative keypoint generation method is based on the work of Brauers *et al.* [15], who used mutual information as an image similarity criterion for patch-wise registration of the images taken under the different filters of a filter-wheel multispectral camera. We divide the image plane into patches, and then compute patch-specific translations between images taken under different bandpass-filtered illuminations or between image colour channels (depending on whether we want to calibrate dispersion in the spectral domain, or the colour domain). We use the image registration routines from MATLAB's Image Processing Toolbox [74] to estimate the translations. Specifically, we

selected mutual information as the similarity criterion [76, 97], and an evolutionary optimization method [110] to find the translations maximizing the similarity criterion. The result is a grid of keypoints that quantify dispersion across the image plane.

Dispersion keypoint generation by image registration has the advantage of being able to process any images with sufficient texture, in contrast to disk keypoint localization, which requires images of calibration patterns. The primary disadvantage of image registration, in our application, is the need to demosaic the input images. While images can be registered by sampling a sparse set of pixels to compute the similarity metric during optimization, in order to estimate the registration transformation with subpixel accuracy, pixels must be interpolated, which is a form of demosaicing. We try to minimize errors from demosaicing when registering images taken under different filtered illuminations by only registering the Green channels of the images. We select the Green channel because it has the highest sampling density in conventional colour-filter arrays. We chose bilinear interpolation for demosaicing the Green channel because filtered illumination may violate the inter-channel correlation assumptions built into more sophisticated demosaicing algorithms.

3.2.4 Dispersion Interpolation

From the sparse map of dispersion, constructed as described in Sections 3.2.1 or 3.2.3, we create a map of dispersion at arbitrary locations by smoothed interpolation, on the assumption that dispersion varies smoothly throughout the image. Furthermore, we also interpolate dispersion in the spectral domain, which is justified by the smooth change in the indices of refraction of lens materials with wavelength (Section 1.1.2).

We tested two interpolation methods: The first is based on the bivariate polynomials used by Rudakova and Monasse [99]. Their approach is also the inspiration for our vignetting correction method (Section 3.2.2). The difference between their procedure and our vignetting correction procedure is that they use separate polynomial models for each of the x and y components of the dispersion vectors between the Green and Red channels, and between the

Green and Blue channels. We use their approach to model dispersion between colour channels in calibration images taken under unfiltered light, for comparison with our spectral models of dispersion.

For our spectral models of dispersion, we fit trivariate polynomials in x, y, and λ (wavelength), instead of computing bivariate polynomials in x and y. The trivariate polynomials model the dispersion measured between the images for each bandpass-filtered illumination and those for the reference filtered illumination. As a consequence of their dependence on wavelength, our polynomial models are globally optimized over all filtered illuminations.

The procedure we use to fit the trivariate polynomials is analogous to the one we use for bivariate polynomials, but differs slightly in how we preprocess the data used for fitting. Specifically, we normalize the λ values of the image pixels, given by the center wavelengths of the filter passbands, to have zero mean and unit variance. We compute the wavelength normalization transformation separately from the normalization transformation for the image (x, y) coordinates. Note that, when fitting either bivariate or trivariate polynomial coefficients, we use x and y-components of dispersion vectors that have been normalized jointly, not separately.

To select the degrees of the fitted polynomials, whether bivariate or trivariate, we use cross-validation. For bivariate polynomials, we use tenfold cross-validation, as done for vignetting correction (Section 3.2.2) [39], but we choose the degree from a larger set of possible values, $\{0, 1, \ldots, 12\}$. For trivariate polynomials, we allow for a different degree in x and y from the degree in λ. We evaluate the cross-validation error over all possible values of the two degrees (d_{xy}, d_λ) from the set of pairs $\{0, 1, \ldots, 12\} \times \{0, 1, \ldots, N_\lambda - 2\}$. The upper limit of $N_\lambda - 2$, where N_λ is the number of bandpass filters, is necessary because all of our datapoints are sampled at only N_λ different wavelengths, so we cannot fit polynomials of degree in wavelength higher than $N_\lambda - 1$. The discrete set of values for λ also prevents us from applying tenfold cross-validation to select the degree of the polynomial in λ, as we will naively choose a degree of $N_\lambda - 1$. Such a degree is too high, and results in instability of the polynomial between the center wavelengths of the filter passbands, but will be selected

because there are no datapoints at intermediate wavelengths at which to measure cross-validation error. Therefore, we use the modified cross-validation algorithm given in Algorithm 1.

Algorithm 1 partitions the dispersion keypoints into training and validation sets first by illumination centre wavelength, then by random selection among keypoints. Consequently, each candidate polynomial model is evaluated on keypoints from an illumination on which it was not trained. All candidate models are trained on the illuminations with the smallest and largest passband centre wavelengths, however. If these illuminations were instead used for validation, the standard deviations of the cross-validation errors (s_{db} in Algorithm 1) would be inflated by the poor extrapolation behaviour of polynomials, and the "one-standard error" rule from Hastie *et al.* [39] would lose its utility.

In comparison with our use of cross-validation to select the degrees of polynomial interpolants, Rudakova and Monasse selected a polynomial degree of 11 in both x and y. They state that the error should stabilize for degrees 7 to 11 [99], but do not explain what error they are referring to.

The second interpolation method we tested is thin-plate splines. For modelling dispersion between colour channels, we use two-dimensional thin-plate splines in the image (x, y) coordinates, whereas for modelling dispersion between the images for narrowband regions of the spectrum, we use three-dimensional thin-plate splines. (The three dimensions are the image coordinates and wavelength, λ.) We compute one thin-plate spline for each of the x and y-components of the dispersion vectors. For numerical stability, we apply the same normalization transformations to the disk keypoints data that we used prior to fitting polynomials.

The theory of thin-plate splines is presented by Wahba [120], for instance, whereas we used David Eberly's Geometric Tools library as the reference for our implementation [26]. A regularized thin-plate spline modelling the $j \in \{x, y\}$ component of dispersion vectors minimizes the following cost func-

Algorithm 1 Trivariate polynomial degree selection for dispersion interpolation

Require: A maximum spatial degree, D_{xy}

1: $D_\lambda \leftarrow N_\lambda - 2$ ▷ N_λ is the number of filtered illuminations
2: **for** $d \leftarrow 0, D_{xy}$ **do**
3: **for** $b \leftarrow 0, D_\lambda$ **do**
4: **for** $i \leftarrow 2, N_\lambda - 1$ **do**
5: Create a set, V_i, of datapoints for the i-th illumination
6: Create a set, T_i, of the remaining datapoints
7: **for** $j \leftarrow 1, 10$ **do**
8: Select a tenth of the points in T_i and unite them with V_i to form the set V_{ij}
9: Select the rest of the points in T_i to form the set T_{ij}
10: Train a trivariate polynomial model of dispersion with a degree of d, in x and y, and a degree of b, in λ, on the data in T_{ij}
11: $e_{ij} \leftarrow$ prediction error of the model on the data in V_{ij}
12: **end for**
13: **end for**
14: $e_{db} \leftarrow \frac{1}{10(N_\lambda - 2)} \sum_{i=2}^{N_\lambda - 1} \sum_{j=1}^{10} e_{ij}$ ▷ Sample mean
15: $s_{db} \leftarrow \sqrt{\frac{1}{10(N_\lambda - 2) - 1} \sum_{i=2}^{N_\lambda - 1} \sum_{j=1}^{10} (e_{ij} - e_{db})^2}$ ▷ Sample standard deviation
16: **end for**
17: **end for**
18: $e_{\min} \leftarrow min\left(\{e_{db}\}\right)$.
19: $s \leftarrow$ element of $\{s_{db}\}$ corresponding to $e_{\min}$
20: Set the degree in λ to the smallest value of b such that $e_{db} - e_{\min} \leq s$
21: Set the degree in x and y to the smallest value of d such that $e_{db} - e_{\min} \leq s$, where b is the degree in λ chosen in Line 20

Figure 3.2: Cross-validation for selecting the degrees in the image x and y-coordinates, and in wavelength, λ, of trivariate polynomial models of dispersion. Rather than choosing the degrees minimizing the cross-validation error, we use the "one-standard error" rule mentioned in Hastie *et al.* [39].

tion [120]:

$$C\left(f_j\right) = \sum_k \left\| f\left(\mathbf{p}_k\right) - d_{jk} \right\|_2^2 + w \int \left\| D^2 f \right\|_2^2 \tag{3.5}$$

where d_{jk} denotes the $j = x$ or $j = y$ component of dispersion for the k-th disk keypoint, and $D^2 f$ denotes the matrix of second-order derivatives of f with respect to its vector argument, $\mathbf{p}$. The regularization penalty, $\int \|D^2 f\|_2^2$, has a null space that, in our problem, is spanned by linear functions [120]. These linear functions describe the behaviour of f_j far from the sample points $\{\mathbf{p}_k\}$, whereas Green's functions, radial functions erected around each sample point, describe the behaviour of f_j close to the sample points. Therefore, with thin-plate splines, we can generate models of dispersion which are well-behaved outside the region of the image plane in which the disk keypoints lie. In contrast, polynomial models of dispersion tend to vary rapidly with position outside of the region occupied by the disk keypoints.

In practice, if we can cover the image plane with disk keypoints, polynomials are suitable for interpolating dispersion. While they give inaccurate results near the image borders, they have the advantages relative to thin-plate splines of being more computationally efficient to fit and to evaluate. We compare the two approaches for smoothly interpolating dispersion on experimental data in Section B.3.

As with the degree of the polynomial models of dispersion, we must select the thin-plate spline smoothing parameter, w in (3.5). We choose w which minimizes the generalized cross-validation objective [120], according to the procedure given in Algorithm 2. Algorithm 2 is a grid search (using a one-dimensional grid), inspired by the method of Song *et al.* for selecting regularization parameters [108]. We start with a grid of 15 values, rather than using 4 values at every iteration of the grid search, in order to avoid getting stuck in local minima of the generalized cross-validation objective function.

A final clarification of our dispersion interpolation methods concerns the image coordinates used to fit our parametric models of dispersion: We can use our models of dispersion in two ways: To correct images for dispersion by image warping, or as constraints for solving image estimation problems (3.1).

Algorithm 2 Thin-plate spline smoothing parameter selection

$a \leftarrow$ machine epsilon (*e.g.* 2.2×10^{-16}) ▷ A lower bound on w
$b \leftarrow 10^{10}$ ▷ An upper bound on w
Generate logarithmically-spaced values $w_1, w_2, \ldots, w_{15}$ from $w_1 = a$ to $w_{15} = b$
Evaluate the generalized cross-validation objective, $V(w)$ [120] at $w_2, w_3, \ldots, w_{14}$
Find the index k of the value w_k minimizing the objective
$a \leftarrow w_{k-1}$
$b \leftarrow w_{k+1}$
$w^* \leftarrow w_k$
$e_{\text{prev}} \leftarrow V(w_k)$
$s \leftarrow 2$ ▷ Iteration counter
loop
    Generate logarithmically-spaced values $w_1 = a, w_2, w_3, w_4 = b$
    Evaluate $V(w)$ at w_2 and w_3
    Find the index k of the value w_k minimizing the objective
    $e_{\text{current}} \leftarrow V(w_k)$
    if $e_{\text{current}} \leq e_{\text{prev}}$ **then**
        $w^* \leftarrow w_k$
        if $\frac{|e_{\text{current}} - e_{\text{prev}}|}{e_{\text{prev}}} < tolerance$ and $s \geq s_{\min}$ **then**
            return w^*
        end if
        $e_{\text{prev}} \leftarrow e_{\text{current}}$
    end if
    $s \leftarrow s + 1$
    if $s \geq s_{\max}$ **then**
        return w^*
    end if
    $a \leftarrow w_{k-1}$
    $b \leftarrow w_{k+1}$
end loop

Figure 3.3: Thin-plate spline smoothing parameter selection by generalized cross-validation [120], minimizing the generalized cross-validation objective, $V(w)$ [120], using a grid search strategy based on the method of Song *et al.* [108]

In the second case, dispersion projects the ideal image onto the dispersed (aberrated) image. In both cases, we need to express dispersion vectors as functions of positions in the images that are created by applying our dispersion models, as we will explain in Section 3.2.5. When using dispersion to correct images, we must express models of dispersion in terms of coordinates in the ideal image —the reference colour channel, or the image for the reference bandpass-filtered illumination. In contrast, when using dispersion for image estimation (3.1), we must express models of dispersion in terms of coordinates in the aberrated image —the non-reference colour channel or spectral band of the image captured by the camera.

In concrete terms, suppose we have a disk keypoint in the reference image or colour channel, with its centre at location $\mathbf{p}_i = (x_i, y_i)$. In the image for a different bandpass-filtered illumination, or in a different colour channel, the same disk keypoint has its centre at location $\mathbf{p}_j = (x_j, y_j)$. The dispersion between the two positions is $\mathbf{d}_{ij} = \mathbf{p}_j - \mathbf{p}_i$. For image warping, we want to model dispersion as depending on the position $\mathbf{p}_i$ and having a value of $\mathbf{d}_{ij}$. In contrast, for image estimation, we want to model dispersion as depending on the position $\mathbf{p}_j$ and having a value of $-\mathbf{d}_{ij}$.

3.2.5 Image Warping

Our models of dispersion across the image plane, described in Section 3.2.4, express dispersion as a continuous vector field. Such representations of dispersion are not directly usable for two reasons: First, they do not account for the discrete nature of digital images. Second, our image estimation algorithm is iterative (Section 3.7), and it is too time-consuming to sample complex dispersion functions each iteration.

We address both concerns by creating discrete representations of dispersion: matrices that map vectorized images to their dispersed or corrected versions. To construct a dispersion matrix operator, $\mathbf{\Phi}$, we iterate over all pixel locations in the output image, $\mathbf{I}_{\text{out}} = \mathbf{\Phi}\,\mathbf{I}_{\text{in}}$. For a given pixel in the output image, in a given colour channel or spectral band, we evaluate the dispersion function to find its corresponding position in the input image. As the corresponding

position is pair of real-valued coordinates, we use bilinear interpolation of the surrounding pixels to sample the input image at the position. Therefore, the elements of $\mathbf{\Phi}$ giving the value of the output pixel as a linear combination of the pixels of the input image are the weights computed for bilinear interpolation. Such an approach follows from related works on correcting images for chromatic aberration by image warping [14, 99].

While we described how our dispersion model accommodates for spatial discretization, the reader may wonder how we have treated spectral discretization. Our optimization problem (3.1) includes an operator, $\mathbf{B}$, converting spectra represented in an arbitrary basis to spectra represented using a discrete sampling of the visible spectrum. As we will discuss in Section 3.3.2, we presently use a representation for spectral images which is already very close to a discrete sampling of the visible spectrum. Therefore, we could use the same spectral sampling for our dispersion operators. Instead, however, we have chosen to define the spectral resolution of the dispersion operators partially-independently of the spectral sampling of the input image. Doing so allows us to change our spectral basis representation without reducing the quality of our dispersion operators.

Given a rectangular region of the image plane, for which we want to select a spectral resolution at which to sample dispersion, we divide the rectangle into quadrants. We evaluate the dispersion function at the centre of each quadrant, for the minimum and maximum wavelengths (λ_{min}, λ_{max}) at which the camera has a non-negligible sensitivity in any colour channel (*e.g.* 5% of the peak sensitivity in the same colour channel). We avoid sampling dispersion at the centre of the rectanglular region, or at its edges, because dispersion tends to be minuscule at the centre of an image, and because our models of dispersion are less accurate towards the edges of the image. We take the mean length, l, of the eight sampled dispersion vectors (one vector for each of λ_{min} and λ_{max} for each quadrant) to be the scale of dispersion for this region. Next, we calculate the number, n, of dispersion sampling wavelengths as the maximum of the number of spectral basis vectors represented in $\mathbf{B}$ and the value $\lceil \frac{l}{0.1} \rceil + 1$. (0.1 is the value we have chosen to control the spectral resolution, as described below.)

When generating our dispersion matrix operator, $\mathbf{\Phi}$, we will sample dispersion for the n equally-spaced wavelengths in the interval from the minimum to the maximum wavelengths used above ($\lambda_{\min}$, $\lambda_{\max}$).

Consequently, we will always sample dispersion such that the shift between consecutive spectral bands at which dispersion is sampled is approximately no greater than 0.1 pixels on average. We chose this limit of 0.1 pixels as it was determined by Rudakova and Monasse to be the lower limit at which dispersion is perceivable [99]. Also, by ensuring that our spectral sampling of dispersion is at least the dimension of the spectral basis represented by $\mathbf{B}$, we will not lose information by spectrally-undersampling the image when applying $\mathbf{\Phi}$.

3.3 Image Representation

We introduced the problem of image sampling in Section 3.2.5 with respect to evaluating models of dispersion. As digital cameras capture discretized images, we must convert our estimated images to discrete samples in order to evaluate the data-fitting term in problem (3.1). To take full advantage of the image sensor's pixel resolution, we have chosen to use the same spatial sampling (*i.e.* pixel representation) for our estimated images as the captured images. In the following sections, we explain our choice of spectral sampling, after first providing some background information.

3.3.1 Converting Spectra to Colour

If we assume the camera sensor has a linear response to light irradiance, and that there is no crosstalk between pixels, then the spectral irradiance, $S_{(x,y)}(\lambda)$, of light reaching the pixel at image coordinates (x, y), determines the pixel's value as follows:

$$I_{(x,y,c)} = \int_{\lambda} G_{(x,y)}(\lambda)\, Q_c(\lambda)\, S_{(x,y)}(\lambda)\ \mathrm{d}\lambda,\ c \in \{\text{Red, Green, Blue}\} \tag{3.6}$$

In (3.6), we assume that the spectral irradiance of the light is independent of time. $G_{(x,y)}(\lambda)$ is a term which encapsulates the ISO sensitivity setting of the camera, the exposure time, the size of the lens aperture, and the dependence

of light attenuation on the position of the pixel. We assume $G_{(x,y)}(\lambda) = 1$. $Q_c(\lambda)$ represents the spectral response of the camera, which accounts for the spectral transmittance of the lens, the colour filter for colour channel c, and the quantum efficiency of the sensor. (The latter two quantities are referred to as the "spectral sensitivity" of the camera, which is independent of the lens system [72].) Equation (3.6) is only an approximation —For a more sophisticated model of raw image responses, see Qiu and Xu [96], for example. In practice, we express the integral in (3.6) as a summation over the discrete spectral bands at which $Q_c(\lambda)$ and $S_{(x,y)}(\lambda)$ have been sampled, and use the trapezoid rule for numerical integration.

Accurate estimation of the camera's colour channel spectral response functions, $Q_c(\lambda)$ for $c \in \{\text{Red, Green, Blue}\}$, requires professional optical equipment [30]. Methods for camera spectral calibration using lower-cost equipment usually still require knowing the spectral power distribution of the light entering the camera [30, 96], and so are still impractical. In constrast, methods that can accommodate unknown light spectral power distributions rely on strong assumptions, such as that the camera spectral sensitivities are well-described by an empirical basis [54], or that they can be approximated by linear combinations of the CIE tristimulus functions (known as the Luther condition) [94]. We show the CIE tristimulus functions in Figure 3.4.

In our work, we have used three sets of spectral response or spectral sensitivity functions. The first are the CIE 1931 Standard 2° Observer spectral tristimulus functions (Figure 3.4), approximating the colour vision of a human observer. We use these functions to convert spectral power distributions to the CIE 1931 XYZ colour space, by following the ASTM E308 standard [6]. From the CIE 1931 XYZ colour space, we can produce colours in the sRGB colour space that are suitable for visual inspection when sent to a display device. For the details of colour space conversions, we refer the reader to the online resource created by Bruce Lindbloom[1].

We use a second set of spectral sensitivity functions for our simulated experiments. In our simulated experiments, we simulate raw colour-filter array

[1] http://brucelindbloom.com/

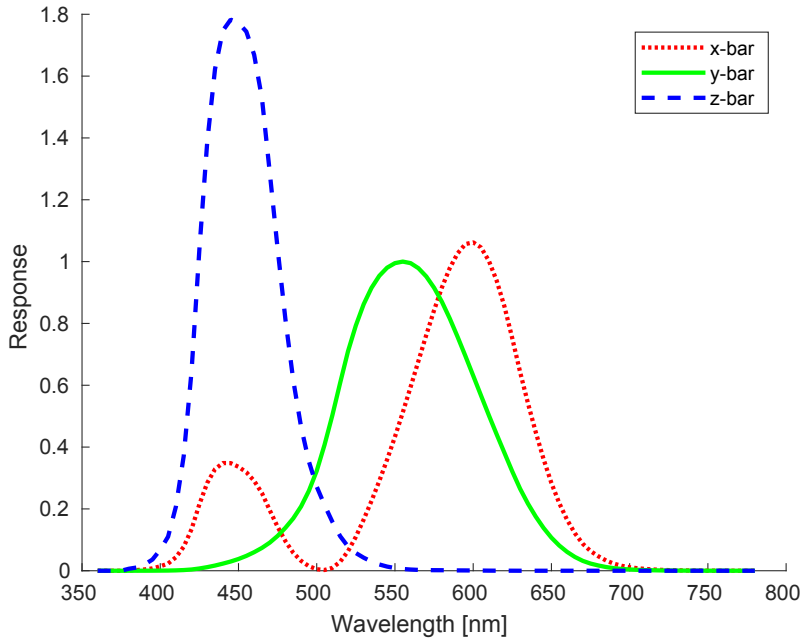


Figure 3.4: The CIE 1931 Standard 2° Observer spectral tristimulus functions, used to convert spectral power distributions to the CIE 1931 XYZ colour space

images from spectral images. We selected the spectral sensitivities of the Nikon D5100 consumer DSLR camera (Figure 3.5), measured by Darrodi *et al.* [22], because their small dataset of camera spectral sensitivities is both recent and accurate.

Finally, in our real experiments, we use spectral sensitivities measured by FLIR for the Sony ICX655, 2/3" image sensor (Figure 3.6) [32], as our Point Grey BlackFly Flea3 camera uses this sensor. While spectral sensitivities may vary across individual cameras, and while the data from FLIR does not account for our lenses, we believe that it is still a more accurate characterization of our camera than we could obtain using our available equipment. We justify our use of FLIR's data in Figure 4.12.

3.3.2 Spectral Image Representation

Having described all of the scene-independent quantities in (3.6), we now focus on the spectral irradiance of light reaching the pixel, $S_{(x,y)}(\lambda)$. In works on data-driven reconstruction of spectral images from RGB images (*e.g.* Nguyen *et al.* [84]), spectral irradiance is sometimes decomposed into the product of

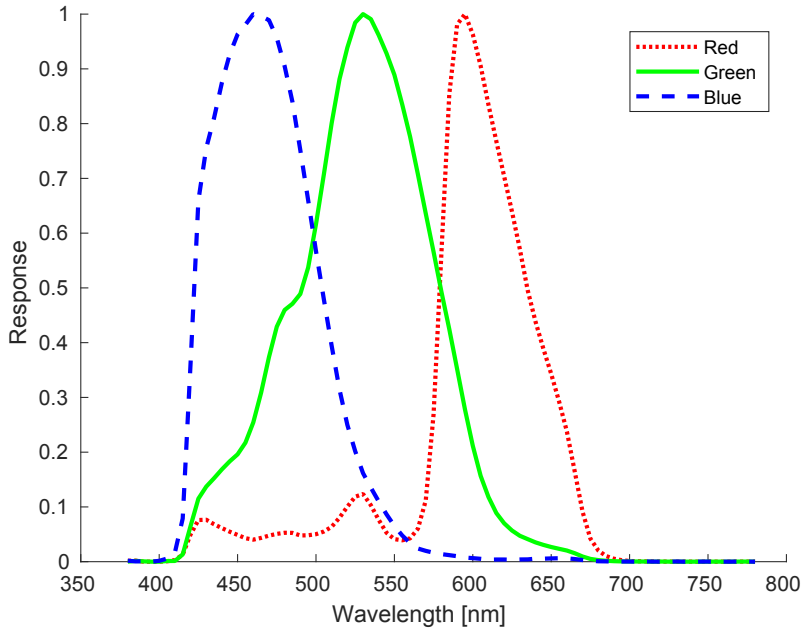


Figure 3.5: The spectral sensitivities of a Nikon D5100 camera, measured by Darrodi *et al.* [22]

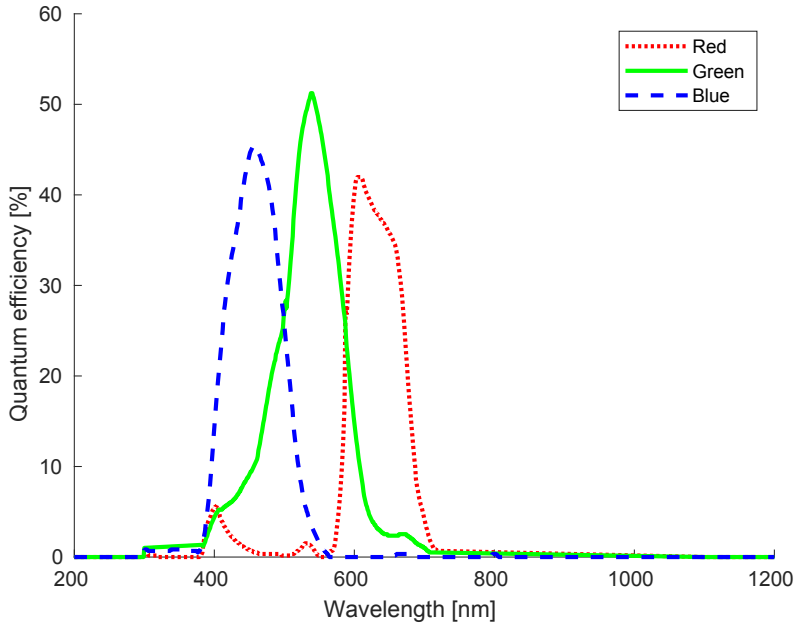


Figure 3.6: The spectral sensitivities of a Sony ICX655, 2/3" image sensor, measured by FLIR [32]

the spectral reflectance $R_{(x,y)}(\lambda)$ of the material being viewed, and the spectral power distribution of an illuminant $L(\lambda)$:

$$S_{(x,y)}(\lambda) = R_{(x,y)}(\lambda) L(\lambda) \tag{3.7}$$

One reason for doing so is that datasets may provide effective reflectance images. *Effective reflectance images* are obtained from spectral irradiance images by dividing the spectral signal at each pixel by the spectral signal of a reference object in the image, and then multiplying by the known reflectance of the reference object [33]. Effective reflectance images are approximately illuminantion-invariant.

The decomposition in (3.7) may seem problematic. For one, the illumination may not be constant over the scene. For another, spectral reflectance is only indirectly related to the captured image. The advantage of the decomposition in (3.7) is that separately estimating the two quantities may be simpler than directly recovering the spectral irradiance: Illumination estimation from RGB images is a well-studied problem [57], whereas the reflectances of real objects are well-approximated by a low-dimensional manifold [53], and so can be approximated using a small set of basis spectra. In practice, many data-driven spectral reconstruction methods neglect the illuminant and learn illuminant-specific mappings from RGB images to spectral irradiance images, where the spectral irradiance images are part of a dataset generated using a spectral camera, and the RGB images were simulated from the spectral irradiance images [3, 2, 4, 104]. The limited variety of illuminants used in the datasets [4, 17, 19, 84, 128] may be critical to the success of such techniques. For instance, Jia *et al.* [53] concentrate on images taken under outdoor illumination, and acknowledge that the complexities of indoor illumination may violate their assumption of low-dimensional spectra.

We choose to express spectral images in terms of spectral irradiances, rather than factoring spectral irradiance into reflectance and illumination (3.7). While spectral reflectances may allow us to leverage a useful low-dimensional basis for accurate spectral reconstruction, we do not want the accuracy of our method to be limited by the relevance of the training data used to create such

a basis. Moreover, we are unable to collect high-quality spectral images to experiment with training data of our own. We also cannot adequately test our method on datasets of spectral images because they do not provide images that are ideal for calibrating dispersion.

Nevertheless, working with spectral irradiance still leaves open the question of a basis representation, because the visible spectrum is a continuous space, whereas our optimization problem (3.1) recovers discretized images. An obvious choice of spectral basis is simply to sample spectra at evenly-spaced wavelengths. For instance, Baek *et al.* [7] use 23 bands, spaced 10 nm apart, but do not explain how they selected this number of bands. We also use an evenly-spaced sampling of the visible spectrum, but we propose criteria for selecting the appropriate number of bands.

The first criterion is to use only three bands, because the camera has three colour channels, and so our optimization problem (3.1) would be well-posed if we use strong regularization terms that properly demosaic the image. As we show in Section 5.4.3, however, three bands are insufficient for modelling the full range of colours in an image.

Second, we can use a set of known spectral measurements to find a set of bands which optimizes the spectral reconstruction error. While this approach has empirical validity, it may be expensive to obtain high-quality reference measurements. Furthermore, the evaluation will focus on spectral reconstruction accuracy in textureless regions of the image, because equipment for measuring spectral power distributions, such as point spectrometers, average light from non-negligible areas. Dispersion is only visible in textured regions, so the spectral reconstruction accuracy in textured regions may be different from that in textureless regions.

Third, and last, we can use the spectral response functions of the camera to find a reasonable spectral sampling for the estimated image. The spectral response functions of any trichromatic camera are approximately bandlimited: They can be accurately approximated using a small number of samples relative to the sampling resolution at which they were measured. For example, Figure 3.7 shows that the spectral sensivitities of the Nikon D5100 camera [22]

satisfy this assumption. For the Nikon D5100 spectral sensitivities, if we define their bandlimit as containing 95% of their signal power, then we need a sampling period of 35.5 nm in the visible spectrum to capture all of the spectral information that falls within this bandlimit. The number of samples needed is further reduced by the limited region of high sensitivity of the camera —We do not need to sample spectral images at wavelengths to which the camera is insensitive. In practice, we restrict spectral samples to the interval of the visible spectrum where any colour channel responds with at least 5% of its peak sensitivity. Combining both the domain and frequency constraints, for the Nikon D5100 spectral sensitivities, we use a set of 8 equally-spaced spectral bands in the range from 418 to 667 nm to represent our spectral images.

3.3.3 Spectral irradiance computation

From the discrete, constant-spacing sampling representation of our spectral images, we can derive the change of basis operator, $\mathbf{B}$, in problem (3.1) after choosing an interpolation kernel, K. The spectral irradiance at a wavelength of λ, given the basis coefficients, $\{v_i\}$, $i \in \{1, 2, \ldots, b\}$, of the estimated spectral image, is:

$$S_{(x,y)}(\lambda) = \sum_{i=1}^{b} K(v_i - \lambda) I_{(x,y,v_i)} \tag{3.8}$$

Usually, we are only interested in sampling the spectral irradiance at a finite set of equally-spaced wavelengths, $\{s_j\}$, $j \in \{1, 2, \ldots, N\}$. Furthermore, our interpolation kernel is always a function with even symmetry (*i.e.* $K(-x) = K(x)$). Therefore, we can erect the interpolation kernel around each wavelength s_j in the sampled representation of the spectral irradiance:

$$S_{(x,y,s_j)} = \sum_{i=1}^{b} K(s_j - v_i) I_{(x,y,v_i)} \tag{3.9}$$

From (3.9), we obtain $\mathbf{B}$:

$$\mathbf{B} = \begin{bmatrix} K(s_1 - v_1) & K(s_1 - v_2) & \cdots & K(s_1 - v_b) \\ K(s_2 - v_1) & K(s_2 - v_2) & \cdots & K(s_2 - v_b) \\ \vdots & \vdots & & \vdots \\ K(s_N - v_1) & K(s_N - v_2) & \cdots & K(s_N - v_b) \end{bmatrix} \tag{3.10}$$

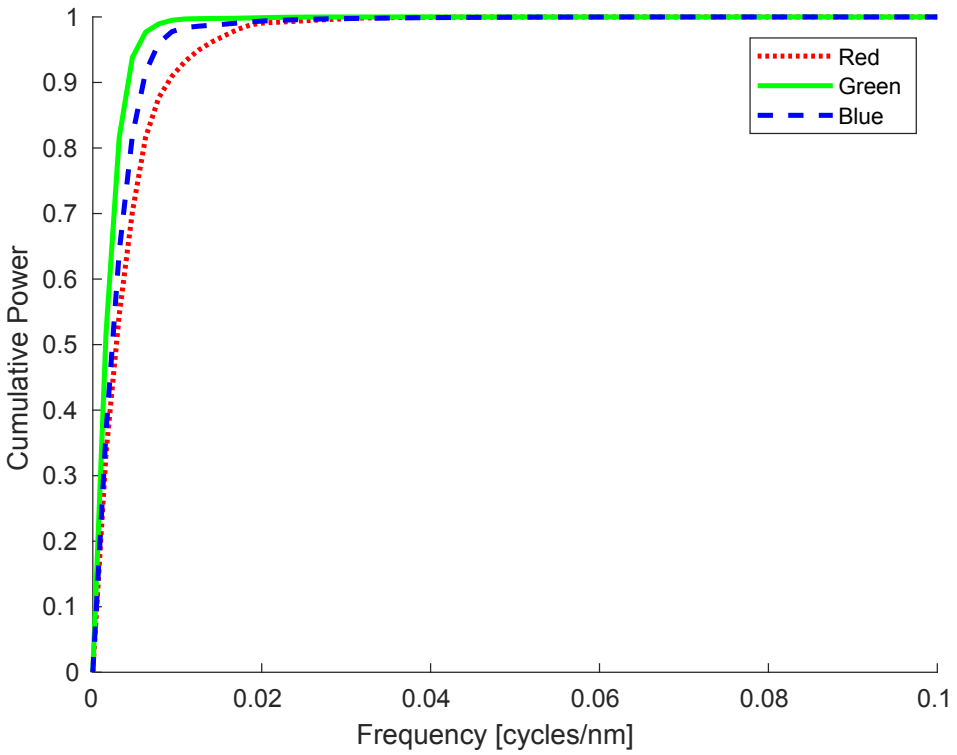


Figure 3.7: Cumulative power distributions of the spectral sensitivities of a Nikon D5100 camera. The cumulative power distributions are the cumulative sums of the squared moduli of the discrete Fourier transforms of the spectral sensitivities. We exclude the power at the zero frequency from the sum. The right edge of the plot is the Nyquist frequency corresponding to the 5 nm sampling period of the sensor spectral sensitivities. Note that the cumulative power distributions in the figure are expressed relative to their peak values. (The spectral sensitivities of the camera were measured by Darrodi *et al.* [22], and are shown in Figure 3.5.)

If we were to use the sinc function, $\text{sinc}\,(x) = \frac{\sin(\pi x)}{\pi x}$, as the kernel K, then K would perform ideal bandlimited interpolation [37]. In the case where the output spectral irradiance has a lower sampling frequency than our spectral image, we compute $\mathbf{B}$ as the product of two interpolation matrix operators:

$$\mathbf{B} = \mathbf{B}_{b\to N}\mathbf{B}_{b\to b} \tag{3.11}$$

where $\mathbf{B}_{b\to b}$ will interpolate the image to the same sampling frequency, and is needed in case $K\,(0) \neq 1$. $\mathbf{B}_{b\to N}$ is a downsampling operator that uses the sinc function as the interpolation kernel to remove frequencies above the Nyquist limit of the sampling space $\{s_j\}\,,\ j \in \{1, 2, \ldots, N\}$, therefore preventing aliasing artifacts in the result.

Aside from spectral downsampling (3.11), we do not use the sinc function as an interpolation kernel because it may produce negative values between sampling locations, violating the non-negativity constraint in problem (3.1). Instead, we chose a Gaussian kernel, because it produces smooth interpolated spectral irradiances, while preserving non-negativity:

$$K\,(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \tag{3.12}$$

We selected the σ parameter of the Gaussian kernel $K\,(x)$ such that 95% of the power of the kernel is carried by frequencies within the Nyquist frequency of the estimated spectral image. In other words, we match the bandlimit of the Gaussian kernel to that of the estimated spectral image.

As discussed in Section 3.3.2, our estimated images are sampled at locations which lie within the portion of the visible spectrum where the camera spectral sensitivities are non-negligible. Real spectral signals are defined over the entire electromagnetic spectrum, however, and we need to extrapolate to points slightly outside of our sampling domain to convert estimated images to colour, or to compare them with reference spectral data. We calculate $\mathbf{B}$ as though our spectral images are padded with large numbers of samples at shorter and longer wavelengths. These short-wavelength and long-wavelength samples have values equal to the values of the images at the shortest and

longest wavelengths of their actual sampling domains, respectively. This approach gives our estimated images approximately constant extrapolation conditions, which is more physically plausible than using values of zero outside of the sampling domain, for example.

3.4 Mosaicing

Mosaicing refers to the subsampling of colour information in a conventional single-sensor colour camera, in which each pixel senses light filtered according to a single colour channel. Most colour cameras use a Bayer pattern of colour filters [8], which is a repeating pattern of 2×2 squares of pixel filters having one Red colour filter, one Blue colour filter, and two Green colour filters (Section 2.1).

We implement mosaicing, $\mathbf{M}$ in problem (3.1), as a sparse matrix with elements equal to one where the corresponding pixel measures the corresponding colour channel. Mosaicing introduces effects which we presently do not account for, however. In particular, the colour filters affect crosstalk between pixels [46, 96]. Our image formation model assumes that all pixels respond only to the light directly incident upon them, although crosstalk may have an important effect on our results, as we discuss in Section 4.2.4.

3.5 Regularization Penalties

Having described our image formation model, we have explained the data-fitting term in problem (3.1), and now change our focus to the second term, which is a weighted sum of prior terms:

$$\sum_{j=1}^{N} w_j \mathbf{S}_j \left(\mathbf{I} \right) \tag{3.13}$$

There is a wide variety of image priors proposed in literature, and some authors, such as Tan *et al.* [115] use many priors in an attempt to combine their strengths and compensate for their weaknesses. We selected image priors that were widely-used in literature, that do not directly rely on training data

Table 3.1: The summations of image prior terms that we tested in our image estimation experiments. Aside from varying the prior terms in our image estimation problem (3.1), for each combination of prior terms, we tested image estimation with, and without, a non-negativity constraint on the image.

Condition	Prior terms					
	Spatial gradient $\mathbf{S}_1(\mathbf{I}) = \|\nabla_{xy}\mathbf{I}\|_p^p$		Mixed gradient $\mathbf{S}_2(\mathbf{I}) = \|\nabla_\lambda \nabla_{xy}\mathbf{I}\|_p^p$		Spatial Laplacian $\mathbf{S}_3(\mathbf{I}) = \left\|\nabla_{xy}^2\mathbf{I}\right\|_p^p$	
	$p=1$	$p=2$	$p=1$	$p=2$	$p=1$	$p=2$
L1	✓					
L2		✓				
Lap1					✓	
Lap2						✓
L1L1	✓		✓			
L2L2		✓		✓		
L1SpatialLap2	✓					✓
L1SpectralLap2			✓			✓

(but that may be inspired from statistical studies of images), and that incur a reasonable computational cost during image estimation. We list the combinations of prior terms we tested in Table 3.1. We express our prior terms as sparse matrix operators applied to the vectorized form, $\mathbf{I}$, of the spectral image I.

The first prior, also used by Baek *et al.* [7] (with $p = 1$), penalizes the spatial gradient of the image:

$$\mathbf{S}_1(\mathbf{I}) = \|\nabla_{xy}\mathbf{I}\|_p^p \tag{3.14}$$

where the image spatial gradient is defined as

$$\nabla_{xy} I\big|_{(x,y,\lambda)} = \begin{bmatrix} I_{(x+1,y,\lambda)} - I_{(x,y,\lambda)} \\ I_{(x,y+1,\lambda)} - I_{(x,y,\lambda)} \end{bmatrix} \tag{3.15}$$

This penalty on the image spatial gradient is often used in image restoration, where $p = 2$ allows for a simple least squares optimization, whereas $p \leq 1$ enforces sparsity ($p = 1$ is a common choice) [64]. We have tested $p = 2$ (the L2 norm) and $p = 1$ (the L1 norm), although natural images are best modelled by $p \in [0.5, 0.8]$ [106]. Note that the L1 norm penalty is anisotropic, in contrast to the closely-related isotropic total variation norm proposed by

Rudin *et al.* [100],

$$\int_{(x,y)} \sqrt{\left(\frac{\partial I_{(x,y,\lambda)}}{\partial x}\right)^2 + \left(\frac{\partial I_{(x,y,\lambda)}}{\partial y}\right)^2} \tag{3.16}$$

We have tested the anisotropic L1 norm penalty, rather than the total variation norm, for closer comparison with Baek *et al.* [7].

The second prior, proposed by Baek *et al.* [7] (with $p = 1$), penalizes the spectral-spatial mixed gradient of the image, with the intention of aligning edges between spectral bands:

$$\mathbf{S}_2(\mathbf{I}) = \|\nabla_\lambda \nabla_{xy} \mathbf{I}\|_p^p \tag{3.17}$$

The image spectral gradient is defined by

$$\nabla_\lambda I\big|_{(x,y,\lambda)} = I_{(x,y,\lambda+\Delta\lambda)} - I_{(x,y,\lambda)} \tag{3.18}$$

with $\Delta\lambda$ equal to the spacing between adjacent spectral bands in the spectral image. As with the first prior, we tested both L2 ($p = 2$) and L1 ($p = 1$) norm regularization with the spatial-spectral prior.

Finally, the third prior we tested is a prior on the image Laplacian, and was used by Song *et al.* [108] (with $p = 2$):

$$\mathbf{S}_3(\mathbf{I}) = \left\|\nabla_{xy}^2 \mathbf{I}\right\|_p^p \tag{3.19}$$

The image Laplacian is a second-order derivative operator defined by the equation:

$$\nabla_{xy}^2 I\big|_{(x,y,\lambda)} = 4I_{(x,y,\lambda)} - I_{(x+1,y,\lambda)} - I_{(x-1,y,\lambda)} - I_{(x,y+1,\lambda)} - I_{(x,y-1,\lambda)} \tag{3.20}$$

We tested both L2 ($p = 2$) and L1 ($p = 1$) norm penalties on the image Laplacian for comparison with the L2 and L1 versions of our other priors. The Laplacian is used in edge detection [113], and so can serve as an effective smoothness prior. Moreover, we have also selected it because it is related to demosaicing. As discussed by Kiku *et al.*, bilinear interpolation can perfectly demosaic images with Laplacian magnitudes of zero [59] (as mentioned in Section 2.1), suggesting that a penalty on the image Laplacian may favour some of the desirable properties that could be obtained by bilinear interpolation.

Many authors use penalties on the spectral gradients of images during spectral image reconstruction (*e.g.* [27, 108]). Spectral radiances originate from diverse phenomena, however, and many real-world spectra, such as fluorescent light spectral power distributions, or atomic emission spectra, are not smooth. We also note that spectral smoothness penalties would need to be stronger at wavelengths which typically carry less power, as variability at these wavelengths would be smaller on average, assuming the light incident on the sensor has Poisson noise characteristics [41]. Furthermore, integrating both spatial and spectral regularization terms into the same cost function is challenging because spatial roughness and spectral roughness are incomparable in magnitude. In light of these issues, we have used only prior terms which operate primarily on spatial information. We have avoided explicit spectral regularization, but have implicitly enforced spectral smoothness by representing our spectral images with a limited number of spectral bands (Section 3.3.2).

Note that, as shown in Table 3.1, we have not tested the spectral-spatial gradient prior (3.17) except in combination with the other prior terms. Preliminary experiments showed that this prior cannot demosaic the image on its own, such as shown in Figure 3.8.

3.6 Weights on Regularization Penalties

Selecting the values of the weights w_j in (3.1) is challenging. The true spectral image is unknown, so it is not possible to choose weights which result in the most accurate estimate of the spectral image. Even if ground truth data was available for a set of training images, the weights selected for the training images may not be optimal for images outside the training set. Finally, any procedure that tests different values for the weights can become intractable because the optimization problem (3.1) is high-dimensional.

It is not surprising that many authors working with similar optimization problems have given values for regularization weights without indicating how they chose the values [7, 19, 115]. In contrast, Fang *et al.* used a fixed-point algorithm to select their regularization parameter [27], based on a *balancing*

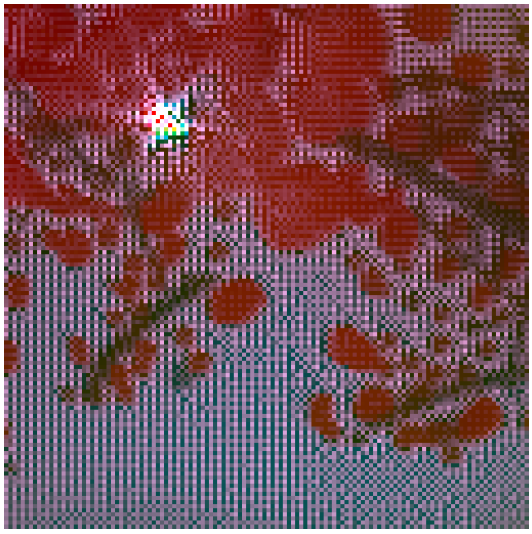


Figure 3.8: A sample result obtained with only a prior on the L1-norm of the spectral-spatial image gradient (3.17), and a non-negativity constraint. The high-frequency, gridded patterns of colours in images regions that should be smooth (*e.g.* the background, which should be a uniform white) indicate that the image is not fully demosaiced.

principle that constrains the data-fitting term to have a magnitude that is a constant multiple of the magnitude of the regularization penalty [20]. Unfortunately, the approach depends on meta-parameters which must be set manually, and is limited to a single regularization weight.

We follow the approach of Song *et al.* [108], who proposed an iterative grid search for one or more regularization weights. They demonstrated the effectiveness of their approach by comparing it against the L-curve method of Belge *et al.* [9] and generalized cross-validation [34], neither of which can be applied when the regularized optimization problem includes a non-negativity constraint. Song *et al.* use the Minimum Distance Criterion (MDC) to select regularization weights, which is defined in terms of the response surface. In our problem (3.1), the response surface is a surface of dimension N, defined by the following set of points in $N+1$-dimensional space:

$$\left\{\left(\left\|\mathbf{M}\left(\mathbf{\Omega}\left(\mathbf{\Phi}\left(\mathbf{B}\left(\mathbf{I}^{*}\right)\right)\right)\right)-\mathbf{I}_{\text{raw}}\right\|_{2}^{2},\mathbf{S}_{1}\left(\mathbf{I}^{*}\right),\mathbf{S}_{2}\left(\mathbf{I}^{*}\right),\ldots,\mathbf{S}_{N}\left(\mathbf{I}^{*}\right)\right)\right\} \tag{3.21}$$

where $\mathbf{I}^{*}$ is the solution to problem (3.1), and depends on the values assigned to the regularization weights w_j.

The MDC is the Euclidean distance between a point on the response surface and the *ideal point* where all terms in the optimization problem are simultaneously minimized. We find the first coordinate of the ideal point by setting all weights to very low values (the machine epsilon), and evaluating $\|\mathbf{M}\left(\mathbf{\Omega}\left(\mathbf{\Phi}\left(\mathbf{B}\left(\mathbf{I}^*\right)\right)\right)\right) - \mathbf{I}_{\text{raw}}\|_2^2$. Each subsequent coordinate, the $j+1$-th coordinate, $j \in [1, n]$, is the value of $\mathbf{S}_j\left(\mathbf{I}^*\right)$ obtained when all weights are set to very low values, but w_j is set to a very high value of 10^{10}.

We find the MDC as proposed by Song *et al.* [108] problematic, because it gives equal importance to all terms in the optimization problem. In our implementation, we have normalized the displacement from the ideal point by the approximate range of each coordinate:

$$MDC\left(\mathbf{I}^*\right) = \left(\frac{\|\mathbf{M}\left(\mathbf{\Omega}\left(\mathbf{\Phi}\left(\mathbf{B}\left(\mathbf{I}^*\right)\right)\right)\right) - \mathbf{I}_{\text{raw}}\|_2^2 - p_0}{\alpha_0}\right)^2 + \sum_{j=1}^{N}\left(\frac{\mathbf{S}_j\left(\mathbf{I}^*\right) - p_j}{\alpha_j}\right)^2 \tag{3.22}$$

In (3.22), $p_0, p_1, \ldots, p_N$ are the coordinates of the ideal point. α_0 is equal to $\|\mathbf{M}\left(\mathbf{\Omega}\left(\mathbf{\Phi}\left(\mathbf{B}\left(\mathbf{I}^*\right)\right)\right)\right) - \mathbf{I}_{\text{raw}}\|_2^2 - p_0$, where $\mathbf{I}^*$ is the image obtained with all weights set to the maximum value, 10^{10}. α_j, $j \in [1, N]$ is equal to $\mathbf{S}_j\left(\mathbf{I}^*\right) - p_j$, where $\mathbf{I}^*$ is the image obtained with all weights set to the minimum value (the machine epsilon).

Aside from our MDC, which in practice lead to oversmoothing (Section 5.2.4), we tested two other criteria for selecting regularization weights. The first is the mean squared error with respect to the true spectral image, the Mean Squared Error Criterion (MSEC) and therefore can only be used in simulations:

$$MSE\left(\mathbf{I}^*\right) = \frac{\sum_x \sum_y \sum_i \left(I_{(x,y,\lambda_i)} - \mathbf{B}\left(I^*\right)_{(x,y,\lambda_i)}\right)^2}{M} \tag{3.23}$$

where M is the product of the number of pixels and the number of spectral bands in the true spectral image.

The second additional criterion, which we label the Demosaicing Criterion (DMC), is the mean squared error with respect to a demosaiced version of the raw image, I_D:

$$DM\left(\mathbf{I}^*\right) = \frac{\sum_x \sum_y \left(I_{D,(x,y,\text{Green})} - \mathbf{\Omega}_{\text{Green}}\left(\mathbf{\Phi}\left(\mathbf{B}\left(I^*\right)\right)\right)_{(x,y)}\right)^2}{N} \tag{3.24}$$

where N is the number of pixels in the image. The DMC can be thought of as a cross-validation error, with the colour channel values missing from the colour-filter array used as the validation set. We evaluate the error of the Green channel only, because of its higher sampling density in a conventional Bayer colour-filter array.

With the DMC, we can set the smoothness of our estimated images based on the latest demosaicing algorithms. We are not constraining our estimated images to exactly match demosaicing results, however, and therefore we do not introduce artifacts from demosaicing algorithms into our estimated images. Presently, we use bilinear interpolation as the demosaicing algorithm, because it relies only on the assumption of spatial smoothness, and therefore is unlikely to fail catastrophically in peculiar conditions we would like to test, such as narrowband-filtered illumination.

We find values for the regularization weights that minimize a given criterion (MDC, MSEC, or DMC) using the grid search method presented in Algorithm 3. Algorithm 3 is based on the grid search method used by Song *et al.* [108]. Song *et al.* used 6 iterations, whereas we use a minimum number of 6 iterations, and a maximum number of 19 iterations, the latter number being set such that the relative error with respect to the ideal weights is at most 5%. In-between the minimum and maximum number of iterations, the algorithm terminates if the weight selection criterion improves by less than a convergence tolerance (we used a tolerance of 10^{-6}).

While we use Algorithm 3 for all of our weight selection criterion, the algorithm might not converge to the ideal weights for the MSEC and DMC, as these criteria may not be unimodal, depending on the image content. The MDC, in contrast, is unimodal [108]. In practice, we have observed that Algorithm 3 may select very low weights when using the MSEC and DMC, but the extreme values of these weights makes them easily identifiable. Running the algorithm on different image content is sufficient to obtain more reasonable weights. A statistical analysis of how frequently these failures arise is unfortunately very computationally-intensive, and the findings may not generalize to different cameras, nor to different image subjects.

Algorithm 3 Grid search for regularization weight selection

Require: A criterion $Cr(\mathbf{I})$ evaluated on the image $\mathbf{I}$ estimated under a given assignment of weights $w_j,\ j \in [1, N]$

1: **for** $j \leftarrow 1, N$ **do**
2: $a_j \leftarrow$ machine epsilon (*e.g.* 2.2×10^{-16}) ▷ A lower bound on w_j
3: $b_j \leftarrow 10^{10}$ ▷ An upper bound on w_j
4: **end for**
5: $e_{\text{prev}} \leftarrow \infty$
6: $s \leftarrow 1$ ▷ Iteration counter
7: **loop**
8: **for** $j \leftarrow 1, N$ **do**
9: Generate logarithmically-spaced values $w_{j1} = a_j, w_{j2}, w_{j3}, w_{j4} = b_j$
10: **end for**
11: Evaluate $Cr(\mathbf{I})$ for each tuple in the set $\{w_{12}, w_{13}\} \times \{w_{22}, w_{23}\} \times \cdots \times \{w_{N2}, w_{N3}\}$
12: $e_{\text{current}} \leftarrow$ minimum value of $Cr(\mathbf{I})$ encountered in Line 11
13: **if** $e_{\text{current}} \leq e_{\text{prev}}$ **then** ▷ The criterion improved
14: **for** $j \leftarrow 1, N$ **do**
15: $w_j^* \leftarrow w_{jk}$, where w_{jk} minimized $Cr(\mathbf{I})$
16: **end for**
17: **if** $\frac{|e_{\text{current}} - e_{\text{prev}}|}{e_{\text{prev}}} < tolerance$ and $s \geq s_{\min}$ **then**
18: **return** $w_1^*, w_2^*, \ldots, w_N^*$
19: **end if**
20: $e_{\text{prev}} \leftarrow e_{\text{current}}$
21: **end if**
22: $s \leftarrow s + 1$
23: **if** $s \geq s_{\max}$ **then**
24: **return** $w_1^*, w_2^*, \ldots, w_N^*$
25: **end if**
26: **for** $j \leftarrow 1, N$ **do**
27: $k \leftarrow$ index of the value w_{jk} minimizing $Cr(\mathbf{I})$ in Line 11
28: $a_j \leftarrow w_{j,k-1}$
29: $b_j \leftarrow w_{j,k+1}$
30: **end for**
31: **end loop**

Figure 3.9: The grid search strategy, based on the method of Song *et al.* [108], that we use to select the regularization weights w_j in our image estimation optimization problem (3.1). Note that this algorithm is similar to the algorithm we use for selecting the smoothing parameter of a thin-plate spline (Algorithm 2).

3.7 Optimization Algorithm

To solve problem (3.1) for the latent spectral image, $\mathbf{B}(\mathbf{I})$, we structure the problem according to the Alternating Direction Method of Multipliers (ADMM) framework described in Boyd *et al.* [12]. We note that Baek *et al.* [7] also used ADMM to solve their optimization problems. ADMM can easily accommodate our non-negativity constraint, and our L1 or L2 regularization penalties, facilitating experiments with different objective functions. Furthermore, all terms in our optimization problem, and the non-negativity constraint, satisfy the assumptions required for convergence, presented in Section 3.2 of Boyd *et al.* [12].

ADMM has a disadvantage of being slow to converge in practice [12], and it often fails to converge, to within our tolerance settings, within 1000 iterations. Poor convergence is more common as the number of L1-norm regularization penalties increases. Other optimization algorithms may be worth exploring in the future, to obtain better convergence. For now, while we are experimenting with different regularization penalties, we value the flexibility of ADMM more than fast convergence.

Alternating Direction Method of Multipliers (ADMM) is a useful framework for solving optimization problems by splitting them into pieces which depend on separate variables. A general ADMM problem has the form [12]:

$$\underset{\mathbf{x},\mathbf{z}}{\operatorname{argmin}} \, \mathrm{f}(\mathbf{x}) + \mathrm{g}(\mathbf{z}), \text{ such that } A\mathbf{x} + B\mathbf{z} = \mathbf{c} \tag{3.25}$$

The ADMM algorithm for solving such a problem is given as Algorithm 4.

In our problem, we have up to two penalty terms (Table 3.1). We also use a non-negativity constraint, which we can express as the following cost function:

$$\mathrm{v}(\mathbf{I}) = \begin{cases} \infty \text{ if } \exists I_i \in \mathbf{I} \; I_i < 0 \\ 0 \text{ if } \forall I_i \in \mathbf{I} \; I_i \geq 0 \end{cases} \tag{3.26}$$

$\mathrm{v}(\mathbf{I})$ is an indicator function that is zero if and only if all elements of $\mathbf{I}$ are non-negative, and an infinite positive value otherwise.

We can reformulate (3.25) for our problem as follows: First, we set $\mathbf{x}$ to be the estimated image, $\mathbf{x} = \mathbf{I}$. We set $\mathrm{f}(\mathbf{x})$ to be the sum of the data-fitting

Algorithm 4 The canonical Alternating Direction Method of Multipliers (ADMM) algorithm

Require: A positive penalty parameter, ρ

1: Initialize $k = 0$ and $\mathbf{u}^0 = \mathbf{0}$
2: Initialize $\mathbf{x}^0$ and $\mathbf{z}^0$ to starting guesses
3: **repeat**
4: $\quad \mathbf{x}^{k+1} \leftarrow \underset{\mathbf{x}}{\operatorname{argmin}} \, \mathrm{f}(\mathbf{x}) + \frac{\rho}{2} \left\| A\mathbf{x} + B\mathbf{z}^k - \mathbf{c} + \mathbf{u}^k \right\|_2^2$
5: $\quad \mathbf{z}^{k+1} \leftarrow \underset{\mathbf{z}}{\operatorname{argmin}} \, \mathrm{g}(\mathbf{z}) + \frac{\rho}{2} \left\| A\mathbf{x}^{k+1} + B\mathbf{z} - \mathbf{c} + \mathbf{u}^k \right\|_2^2$
6: $\quad \mathbf{u}^{k+1} \leftarrow \mathbf{u}^k + A\mathbf{x}^{k+1} + B\mathbf{z}^{k+1} - \mathbf{c}$
7: $\quad k \leftarrow k + 1$
8: **until** convergence

Figure 3.10: The scaled form of the ADMM algorithm for solving a problem of the form (3.25), given in Section 3.1.1 of [12]

term, $\|\mathbf{M}(\mathbf{\Omega}(\mathbf{\Phi}(\mathbf{B}(\mathbf{I})))) - \mathbf{I}_{\text{raw}}\|_2^2$, and any penalty terms subject to the L2 norm. Next, we set $\mathbf{z}$ to be a vector formed by stacking the arguments of the L1-norm in all L1-norm penalties and the image itself (which is the argument of v in (3.26)). Finally, we formulate the constraint $A\mathbf{x} + B\mathbf{z} = \mathbf{c}$ by setting A to be a stack of regularization operators corresponding to the L1-norm penalty terms, and the non-negativity constraint.

In the following discussion, we will present ADMM algorithms for two cases:

Case (i) L1-norm penalties on the spatial and spectral-spatial gradients of the image (L1L1 in Table 3.1), and a non-negativity constraint

Case (ii) L2-norm penalties on the spatial and spectral-spatial gradients of the image (L2L2 in Table 3.1), and a non-negativity constraint

For Case (i), $\mathbf{z} = \begin{bmatrix} \mathbf{z}_1 & \mathbf{z}_2 & \mathbf{z}_3 \end{bmatrix}^T = \begin{bmatrix} \nabla_{xy}\mathbf{I} & \nabla_\lambda \nabla_{xy}\mathbf{I} & \mathbf{I} \end{bmatrix}^T$, and $\mathrm{g}(\mathbf{z})$ is the sum $\mathrm{g}(\mathbf{z}) = w_1 \|\mathbf{z}_1\|_1 + w_2 \|\mathbf{z}_2\|_1 + \mathrm{v}(\mathbf{z}_3)$. The linear equality constraint in this case has $A = \begin{bmatrix} \nabla_{xy} & \nabla_\lambda \nabla_{xy} & \mathbb{I}_{M\times M} \end{bmatrix}^T$, where $\mathbb{I}$ is the identity matrix, and M is the number of elements in $\mathbf{I}$. $B = -\begin{bmatrix} \mathbb{I}_{M\times M} & \mathbb{I}_{M\times M} & \mathbb{I}_{M\times M} \end{bmatrix}^T$, and $\mathbf{c} = \mathbf{0}$.

For multiple L1-norm penalties and/or a non-negativity constraint, we ex-

press g as a sum of terms, and can minimize g with respect to each term separately, as described by Afonso *et al.* [1]. As in Baek *et al.* [7], however, we use one penalty parameter per element of the sum, instead of setting one global penalty parameter. The resulting algorithms are given as Algorithm 5, for Case (i), and Algorithm 6, for Case (ii). Note that the **I**-minimization steps in Lines 7 and 4 of Algorithms 5 and 6, respectively, are linear least squares problems, which we solve using the conjugate gradient method. We initialize the conjugate gradient method with the result from the previous iteration, $\mathbf{I}^k$, such that it converges quickly.

We have used the convergence criteria discussed in Section 3.3.1 of [12] to determine when to terminate the ADMM iterations. Refer to Algorithms 7 and 8 for Case (i) and Case (ii), respectively. Convergence is determined using a relative tolerance, ϵ_{rel}, and an absolute tolerance, ϵ_{abs}. The relative tolerance prevents the algorithm from stopping until the primal and dual residuals are smaller than a given fraction of the corresponding state variables. The absolute tolerance prevents the algorithm from stopping until the primal and dual residuals are smaller than a given fraction of the magnitude of a typical value in the problem. In our problem, a typical value is a typical spectral intensity in the estimated image. We fix the relative tolerance to 10^{-3}, whereas the absolute tolerance depends on the imaging environment. Instead of manually setting an absolute tolerance for each imaging environment, we automatically estimate an absolute tolerance from the relative tolerance and the input data, as shown in Algorithm 9.

Algorithm 9 uses a loose bound on the values of the estimated latent image to compute the absolute tolerance, ϵ_{abs}. If the output of image estimation was a raw image, then a suitable absolute tolerance would be a given fraction of a typical intensity value in the raw image. We take the median as a typical intensity value, and the fraction as the relative tolerance, and so would arrive at an absolute tolerance of $\epsilon_{\text{rel}} \cdot \texttt{Median}\,(\mathbf{I}_{\text{raw}})$. To find a typical value in the estimated latent image, however, we note that the raw image is formed by integrating the product of the spectral image and the camera spectral response (3.6). We approximate the spectral image with a uniform intensity

Algorithm 5 ADMM for L1-norm penalty terms (Case (i))

Require: Positive penalty parameters ρ_1, ρ_2, ρ_3

1: Initialize $k = 0$, $\mathbf{u}_1^0 = \mathbf{0}$, $\mathbf{u}_2^0 = \mathbf{0}$, and $\mathbf{u}_3^0 = \mathbf{0}$
2: Initialize $\mathbf{I}^0$ to zero
3: $\mathbf{z}_1^0 \leftarrow \nabla_{xy}\mathbf{I}^0$
4: $\mathbf{z}_2^0 \leftarrow \nabla_{\lambda}\nabla_{xy}\mathbf{I}^0$
5: $\mathbf{z}_3^0 \leftarrow \mathbf{I}^0$
6: **repeat**
7: $$\mathbf{I}^{k+1} = \underset{I}{\operatorname{argmin}} \begin{pmatrix} \|\mathbf{M}(\mathbf{\Omega}(\mathbf{\Phi}(\mathbf{I}))) - \mathbf{I}_{\text{raw}}\|_2^2 & +\frac{\rho_1}{2}\left\|\nabla_{xy}\mathbf{I} - \mathbf{z}_1^k + \mathbf{u}_1^k\right\|_2^2 \\ & +\frac{\rho_2}{2}\left\|\nabla_{\lambda}\nabla_{xy}\mathbf{I} - \mathbf{z}_2^k + \mathbf{u}_2^k\right\|_2^2 \\ & +\frac{\rho_3}{2}\left\|\mathbf{I} - \mathbf{z}_3^k + \mathbf{u}_3^k\right\|_2^2 \end{pmatrix}$$
8: $\mathbf{z}_1^{k+1} \leftarrow \textsc{SoftThreshold}(\nabla_{xy}\mathbf{I}^{k+1} + \mathbf{u}_1^k,\ \frac{w_1}{\rho_1})$
9: $\mathbf{z}_2^{k+1} \leftarrow \textsc{SoftThreshold}(\nabla_{\lambda}\nabla_{xy}\mathbf{I}^{k+1} + \mathbf{u}_2^k,\ \frac{w_2}{\rho_2})$
10: $\mathbf{z}_3^{k+1} \leftarrow \textsc{max}(\mathbf{0},\ \mathbf{I}^{k+1} + \mathbf{u}_3^k)$
11: $\mathbf{u}_1^{k+1} \leftarrow \mathbf{u}_1^k + \nabla_{xy}\mathbf{I}^{k+1} - \mathbf{z}_1^{k+1}$
12: $\mathbf{u}_2^{k+1} \leftarrow \mathbf{u}_2^k + \nabla_{\lambda}\nabla_{xy}\mathbf{I}^{k+1} - \mathbf{z}_2^{k+1}$
13: $\mathbf{u}_3^{k+1} \leftarrow \mathbf{u}_3^k + \mathbf{I}^{k+1} - \mathbf{z}_3^{k+1}$
14: $k \leftarrow k + 1$
15: **until** convergence or $k > k_{\max}$

16: **procedure** $\textsc{SoftThreshold}(\mathbf{y}_{n\times 1},\ t)$
17: $\mathbf{y}_t \leftarrow \mathbf{0}_{n\times 1}$
18: **for** $i \leftarrow 1, n$ **do**
19: $a \leftarrow i$-th element of $\mathbf{y}$
20: **if** $a > t$ **then**
21: Set the i-th element of $\mathbf{y}_t$ to $a - t$
22: **else if** $a < -t$ **then**
23: Set the i-th element of $\mathbf{y}_t$ to $a + t$
24: **end if**
25: **end for**
26: **return** $\mathbf{y}_t$
27: **end procedure**

Figure 3.11: The variation of the ADMM algorithm used for L1-norm penalty terms, and a non-negativity constraint (Case (i)). Soft thresholding for minimizing the L1-norm is described in Section 5.2 of [12]. Projecting the solution onto the non-negative orthant in Line 10 enforces the non-negativity constraint (Section 6.3 of [12]).

Algorithm 6 ADMM for L2-norm penalty terms (Case (ii))

Require: A positive penalty parameter, ρ_3

1: Initialize $k = 0$ and $\mathbf{u}_3^0 = \mathbf{0}$

2: Initialize $\mathbf{I}^0$ to zero

3: **repeat**

4: $\quad \mathbf{I}^{k+1} = \underset{I}{\operatorname{argmin}} \begin{pmatrix} \|\mathbf{M}(\mathbf{\Omega}(\mathbf{\Phi}(\mathbf{I}))) - \mathbf{I}_{\text{raw}}\|_2^2 & +\frac{\rho_3}{2}\left\|\mathbf{I} - \mathbf{z}_3^k + \mathbf{u}_3^k\right\|_2^2 \\ & +w_1\left\|\nabla_{xy}\mathbf{I}\right\|_2^2 \\ & +w_2\left\|\nabla_\lambda\nabla_{xy}\mathbf{I}\right\|_2^2 \end{pmatrix}$

5: $\quad \mathbf{z}_3^{k+1} \leftarrow \text{MAX}(\mathbf{0}, \mathbf{I}^{k+1} + \mathbf{u}_3^k)$

6: $\quad \mathbf{u}_3^{k+1} \leftarrow \mathbf{u}_3^k + \mathbf{I}^{k+1} - \mathbf{z}_3^{k+1}$

7: $\quad k \leftarrow k + 1$

8: **until** convergence or $k > k_{\max}$

Figure 3.12: The variation of the ADMM algorithm used for L2-norm penalty terms, and a non-negativity constraint (Case (ii)). Line 5 enforces the non-negativity constraint, as discussed in Section 6.3 of [12].

across the spectrum, let the spectral change-of-basis operator, $\mathbf{B}$ in (3.1), be an identity mapping. Under these conditions, we obtain a lower bound on a typical value in the latent image by dividing a typical value in the raw image by the highest area under the curve of the camera's spectral response. This operation is given as Line 8 in Algorithm 9.

3.8 Implementation

3.8.1 Patch-Wise Image Estimation

Our prototype implementation is written in MATLAB, and is designed for ease of modification as opposed to computational efficiency. We use sparse matrices to represent the latent image to raw image conversion operator, and the image gradient operators, in the ADMM algorithms (Algorithms 5 and 6). Matrices can be precomputed, as can the products of constant matrices, so the ADMM iterations need only evaluate matrix-vector multiplications. An alternative approach would be to create functions that compute the results of applying the matrix operators to the ADMM state variables. Functions could be used because the conjugate gradients method (used in Lines 7 and 4 of

Algorithm 7 ADMM convergence test for L1-norm penalty terms (Case (i))

procedure ConvergenceTest(Relative tolerance ϵ_{rel}, Absolute tolerance ϵ_{abs})
  $r_1 \leftarrow \left\|\nabla_{xy}\mathbf{I}^{k+1} - \mathbf{z}_1^{k+1}\right\|_2$
  $r_2 \leftarrow \left\|\nabla_{\lambda}\nabla_{xy}\mathbf{I}^{k+1} - \mathbf{z}_2^{k+1}\right\|_2$
  $r_3 \leftarrow \left\|\mathbf{I}^{k+1} - \mathbf{z}_3^{k+1}\right\|_2$
  $s_1 \leftarrow \left\|\rho_1\nabla_{xy}^T\left(\mathbf{z}_1^{k+1} - \mathbf{z}_1^{k}\right)\right\|_2$
  $s_2 \leftarrow \left\|\rho_2\left(\nabla_{\lambda}\nabla_{xy}\right)^T\left(\mathbf{z}_2^{k+1} - \mathbf{z}_2^{k}\right)\right\|_2$
  $s_3 \leftarrow \left\|\rho_3\left(\mathbf{z}_3^{k+1} - \mathbf{z}_3^{k}\right)\right\|_2$
  $\epsilon_{1,\text{pri}} \leftarrow \epsilon_{\text{abs}}\sqrt{M} + \epsilon_{\text{rel}}\text{MAX}(\left\|\nabla_{xy}\mathbf{I}^{k+1}\right\|_2, \left\|\mathbf{z}_1^{k+1}\right\|_2)$ ▷ M is the number of elements of $\mathbf{I}$
  $\epsilon_{2,\text{pri}} \leftarrow \epsilon_{\text{abs}}\sqrt{M} + \epsilon_{\text{rel}}\text{MAX}(\left\|\nabla_{\lambda}\nabla_{xy}\mathbf{I}^{k+1}\right\|_2, \left\|\mathbf{z}_2^{k+1}\right\|_2)$
  $\epsilon_{3,\text{pri}} \leftarrow \epsilon_{\text{abs}}\sqrt{M} + \epsilon_{\text{rel}}\text{MAX}(\left\|\mathbf{I}^{k+1}\right\|_2, \left\|\mathbf{z}_3^{k+1}\right\|_2)$
  $\epsilon_{1,\text{dual}} \leftarrow \epsilon_{\text{abs}}\sqrt{M} + \epsilon_{\text{rel}}\left\|\nabla_{xy}^T\left(\rho_1\mathbf{u}_1^{k+1}\right)\right\|_2$
  $\epsilon_{2,\text{dual}} \leftarrow \epsilon_{\text{abs}}\sqrt{M} + \epsilon_{\text{rel}}\left\|\left(\nabla_{\lambda}\nabla_{xy}\right)^T\left(\rho_2\mathbf{u}_2^{k+1}\right)\right\|_2$
  $\epsilon_{3,\text{dual}} \leftarrow \epsilon_{\text{abs}}\sqrt{M} + \epsilon_{\text{rel}}\left\|\rho_3\mathbf{u}_3^{k+1}\right\|_2$
  if $\forall i$ $(r_i < \epsilon_{i,\text{pri}}$ and $s_i < \epsilon_{i,\text{dual}})$ **then**
    return Converged
  else
    return Not converged
  end if
end procedure

Figure 3.13: The convergence criteria used in the ADMM algorithm for L1-norm penalty terms, and a non-negativity constraint (Case (i)), based on Section 3.3.1 of [12]

Algorithm 8 ADMM convergence test for L2-norm penalty terms (Case (ii))

procedure ConvergenceTest(Relative tolerance ϵ_{rel}, Absolute tolerance ϵ_{abs})
    $r_3 \leftarrow \left\|\mathbf{I}^{k+1} - \mathbf{z}_3^{k+1}\right\|_2$
    $s_3 \leftarrow \left\|\rho_3\left(\mathbf{z}_3^{k+1} - \mathbf{z}_3^{k}\right)\right\|_2$
    $\epsilon_{3,\text{pri}} \leftarrow \epsilon_{\text{abs}}\sqrt{M} + \epsilon_{\text{rel}}\text{MAX}(\left\|\mathbf{I}^{k+1}\right\|_2, \left\|\mathbf{z}_3^{k+1}\right\|_2)$
    $\epsilon_{3,\text{dual}} \leftarrow \epsilon_{\text{abs}}\sqrt{M} + \epsilon_{\text{rel}}\left\|\rho_3\mathbf{u}_3^{k+1}\right\|_2$
    if $r_3 < \epsilon_{3,\text{pri}}$ and $s_3 < \epsilon_{3,\text{dual}}$ **then**
        return Converged
    else
        return Not converged
    end if
end procedure

Figure 3.14: The convergence criteria used in the ADMM algorithm for L2-norm penalty terms, and a non-negativity constraint (Case (ii)), based on Section 3.3.1 of [12]

Algorithm 9 Computation of the absolute convergence tolerance for ADMM

Require: Relative tolerance ϵ_{rel}, raw image $\mathbf{I}_{\text{raw}}$, and discretized camera spectral response $\mathbf{Q}_c$, $c \in \{\text{Red, Green, Blue}\}$

$q_{\max} \leftarrow -\infty$
for $c \in \{\text{Red, Green, Blue}\}$ **do**
    $q \leftarrow \sum_\lambda \mathbf{Q}_c$
    if $q > q_{\max}$ **then**
        $q_{\max} \leftarrow q$
    end if
end for
$\epsilon_{\text{abs}} \leftarrow \epsilon_{\text{rel}} \dfrac{\text{Median}(\mathbf{I}_{\text{raw}})}{q_{\max}}$.
return ϵ_{abs}

Figure 3.15: Determination of the absolute tolerance, ϵ_{abs}, for ADMM, from the relative tolerance, ϵ_{rel}, and the input raw image, $\mathbf{I}_{\text{raw}}$

Algorithms 5 and 6, respectively) does not require the matrices themselves, only the results of their application. Unfortunately, to achieve the equivalent of matrix multiplication precomputation with functions, we would need to design monolithic functions to replace simpler functions implementing the individual operators $\mathbf{M}$, $\mathbf{\Omega}$, *etc.*, in the optimization problem (3.1). Consequently, our prototype would be difficult to modify.

The primary disadvantage of using matrix as opposed to function representations of our operators is that matrices consume large amounts of memory. Nevertheless, we need our program to process high-resolution images. Our solution is to divide the image into patches, with some overlap between patches to eliminate border artifacts. Consequently, we can process very large images, and we can also compute the solutions for individual patches in parallel. We have not noticed visible artifacts at the boundaries between patches, except when the regularization penalties are given too much weight. As such, visible boundaries between patches is a useful indicator to the user that the regularization weights need to be reduced. Therefore, we do not blend values from different patches in their overlapping regions when combining the results from individual patches to form the output image. By simply discarding the overlapping regions, we eliminate the need for coordination between threads processing neighbouring patches.

While we originally implemented patch-wise image estimation for computational efficiency, we since realized several other benefits from doing so. In particular, regularization weight selection (Section 3.6) is prohibitively time-consuming to perform when evaluating regularization weight selection criteria on the entire image. We can therefore select regularization weights on one image patch, or on a small number of patches, provided that each patch is large enough to contain a good sample of image features, such as edges. Alternatively, for very high-quality results, we could select regularization weights that are specific to each patch of the image being estimated, provided that the large computational cost is acceptable. Another major benefit of patch-wise image estimation is adaptive sampling of the dispersion model. Specifically, we choose a spectral resolution at which to compute the dispersion operator,

$\mathbf{\Phi}$, for each patch. Therefore, we save time sampling our model of dispersion in patches with little dispersion, whereas in patches with significant dispersion, we still generate high-quality dispersion operators.

3.8.2 Running Time Comparison with Other Systems

The running time of our method is comparable to the running times of similar global optimization approaches [7, 19], although comparisons are only approximate because of differences in computer hardware. Choi *et al.* provided an implementation of their method [19], and we observed that it has performance limitations similar to those that we resolved using a patch-wise decomposition. Specifically, their TensorFlow[2] implementation consumes high amounts of memory when processing large images. We note that other authors downsample the input images used for global optimization-based spectral image estimation [7], or report results on small images, such as 512×512 [108], 600×860 [7], or 256×256 [27] pixels.

We have performed a speed comparision with the prototype of Choi *et al.*, after refactoring it so that it estimates images in patches (sequentially, but the solution for each patch is computed using our GPU). The refactoring was necessary in order to run it on the high-resolution images from their dataset. We also replaced their modulation matrix (defined in Section 2.3.1) with one representing basic spectral-to-colour conversion and colour-filter array sampling, in order to compare our system with theirs under the same image formation model. (We did not include dispersion in the modulation matrix, because their dataset does not allow for dispersion calibration.) Our modified version of their prototype required 20 hours to compute a 2018×3072 spectral image with 31 bands, given their default number of 20 ADMM iterations. Our method does not use GPU acceleration, but processes four patches in parallel on our current hardware. When estimating the same image at a resolution of 8 control bands, in the worst case, we performed 1000 ADMM iterations, and 500 conjugate gradients iterations within each ADMM iteration. Under these

[2]https://www.tensorflow.org/

conditions, our system required 112 hours to process the image[3]. Assuming linear scaling with the number of ADMM iterations, the number of spectral bands being estimated, and the inverse of the number of parallel processing units, our system performed approximately $\frac{1000}{20} \cdot \frac{8}{31} \cdot \frac{1}{4} = 3.22$ times as much work, and so may have taken at most 35 hours to perform 20 ADMM iterations. The comparison is inconclusive because of the differences in the scale and number of iterations of optimization, and in the parallelization methods (GPU vs. CPU-based multithreading)

Our method has regularization terms which are much simpler than the convolutional autoencoder-based image prior of Choi *et al.* [19], so we expect that our method can run faster in general, depending on how it is implemented. In contrast, we have added a non-negativity constraint relative to the first stage of the method of Baek *et al.* [7]. As Baek *et al.* have two further global optimization stages in their method, however, our method is likely faster than theirs overall. We leave in-depth analysis of running time, accounting for the number of iterations required to obtain the same accuracy as comparison methods, to future work.

3.9 Algorithm Variants

The preceding sections described our complete image estimation algorithm, including calibration. We presented several different regularization penalties (Section 3.5), and noted that we tested our algorithm with and without a non-negativity constraint on the latent spectral image. In addition to these minor variations, we tested the following alternative models of image formation.

3.9.1 Direct Colour Image Estimation

In direct colour image estimation, we model the latent image $\mathbf{I}$ as an image in the camera's raw colour space, and estimate it by solving the simpler opti-

[3] In this case, we used the `L1SpatialLap2` combination of penalty terms (Table 3.1), with a non-negativity constraint on the estimated image. The weights on the penalty terms were selected using the MDC, and were very large, leading to poor convergence.

mization problem:

$$\mathbf{I}^* = \underset{\mathbf{I}}{\operatorname{argmin}} \|\mathbf{M}\left(\mathbf{\Phi}\left(\mathbf{I}\right)\right) - \mathbf{I}_{\text{raw}}\|_2^2 + \sum_{j=1}^{N} w_j \mathbf{S}_j\left(\mathbf{I}\right), \quad \mathbf{I} \geq 0 \tag{3.27}$$

In (3.27), the dispersion operator, $\mathbf{\Phi}$, models shifts between colour channels. In this case, we have not simulated $\mathbf{\Phi}$ from a spectral model of dispersion, but have approximated lateral chromatic aberration with colour channel warping, in keeping with previous works (specifically Rudakova and Monasse [99]). We calibrate dispersion by modelling shifts between colour channels in images of disk patterns captured under wideband illumination. The procedure is the same as described in Section 3.2, but is less robust because of the colour-filter array. As noted in Section 3.2.1, if we use disk keypoints to measure dispersion, we must fit them separately to individual colour channels. For dispersion estimation by patch-wise image registration (Section 3.2.3), we would need to demosaic all image channels to register them with the Green channel, thus introducing additional error from demosaicing.

3.9.2 Image Estimation without Dispersion

Another simpler optimization problem disregards dispersion, treating $\mathbf{\Phi}$ in (3.1) as an identity mapping. After image estimation, we then warp the image to correct lateral chromatic aberration.

Ignoring dispersion during image estimation has two clear disadvantages. First, dispersion may serve as a useful constraint to improve spectral image estimation [7], so the image may be less accurately-estimated. Second, taking dispersion correction outside of our global optimization algorithm may remove benefits of global optimization, such as the slight deblurring effect caused by bilinear interpolation in the dispersion operator (Section 3.2.5). We illustrate the deblurring effect later in Figure 5.13.

On the other hand, there are advantages to separating image estimation from dispersion correction. In particular, any errors in the model of dispersion will not affect image estimation. Also, the optimization problem (3.1) is easier to solve because the data-fitting term has a higher degree of sparsity. Dispersion reduces sparsity by increasing the number of pixels in the estimated

image on which a given pixel in the simulated raw image, $\mathbf{M}(\mathbf{\Omega}(\mathbf{\Phi}(\mathbf{B}(\mathbf{I}))))$, depends.

Chapter 4

Experimental Design

Experimental validation of algorithms for spectral reconstruction from RGB images is extremely challenging. To evaluate a reconstructed spectral image, one would need an imaging system that can capture RGB and spectral images (high spectral resolution images) using the same physical pixels. Furthermore, such a system should generate RGB images with the same noise characteristics, crosstalk, chromatic aberration, and other non-idealities as consumer-level digital cameras.

4.1 Approaches to Image Reconstruction Evaluation

4.1.1 Spectral Reconstruction Evaluation

Other researchers have approached experimental validation of spectral image reconstruction in two ways. First, they have collected spectral images using line-scanning spectral imaging systems [4, 17, 84], or using monochromatic cameras behind narrowband optical filters, such as liquid crystal tunable filters [19, 80, 128]. From these images, they have simulated RGB images using the spectral sensitivity functions of trichromatic cameras, or the CIE spectral tristimulus functions. We refer to experiments of this type as simulated experiments, because the RGB images are not affected by realistic imaging artifacts. Simulated artifacts are usually those which the image reconstruction algorithms are designed to be robust against, and the resulting RGB images

lack complex interactions that would occur between these artifacts and other artifacts not simulated. For example, the NTIRE 2018 challenge on spectral reconstruction from RGB images included images with simulated JPEG compression and quantization, but spectral reconstruction algorithms performed well to the point where the organizers suggested that other sources of image noise be simulated in future challenges [5].

The second major approach to image evaluation starts from captured RGB images instead of captured spectral images. Spectral information from the scene presented to an RGB camera is collected separately, such as with a point spectrometer. Typically, data-driven spectral reconstruction algorithms show poorer performance relative to the reference spectral information than when tested on simulated RGB images [4, 53], although some authors report comparable performance on both simulated and real images [2]. The majority of works on data-driven spectral reconstruction algorithms do not report results on real RGB images.

While using captured RGB images as input helps assess how a spectral reconstruction algorithm will perform in practice, such evaluations are far from conclusive. It is unclear how much of the error is attributable to uncertainty in the camera's spectral response functions, if they are used by the image reconstruction algorithm. The evaluations are also less reliable because of the small number of spectral measurements that can be reasonably acquired manually. Unfortunately, spectrometers can only characterize relatively large, textureless patches on objects in the scene. As such, spectrometry data is unavailable at strong edges in the image, even though edges reveal chromatic aberration, and test the limits of demosaicing algorithms. Lastly, spectrometry measurements are often relative to a reference object, rather than giving the absolute spectral radiance incident on the camera.

4.1.2 Chromatic Aberration Correction Evaluation

Evaluating chromatic aberration correction is even more challenging than evaluating spectral reconstruction. Chromatic aberration is difficult to simulate, although a few authors have added simulated chromatic aberration to images

from spectral image datasets [60, 122]. Korneliussen and Hirakawa provide few details concerning their simulations other than stating that they modeled chromatic aberration according to the principles of Gaussian optics, but applied a modulation transfer function to account for diffraction [60]. Wang *et al.* [122] describe their simulation pipeline in more detail, showing that it approximates chromatic aberration by applying geometric distortion (lateral chromatic aberration) and blurring (longitudinal chromatic aberration and other optical aberrations) separately.

Other authors, working on chromatic aberration correction in the RGB domain, have approximated chromatic aberration by magnifying and/or blurring colour channels [18, 111]. Sun *et al.* appear to have simulated chromatic aberration as per-colour channel blurring, despite using images from a spectral image dataset as input [111]. While RGB-based simulated chromatic aberration can be accurately restored by RGB-based chromatic aberration correction algorithms, these experiments are not well-grounded in the principles of optics.

As for chromatic aberration simulation in spectral images, the simulated chromatic aberration may be physically-realistic, but it is applied on top of any existing chromatic aberration in the spectral images. Other artifacts in the images may be very different from the typical artifacts in RGB images. In particular, line-scanning systems likely do not produce images with the same signal characteristics as images captured by conventional consumer cameras with single-frame 2D image capture. In Figure 4.1, we show an example of spectral dispersion in the image from a line-scanning system that would interfere with added synthetic chromatic aberration.

Spectral images captured using narrowband optical filters and monochromatic cameras are more closely-related to images taken by RGB cameras than are spectral images from line-scanning systems. Unfortunately, optical filters introduce spectral dispersion into the captured images, regardless of whether they are tunable filters [7], or regular filters in a filter wheel [14]. Figure 4.2 shows an example of colour fringes in an image from a filter-based spectral imaging system.

Given the lack of suitable input images for simulating chromatic aberra-

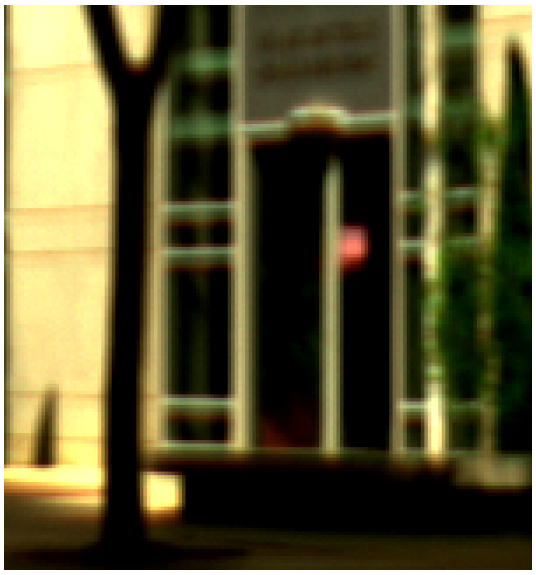


Figure 4.1: A portion of the `bgu_0403-1511` image from the ICVL hyperspectral image dataset [4], showing colour fringes from spectral dispersion or other imaging artifacts. Colour fringes are clearest on the thin horizontal window frames around the door shown in the image. Images in this dataset were captured using a line-scanning spectral camera. The RGB image shown was provided with the dataset.

Figure 4.2: A portion of the `scene04` image from the KAIST hyperspectral image dataset [19], showing colour fringes from spectral dispersion or other imaging artifacts. Images in this dataset were captured using a monochromatic camera equipped with an apochromatic lens and a liquid crystal tunable filter. The RGB image shown was provided with the dataset.

tion, and the complexity of realistic simulation, we have evaluated chromatic aberration correction only in images reconstructed from captured RGB images. Unfortunately, we lack ground truth aberration-free images to compare against. Our evaluation relies on being able to measure residual chromatic aberration in a single image, as we discuss in Section 4.4.3.

4.2 Real Experiments

4.2.1 Apparatus

The primary objective of the apparatus we used for our real experiments is to generate realistic RGB images. Therefore, we use a BlackFly Flea3 colour camera (Point Grey, model FL3-GE-50S5C, pixel resolution of 2448×2048), as opposed to a monochromatic camera, or a spectral camera. We tested two lenses with the same camera, a Computar 07I lens, and a Fujifilm Fujinon 1:1.4/12.5 mm lens. We calibrated spectral dispersion of both lenses, but only used images captured using the Computar 07I lens for our image reconstruction experiments, because both input image capture, and spectral image reconstruction, are time-consuming. In order to calibrate spectral dispersion, we need to capture images under narrowband illuminations. To do so without altering the spectral dispersion affecting the captured images, we filtered the illuminant, rather than placing a filter between the camera and the scene. Alternatively, we could have lit the scene using narrowband illumination, but narrowband illumination is generally more expensive than a limited number of optical bandpass filters.

Our experimental apparatus is shown in Figure 4.3. A large box with a single hole provides a controlled-illumination environment into which we place the camera and scene. The light source is a DLP projector (Optoma EP739) that has had its colour filter wheel removed to provide high-intensity, broad spectrum light. When we capture conventional RGB images, we illuminate the scene using unfiltered light from the projector, configured to project a white test pattern. To capture images under narrowband-filtered illumination, we use seven bandpass filters, having 10 nm full-width half-maxima, optical

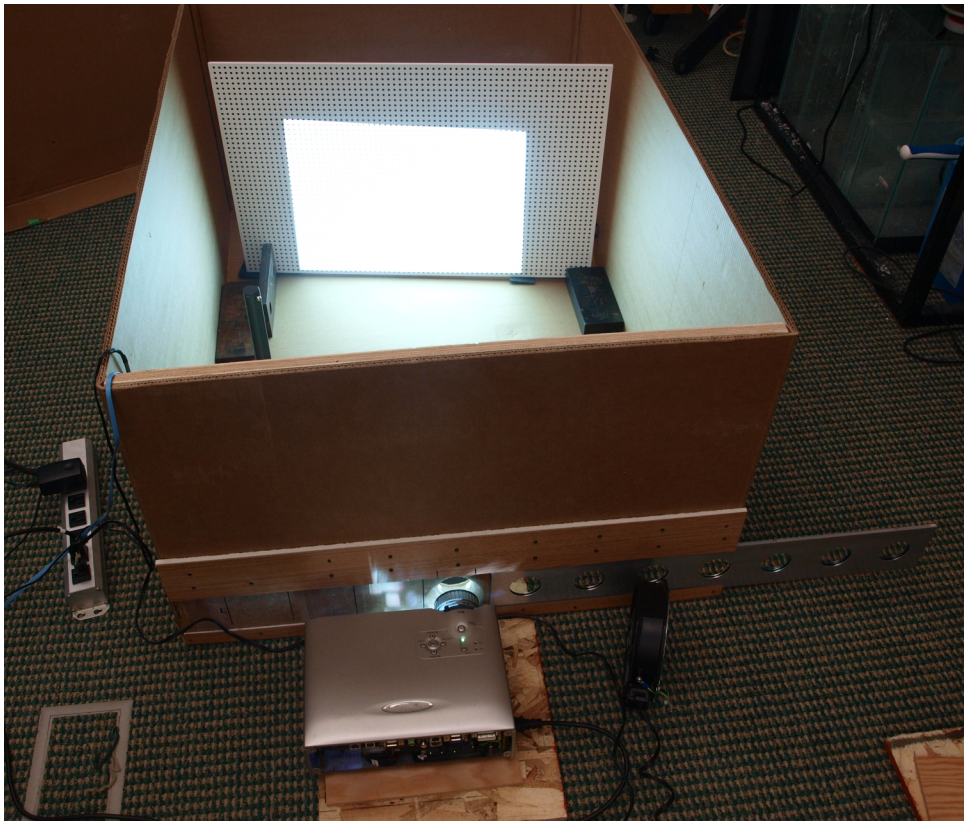


Figure 4.3: The experimental apparatus we used to collect real RGB images and bandpass-filtered images of static scenes, consisting of a light-proof box, a DLP projector, a camera, and a row of optical bandpass filters. The disk calibration pattern we used to estimate dispersion is also visible at the back of the box.

densities of 4, and 50 mm diameters, from Edmund Optics. The filters are supported by a sliding aluminum bar between the projector and the box, and filter the illumination to narrow ranges around 400, 450, 500, 550, 600, 650, and 700 nm, respectively.

We note that our apparatus is not suitable for highly-accurate imaging experiments, although we would need access to much higher-quality spectrometry equipment to quantify the accuracy it allows. There are two major sources of error: The first is the instability of the projector's spectral power distribution. The second is the lack of proper collimation of the light passing through the optical bandpass filters. As our filters are intereference filters, their passbands

will change depending on the angle of the incident light.[1] Fortunately, we roughly verified that the filtered light has the appropriate spectral power distribution by conducting an experiment with just the filters and the projector, at the Core-Net Spectral Laboratory.[2] With the assistance of Dr. Jilu Feng, and the permission of Dr. Benoit Rivard, we manually held each filter in front of the projector, and measured the spectral power distribution of the light reflected from a white calibration tile, relative to the light reflected from the tile without any filter. The resulting relative spectral radiance measurements, taken using an Analytical Spectral Devices point spectrometer, are shown in Figure 4.4. Aside from the 450 nm filter, which seems to have an additional transmission peak close to the transmission peak of the 400 nm filter, all filters conformed to the sample transmission curves provided by Edmund Optics. In our actual experimental setup, however, the filtered light will deviate from the measurements shown in Figure 4.4, because of the different optical environment surrounding the filters.

A final detail of our apparatus visible in Figure 4.3 is the pattern of black dots on a white background which we used to calibrate lateral chromatic aberration of the camera lens, by finding disk keypoints as described in Section 3.2.1. The pattern is a high-quality matte print mounted on a foam core board 711 $\times$ 457 mm in size. Each disk has a radius of approximately 2 mm, and the disks are arranged in a grid with a side length of approximately 9.4 mm. During image collection, we placed the disk pattern at a distance between 30 and 50 cm from the camera.

4.2.2 Image Collection

In hindsight, our experimental apparatus should have included an additional hole in the sliding bar shown in Figure 4.3. In this hole, we would have inserted a neutral density filter, to reduce the intensity difference between the broad-spectrum light from the projector and the narrowband-filtered light. Even then, the narrowband-filtered illuminations still cover a large dynamic

[1] https://www.edmundoptics.com/resources/application-notes/optics/optical-filters/
[2] http://coresensing.net/index.html

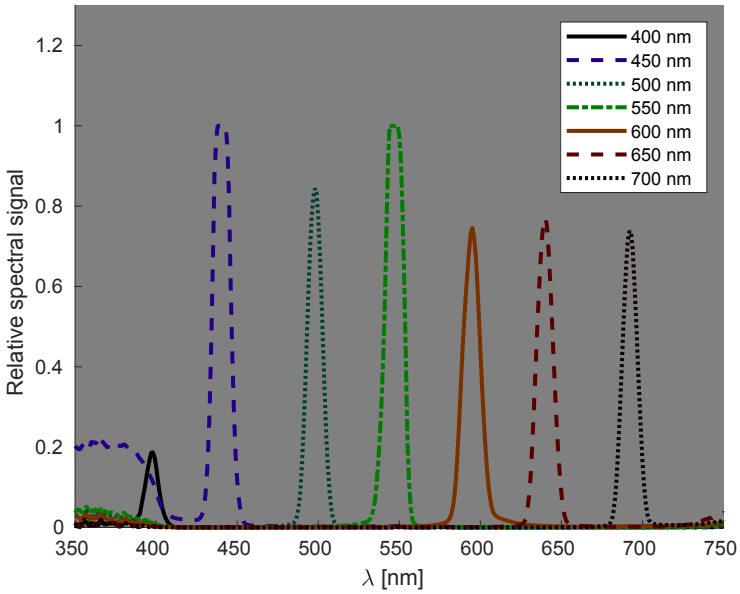


Figure 4.4: Spectral radiance of the light from the Optoma EP739 DLP projector (without a colour wheel), filtered by the 400, 450, 500, 550, 600, 650, and 700 nm centre wavelength optical bandpass filters. Spectral radiance is expressed relative to the unfiltered light from the projector. The measurements were taken with an Analytical Spectral Devices point spectrometer, sensing the light reflected from a white calibration tile. (More details on the spectrometer are in Section 4.6.1.)

range, given the variation in the camera's spectral response with wavelength (Figure 3.6), and the non-uniform spectral power distribution of the projector. We developed the high-dynamic range imaging procedure presented in Algorithm 10 in order to create properly-exposed images under these challenging conditions.

Algorithm 10 uses multiple exposure times to capture images which, when combined, allow every colour channel to be properly exposed at every point in the scene. For our BlackFly Flea3 colour camera with a Computar 07I lens at an aperture of $f/4$, for instance, we selected exposure times of 3, 6, and 12 ms for unfiltered light, and 25, 50, 100, 250, 500, 2000 and 3916 ms for filtered light. To further improve our images, Algorithm 10 takes multiple replicates of each image, which can then be averaged to reduce noise. We set a higher number of replicates for images taken under unfiltered light, because the variability in the shutter time is larger in relation to the shorter exposure times, and because these images are more sensitive to fluctuation in the projector's spectral power distribution.

4.2.3 High-Dynamic Range Image Synthesis

The images we capture as described in Section 4.2.2 are mosaiced low-dynamic range RGB images captured under a variety of shutter times. The images contain raw pixel values. In other words, the images have not been subject to post-processing such as white-balancing, or gamma adjustment. Our high-dynamic range (HDR) image synthesis procedure aims to map all images to a reference exposure time, and then blend them together to produce high-dynamic range images with little noise, and without artifacts from clipping or blooming.

To begin, we average together the 5 or 15 replicates of each image, and then subtract the corresponding average dark frame image from each average image. Assuming an ideal image sensor, but with some thermal noise, the resulting images should be linearly related to the irradiance of the sensor. Therefore, to map the images to a common exposure, we could divide each image by its exposure time [98]. The assumption of linearity seems justified, as shown

Algorithm 10 Image collection procedure

1: Set the DLP projector to display a uniform white pattern.
2: Adjust the camera lens to focus on typical distances in the scene.
3: Adjust the aperture of the lens so that images of a white panel are properly exposed at short shutter times, under the full intensity of the projector's light.
4: Select several exposure times at which to capture images under the full intensity of the projector's light. The exposure times must collectively allow each colour channel to be properly exposed.
5: Filter the light from the projector with each of the optical bandpass filters (with passbands centered at 400, 450, 500, 550, 600, 650, and 700 nm). Select several exposure times which collectively allow for properly exposed images, in each colour channel, for each filtered lighting condition.
6: Cover the hole in the box through which light enters from the projector. Capture dark frames for dark frame subtraction, at all of the exposure times determined previously. For the exposure times from Step 4, capture 15 dark frames, whereas for the exposure times from Step 5, capture 5 dark frames.
7: **for** each scene to be imaged **do**
8: Capture 15 images at each of the exposure times determined in Step 4 under unfiltered light from the projector.
9: Capture 5 images at each of the exposure times determined in Step 5 under light from the projector filtered by each of the optical bandpass filters.
10: **end for**

Figure 4.5: Our exposure bracketing procedure for capturing images of real scenes in our experiments with our controlled light environment (Figure 4.3). For all combinations of camera lenses and lens settings, we included the disk pattern used to calibrate chromatic aberration as at least one of the scenes.

Table 4.1: A comparison of exposure blending scaling factors fitted between dark-subtracted images taken under different exposures with the scaling factors predicted from the exposure times under which the images were taken. Scaling factors computed for different colour channels (last three columns) agree well with each other, and with the predicted scaling factors (second column), suggesting that the BlackFly Flea3 camera's image sensor is highly linear.

Shutter time, t	Ratio $t_{\min}/t$	Red	Green	Blue
25	1	1.0000	1.0000	1.0000
50	0.5	0.5057	0.5090	0.5097
100	0.25	0.2558	0.2574	0.2598
250	0.1	0.1049	0.1053	0.1050
500	0.05	0.0555	0.0542	0.0537
2000	0.0125	0.0147	0.0146	0.0141
3916	0.00638	0.0080	0.0078	0.0074

in Figure 4.6, but we do not want to assume known exposure times, so we calibrate scaling factors relating different exposures, following the approach of Darrodi *et al.* [22]. For each consecutive pair of exposures, we select pixels with values under both exposures that are at least 2%, but no greater than 95%, of the saturation value. We compute the first principal component of the pixels to find the scaling factor relating the two exposures. By multiplying the scaling factors relating different pairs of consecutive exposure times, we then obtain scaling factors relating all exposures to the highest exposure, which we selected as our reference exposure. As a precaution, we repeated the calibration process for each colour channel separately, but found that there was little difference between colour channels (Table 4.1). We use the shortest shutter time as the reference exposure, so that all pixels in the high-dynamic range image will have values less than or equal to the saturation value. Note that we do not calculate scaling factors between all exposures simultaneously, but do so between pairs of exposures, because there are very few pixels that are properly exposed at all exposures. In practice, we found scaling factors which agreed well with the shutter times, as shown in Table 4.1.

When calibrating scaling factors to map images to the reference exposure, we can leverage the large number of pixels for improved precision and ro-

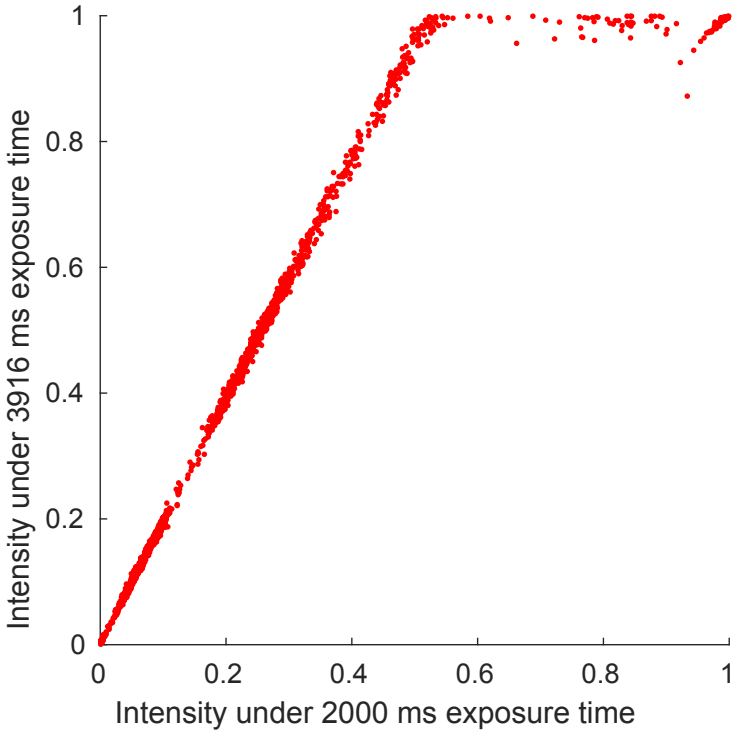


Figure 4.6: A scatterplot of pixel values for the BlackFly Flea3 camera under two different exposures, for images of the disk pattern captured under unfiltered light from the DLP projector. The strongly linear relationship between the values at the two exposures suggests that the image sensor responds linearly to incident light. The deviation from linearity at high pixel values is caused by saturation.

bustness. In contrast, blending the scaled images is more difficult because it involves pixel-level decisions. Lindstrand noted that sensor crosstalk, or interpixel correlation, can be highly detrimental at this stage [69]. Unfortunately, crosstalk is very difficult to calibrate in general, so Lindstrand focused exclusively on sensor bloom, and experimented with different exclusion neighbourhoods around saturated pixels to prevent bloom from affecting the exposure-blended image [69]. We also focus on eliminating bloom in our exposure blending procedure, but did not experiment with different exclusion neighbourhoods. More generally, we do not claim that our high-dynamic range image synthesis procedure is state-of-the-art, especially as high-dynamic range imaging is not the main focus of our research.

In detail, our exposure blending procedure takes as input a set of images scaled to a common exposure, with values in the range $[0, 1]$, where 1 is the saturation value. The output image is a weighted average of the images, computed using per-pixel weights. Each pixel in each image is assigned a *blooming weight*, computed using a polynomial function inspired by the discussion in Reinhard *et al.* [98]

$$w_{\text{bloom}}(p) = \begin{cases} 1 - (2p-1)^{12}, \text{ if } p > 0.5 \\ 1, \text{ otherwise} \end{cases} \tag{4.1}$$

To remove blooming, the blooming weights are subject to greyscale erosion using a disk structuring element of radius 1. The greyscale erosion will down-weight pixels near saturated pixels, on the assumption that their values are corrupted, even if they are not saturated. Each pixel is also assigned a *local weight*, again computed based on the discussion in Reinhard *et al.* [98]

$$w_{\text{local}}(p) = p\left(1 - (2p-1)^{12}\right) \tag{4.2}$$

The final weight for each pixel is the minimum of the two weights. Pixels having weights of zero in all images are given a weight of one in the image taken under the shortest exposure time. The weights are renormalized, and then used to compute the exposure-blended image. We restate our high-dynamic range image synthesis procedure for reference in Algorithm 11.

Algorithm 11 High-dynamic range image synthesis

Require: All images $\{I_k\}$ taken for a given scene, under the same lens settings and illumination conditions (*e.g.* same optical bandpass filter), where k indexes exposure times. Images have been averaged across replicates, dark-frame subtracted, and scaled to a reference exposure. Intensities at the reference exposure range from 0 to 1.

1: **for** $k \leftarrow 1, K$ **do** ▷ Iteration over exposures
2: **for** each pixel in I_k, having intensity I_{kx} **do**
3: Calculate $w_{kx,\text{bloom}}(I_{kx})$ (4.1)
4: Calculate $w_{kx,\text{local}}(I_{kx})$ (4.2)
5: **end for**
6: Update $\{w_{kx,\text{bloom}}\}$ for all pixels in I_k by greyscale erosion
7: **for** each pixel in I_k at index x **do**
8: $w_{kx} \leftarrow \text{MIN}(w_{kx,\text{bloom}}, w_{kx,\text{local}})$
9: **end for**
10: **end for**
11: $H \leftarrow \mathbf{0}$ ▷ Initialize the output image
12: **for** each pixel in H, at index x **do**
13: **if** $\forall k\ w_{kx} = 0$ **then**
14: $H_x \leftarrow I_{1x}$ ▷ Copy pixel from the shortest exposure image
15: **else**
16: $H_x \leftarrow \dfrac{\sum_{k=1}^{K} w_{kx} I_{kx}}{\sum_{k=1}^{K} w_{kx}}$
17: **end if**
18: **end for**
19: **return** H

Figure 4.7: Our procedure for synthesizing high-dynamic range images from the images collected by exposure-bracketing (Algorithm 10)

4.2.4 Multispectral Image Synthesis

Given that our dataset of high-dynamic images contains images captured under both wideband and narrowband-filtered illuminations, one might suppose that we have collected ground truth spectral images corresponding to the RGB images captured under wideband illumination. Unfortunately, the two sets of images are not comparable, primarily because we have a small set of filtered illuminations that do not adequately sample the visible spectrum. There are other, subtler discrepancies between the filtered and unfiltered imaging conditions, however. First, we observed that each bandpass filter has a different vignetting effect on the transmitted light. It is difficult to correct this vignetting in the captured images, except if the scene being imaged was planar, because of parallax between the camera and the projector. In other words, the vignetting effect, as seen by the camera, depends on the depth of objects in the scene. The second issue is that crosstalk is nonlinear, and spectrally-varying [69], and so its effect on the unfiltered-light image is not equivalent to the sum of its effects on the filtered-light images.

Therefore, we create two datasets of input images for spectral reconstruction and chromatic aberration correction. The first is the set of high-dynamic range images corresponding to the images taken under unfiltered illumination, shown later in Figure 4.15. The second is a set of synthetic RGB images created from the images for all filtered illuminations. The synthetic RGB images do not have the same properties as images captured under unfiltered light, but at least they have a known relationship with the filtered-light images, allowing us to quantitatively evaluate spectral reconstruction at the level of individual pixels.

Naively, we could sum over the filtered-light images to compute synthetic mosaiced RGB images. If we then demosaic such an image, the result would appear as in Figure 4.8b. While this image is visually acceptable compared with the true RGB image, Figure 4.8a, it is not clear what the corresponding multispectral image represents, because we used a trichromatic camera. In Figure 4.9a, we show the high-dynamic range, mosaiced image captured under

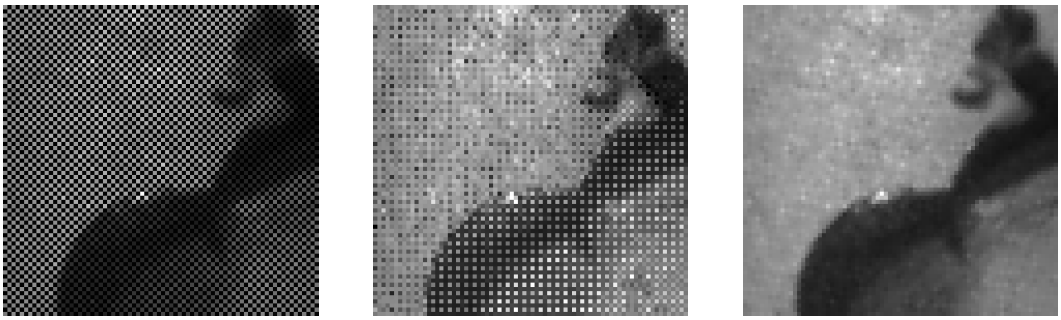
(a) (b) (c)

Figure 4.8: A comparison of different methods for producing RGB images from our experimental setup: (a) An image captured under unfiltered light, demosaiced by bilinear interpolation. (b) An image created by summing over the images captured under each filtered illumination, and then demosaiced by bilinear interpolation. (c) An image created by using bilinear interpolation to demosaic the Green channel of the images captured under each filtered illumination, and then collapsing the Green channel images to an RGB image by applying the calibrated relative spectral sensitivities of the Red and Blue channels. (c) is our preferred approach for simulating RGB images from narrowband-filtered illumination images. Note that all images have been colour-corrected (as described in Section 4.5).

the 550 nm optical bandpass filter. The intensity of light in the spectral band around 550 nm depends on which colour channel is used to measure it. In contrast, real RGB images are formed from spectral irradiances that are colour channel-independent.

To obtain more appropriate spectral band images, we studied the relationships between individual colour channels in textureless regions of the image. Under the assumption that the sensor responds linearly to light, and that there is little crosstalk between pixels, we calibrate best-fit scaling factors between colour channels, as detailed in Algorithm 12.

In most cases, we observed fairly linear relationships between different colour channels in our high-dynamic range images, such as shown in Figure 4.11a, but sometimes the relationships were highly nonlinear, as shown in Figure 4.11b. Some of the nonlinearity may be caused by sensor crosstalk, but it may have been exacerbated by our high-dynamic range image synthesis procedure. We doubt that the sensor itself is nonlinear, irrespective of crosstalk, based on the linear relationship we observed between exposure time and pixel



(a) (b) (c)

Figure 4.9: A comparison of different methods for producing spectral band images from our experimental setup: (a) The raw image captured under light filtered by the 550 nm optical bandpass filter, showing the Bayer filter pattern. (b) The same image divided by the best-fit scaling factors computed between colour channels, showing that the scaling factors do not adequately model the relationships between colour channels (otherwise, the image would be smooth). (c) A dense version of the Green channel of the image, obtained by bilinear interpolation.

values (Figure 4.6). We also doubt that the nonlinearity is a result of improper filtering of the light, as our optical bandpass filters seem to be operating as intended (Figure 4.4).

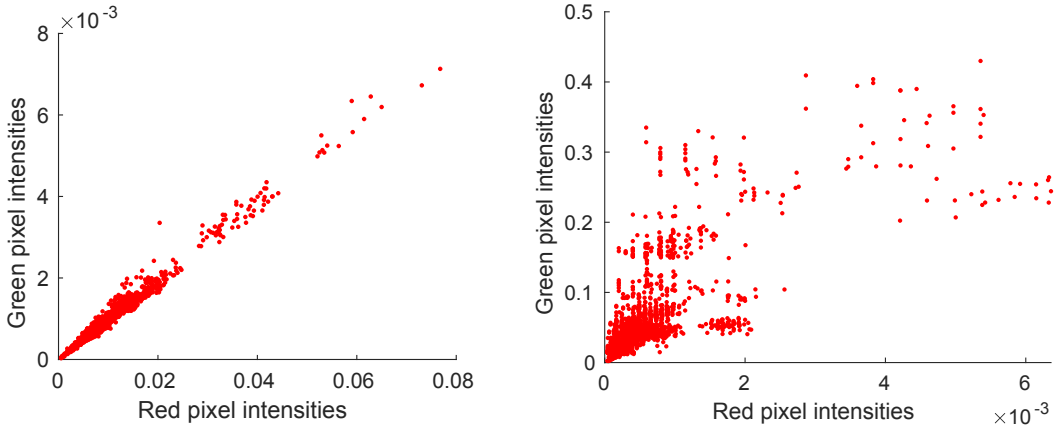
In the future, we may consider calibrating the scaling factors between colour channels from the original low-dynamic range images, to see whether the nonlinear relationships between colour channels were caused by our high-dynamic range image synthesis procedure. We may also consider a more general relationship between colour channels that can accommodate crosstalk. Nevertheless, our pairwise scaling factors between colour channels agree reasonably well with the scaling factors predicted from the spectral sensitivities measured by FLIR [32] for the Sony ICX655, 2/3" image sensor (Figure 3.6), as shown in Figure 4.12.

Using the scaling factors we calibrated between colour channels, we can approximate values in the Green channel at the Red and Blue pixels in an image, in order to synthesize multispectral images, such as shown in Figure 4.9b. Unfortunately, the nonlinear relationships between colour channels produce poor-quality spectral band images. Instead, we synthesize multispectral images for spectral reconstruction evaluation by demosaicing the Green channels

Algorithm 12 Camera multispectral sensitivity calibration

1: Load a set of images for calibrating scaling factors between colour channels. ▷ We used all of our images.
2: Compute the magnitude of the gradient of each image, and find its median value. Retain only pixels with gradient magnitudes less than the median.
3: Further reduce the set of pixels to those with values less than 95% of the saturation value
4: **for** pixels from each narrowband-filtered illumination **do**
5: Create a set of pairs of Green pixels and their adjacent Red pixels
6: Use the first principal component of the pairs of pixels (computed from the pixels expressed as points in Green-Red space), to obtain a best-fit scaling factor between the Green and Red colour channels for the spectral band corresponding to the narrowband-filtered illumination
7: Repeat Steps 5 and 6 for the Green and Blue channels
8: **end for**

Figure 4.10: Our procedure for calibrating the relative sensitivities of different colour channels to light filtered by each optical bandpass filter. The images used for calibration are high-dynamic range images, produced as described in Section 4.2.3



(a) (b)

Figure 4.11: We observed wide variations in the relationships between different colour channels in textureless regions of our images. (a) A scatterplot of intensities in neighbouring Red and Green pixels in images captured under light filtered by the 650 nm optical bandpass filter. The two colour channels show a linear relationship. (b) A scatterplot of intensities in neighbouring Red and Green pixels in images captured under light filtered by the 550 nm optical bandpass filter. In this setting, the relationship between the two channels is highly nonlinear. Scatterplots such as shown in this figure are the basis for the scaling factors computed in Line 6 in Algorithm 12.

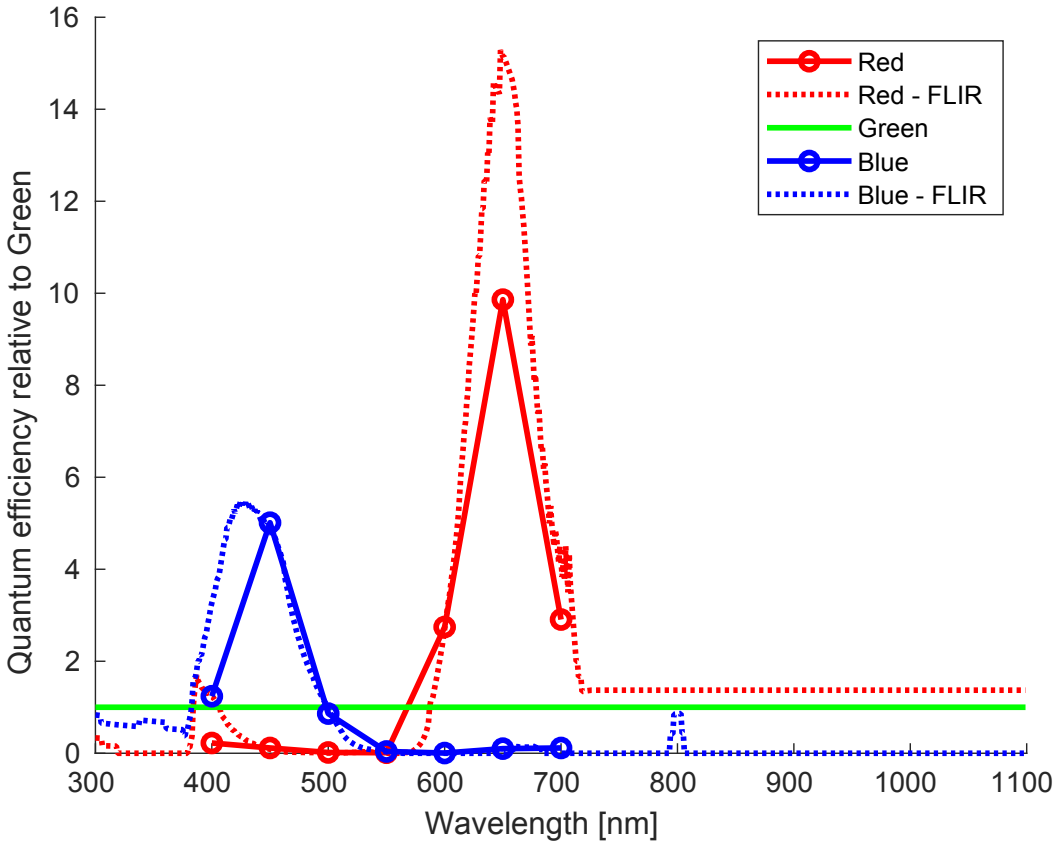


Figure 4.12: A comparison of the spectral sensitivity of a typical Sony ICX655, 2/3" image sensor measured by FLIR [32] (dotted lines) with our measurements of the relative spectral response between colour channels of our Point Grey BlackFly Flea3 camera with a Computar 07I lens (solid lines). We calibrated our relative spectral responses as described in Algorithm 12.

of the images captured under bandpass-filtered illumination. Doing so results in spectral band images with less noise, such as shown in Figure 4.9c. Unfortunately, these multispectral images no longer have a simple linear relationship with RGB images produced by directly summing mosaiced images from the different filtered illuminations (*e.g.* Figure 4.8b). Therefore, if we used them to evaluate spectral reconstruction results from such images, we would not be able to distinguish error introduced by the different image formation model from spectral reconstruction error.

To allow for quantitative evaluation of spectral reconstruction, although at the expense of using an oversimplified image formation model, we synthesize input images for spectral reconstruction algorithms by collapsing our demosaiced Green channel images (Figure 4.9c) to RGB. We use the spectral sensitivities in Figure 4.12, which are normalized relative to the spectral sensitivity of the Green channel, to synthesize the Red and Blue channels. As the Green channel spectral sensitivity is unity at all wavelengths, the Green channel of the synthetic raw images is simply the unweighted sum of the demosaiced Green channel images.

We restate our multispectral and corresponding RGB image synthesis procedure for reference in Algorithm 13. An example of a synthesized RGB image is provided in Figure 4.8c. It is visually similar to Figure 4.8b, illustrating that interesting behaviours in the spectral domain are not visible in RGB images.

As presented in Algorithm 13, we use bilinear interpolation to demosaic the Green channel of the images captured under filtered light. We chose to use bilinear interpolation as the demosaicing method, because the Green channel has a high sampling density, so even a naive demosaicing approach will produce a reasonable result. Furthermore, sophisticated demosaicing algorithms were designed to operate on images captured under wideband illumination, and leverage relationships between colour channels that may not exist under narrowband-filtered illumination.

Algorithm 13 Multispectral and RGB image synthesis

1: **for** each scene in the dataset **do**
2: Load the images for all bandpass-filtered illuminations
3: Compute values for the Green channel at all pixels in each image by bilinear interpolation
4: Output the resulting set of Green channel images as a multispectral image
5: Compute Red and Blue channels for each Green channel image by applying the scaling factors calibrated using Algorithm 12 (shown in Figure 4.12)
6: Sum these RGB images to produce the output RGB image
7: Subsample the output RGB image according to the sensor's colour-filter array to produce an output raw image to be input to image reconstruction algorithms
8: **end for**

Figure 4.13: Our procedure for generating multispectral images and corresponding synthetic RGB images from images captured under narrowband-filtered illuminations. The images loaded in Line 2, which are converted into multispectral and RGB images, are high-dynamic range images, produced as described in Section 4.2.3

4.3 Synthetic Experiments

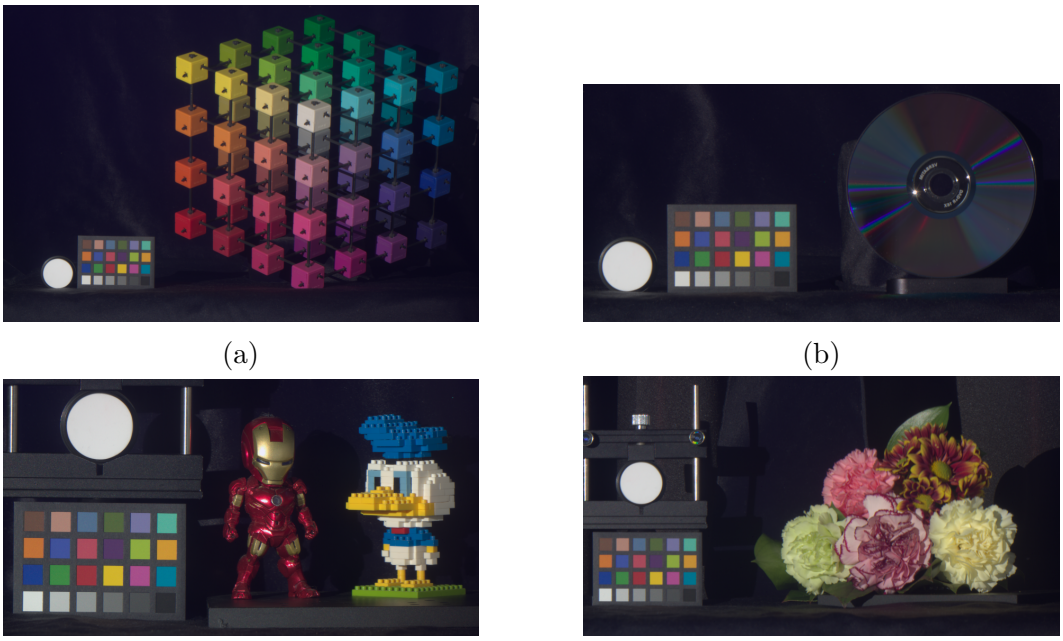
In our synthetic experiments, we follow the approach of other authors, described in Section 4.1.1: We take true spectral images, and convert them to mosaiced RGB input images for image reconstruction algorithms, according to the model of RGB image formation discussed in Section 3.3.1. We have not added noise or blur to the images, because the spectral images already contain such artifacts, and because we feel that experiments on real RGB images are better tests of robustness to imaging artifacts.

4.3.1 Datasets

KAIST Dataset

We used two datasets of images for our synthetic experiments. The first is the KAIST spectral image dataset [19], which we selected because its images were captured by a monochromatic camera, not by a line-scanning spectral camera, and so should have properties more similar to those of natural RGB images. Furthermore, as mentioned by the authors, the images are of higher-resolution, and have less blur, relative to images from many other datasets. The KAIST dataset contains 30 images of size 3376×2704 pixels, with 31 spectral bands at 5 nm intervals from 420 nm to 720 nm. All scenes were illuminated using a Xenon light source, and spectral band images were captured using a liquid crystal tunable filter. Although the camera had an apochromatic lens, chromatic dispersion is visible in the images (Figure 4.2). We suspect that the colour fringes are primarily caused by the tunable filter, because they are vertically oriented, whereas most lenses will produce chromatic aberration with radial symmetry about the image centre.

The images are provided as reflectance images, having been normalized by the reflectance of a Spectralon white reference, whereas we wish to evaluate spectral radiance reconstruction. Therefore, we convert the images to spectral radiance images by multiplying them with the spectral power distribution of a D65 illuminant, which we retrieved from the colour resource website of Lindbloom [68]. Next, to convert the spectral radiance images to RGB, we



(a) (b)

(c) (d)

Figure 4.14: A subset of the images from the KAIST dataset, cropped to exclude some of the dark background. We used these images for our synthetic experiments on the KAIST dataset.

use the spectral sensitivities of a Nikon D5100 DSLR (Figure 3.5), measured by Darrodi *et al.* [22], as the simulated camera.

While Choi *et al.* were careful to capture a wide variety of spectral reflectances in their dataset [19], the dataset is devoid of objects with mid-frequency textures. As such, the scenes are ideal for evaluating spectral radiances of small regions, but cannot give us a clear idea of how our method performs on image structures with a variety of spatial frequencies and edge strengths. In addition, large portions of each image are occupied by dark background drapery. Consequently, we feel there is little value in processing all images in the dataset, and that image-wide error statistics are not very meaningful. Instead, we chose a subset of four images from the dataset, and cropped them to remove some of the background drapery. The resulting small dataset that we used for our simulated experiments is shown in Figure 4.14. We selected the four images in an attempt to capture the variety present in the dataset, while keeping our experiments manageable in duration.

Our Dataset

The second dataset we used for synthetic experiments is our own dataset of multispectral images, created as described in Section 4.2.4. While our multispectral images have low spectral resolution, they are relatively sharp, have high spatial resolution (2448×2048 pixels), and include a variety of texture frequencies. More importantly, we captured images of a disk pattern in order to calibrate models of lateral chromatic aberration, so we can evaluate residual chromatic aberration in reconstructed images.

In total, we captured images of seven scenes, shown in Figure 4.15. Two scenes (Figures 4.15a and 4.15e) contain only the disk pattern used for calibrating lateral chromatic aberration. We pooled disk keypoint locations from both scenes when fitting models of lateral chromatic aberration. Doing so proved to be important for robust cross-validation or estimation of the spline smoothing parameter (Section 3.2.4), because it provided multiple keypoints in each local region of the image plane.

Two other scenes contain the X-Rite 24-patch ColorChecker CLASSIC colour calibration chart, the patches of which have standard tristimulus values that are used for colour correction. We also used the patches for spectral reconstruction evaluation, according to the method described in Section 4.4.1.

Finally, there are three scenes for evaluating algorithm performance on challenging imaging scenarios. The scene in Figure 4.15d tests recovery of fine details, such as the images on the book cover, and is especially useful for evaluating demosaicing algorithms. The scene in Figure 4.15f is an interesting challenge for chromatic aberration correction algorithms, because it contains refractive objects whose spectral dispersion effects should be preserved when the spectral dispersion of the camera lens is corrected. The scene in Figure 4.15g contains both sharply focused regions and defocused, or refractively distorted regions. This last image can be used to evaluate deblurring algorithms and the robustness of other algorithms to blur, and to unusual light propagation through the scene.

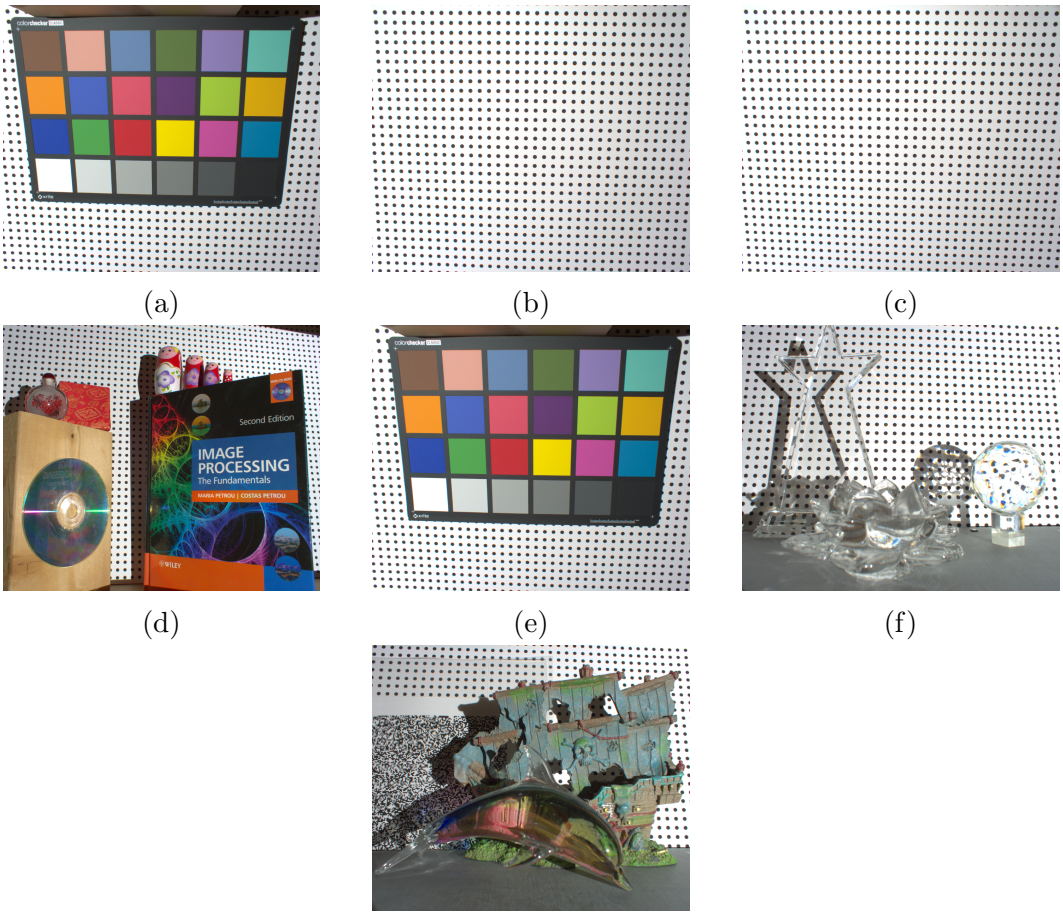


Figure 4.15: Our dataset for image reconstruction evaluation that we collected using our experimental apparatus, described in Section 4.2.1, with our BlackFly Flea3 colour camera, and a Computar 07I lens. The high-dynamic range images shown here were captured under unfiltered light, and have been colour-corrected with respect to the ColorChecker chart in (e) using a colour homography [28, 31]. The RGB images we simulated from images captured under narrowband-filtered illuminations appear similar to these images, but have different colour casts owing to the spectrally-varying vignetting effect of the optical bandpass filters. We use the former set of images for real experiments, as they were captured directly (although they were subject to exposure blending). We use the latter set of images for synthetic experiments, as they correspond directly to multispectral images that can be used for spectral reconstruction evaluation.

4.3.2 Limitations

Simulated images always differ from natural images, but there is value in determining the most important differences to understand how they might affect image reconstruction algorithms. We believe the greatest difference arises from assuming that RGB images are linearly related to spectral images. We base this conjecture on the nonlinear relationships between colour channels that we observed in images taken under narrowband-filtered illumination (Section 4.2.4). Unfortunately, the assumption of linearity is built into our spectral reconstruction algorithm. The same assumption is also the basis for the simulated spectral-RGB image pairs used to train most data-driven spectral reconstruction algorithms in literature, such as in the NITRE challenge [5]. Therefore, we might expect most spectral reconstruction methods to have inflated accuracy on synthetic input images, which has been observed by other authors in practice, as mentioned in Section 4.1.1.

With respect to chromatic aberration correction, our experiments are limited by the presence of chromatic aberration in all sources of data: both images we captured under narrowband-filtered illumination, and images from other datasets (*e.g.* Figures 4.1 and 4.2). Moreover, published datasets do not contain images of disk patterns for high-precision calibration of chromatic aberration (Section 3.2.1). In fact, spectral image datasets, including the KAIST dataset [19], tend to focus on objects with large textureless regions, so it would also be difficult to calibrate models of chromatic aberration using image registration (Section 3.2.3).

4.4 Quantitative Evaluation

Our experiments provide us with several points for evaluation: We can evaluate spectral images reconstructed from the input images, or we can evaluate RGB images, whether they were reconstructed directly, or are projections of reconstructed spectral images. In both cases, we can evaluate similarity with respect to reference images, or we can specifically evaluate chromatic aberration correction. Previous works have proposed a variety of different evaluation

metrics for these purposes, which we compare with the evaluation metrics we selected.

4.4.1 Spectral Error

Spectral radiances can be represented as vectors in a high-dimensional space of wavelengths, and therefore distance measurements can be computed bewteen them to quantify spectral reconstruction error.

Mean Relative Absolute Error (MRAE) is perhaps the most strongly-favoured error metric, because it was used in the NTIRE 2018 challenge [5]. If b indexes spectral bands, x indexes pixels, and M is the number of values (the product of the number of pixels, N, and the number of spectral bands, B) in the image, then the MRAE of the estimated image $\hat{I}$ with respect to the true image I is defined as [5]:

$$MRAE = \frac{\sum_x \sum_b \left| \frac{I_{x,b} - \hat{I}_{x,b}}{I_{x,b}} \right|}{M} \tag{4.3}$$

MRAE is intended to weight relative errors in intensity equally, regardless of the absolute intensity values. The human visual system roughly senses proportional differences in light intensity, not absolute differences [102], which lends credibility to the MRAE metric. On the other hand, MRAE is problematic in cases where the denominator, $I_{x,b}$, becomes arbitrarily small.

In contrast with MRAE, mean spectral Root Mean Squared Error (RMSE) gives equal weight to absolute differences in intensity [5]. While it may be less perceptually-relevant, it corresponds to a measure of likelihood from an isotropic Gaussian noise model, and so is meaningful from a signal processing perspective:

$$RMSE_{\text{pixel}} = \frac{\sum_x \sqrt{\frac{\sum_b \left(I_{x,b} - \hat{I}_{x,b}\right)^2}{B}}}{N} \tag{4.4}$$

Rather than computing the RMSE between individual spectral radiances, we can also express it as a global measure, comparing two images as though they were single high-dimensional quantities. This alternate method of calcu-

lating RMSE was presented by Nguyen *et al.* [84]:

$$RMSE_{\text{global}} = \sqrt{\frac{\sum_x \sum_b \left(I_{x,b} - \hat{I}_{x,b}\right)^2}{M}} \tag{4.5}$$

Nguyen *et al.* [84] also describe the spectral Goodness of Fit coefficient (GOF):

$$GOF = \frac{1}{N} \sum_x \frac{\left|\sum_b I_{x,b} \hat{I}_{x,b}\right|}{\sqrt{\left(\sum_b \hat{I}_{x,b} \hat{I}_{x,b}\right) \left(\sum_b I_{x,b} I_{x,b}\right)}} \tag{4.6}$$

The inverse cosine of the GOF is the Spectral Angle Mapper metric [62]. In our evaluations, we use the GOF instead of the Spectral Angle Mapper, because the GOF is conveniently in the interval $[0, 1]$. The GOF has the same limitation as MRAE: It becomes unreliable at very low intensities.

Lastly, treating spectral images as stacks of greyscale images, instead of as collections of spectral vectors, we can compute the mean Structural Similarity index (SSIM) [123] across spectral bands:

$$SSIM_{\text{mean}} = \frac{\sum_b SSIM\left(I_b, \hat{I}_b\right)}{B} \tag{4.7}$$

Image Spectral Registration

For synthetic experiments, where we have reference spectral images from which the input images for spectral reconstruction were created, the above error metrics can be computed immediately between the reference and reconstructed images. For real experiments, in contrast, if we have reference spectral data, it has been obtained using a sensor other than the camera used to capture the input raw RGB images. In this case, we need to register the reconstructed images with the reference data before computing spectral error metrics.

The most obvious spectral registration procedure follows from the most common form in which spectral data is expressed: As relative to some reference spectral radiance. By dividing an image by the measured spectral radiance of a reference object, and then multiplying the image by the known spectral reflectance of the object, one converts a spectral radiance image into an effective reflectance image [33]. Effective reflectance images can be compared, even

when generated by different spectral imaging devices, as they are independent of the spectral sensitivities of the devices, and of global illuminants.

Unfortunately, the assumption of a constant illuminant means that shading changes are indistinguishable from spectral reflectance changes in effective reflectance images [33]. This ambiguity is especially problematic for the multispectral images we created using our apparatus, as they are subject to spectrally-varying spatial illumination non-uniformity. Also, spectral reconstruction error in the reconstructed reference patch used to compute the effective reflectance image will be propagated to the rest of the image.

To avoid the problems associated with effective reflectance images, we use a different spectral registration procedure. We apply band-wise scaling factors that globally align two sets of spectral measurements. (The sets of measurements could represent spectral images, or pointwise spectral measurements.) Specifically, we solve the following optimization problem:

$$\mathbf{S}^* = \underset{\mathbf{S}}{\operatorname{argmin}} \sum_x \sum_b \left(I_{x,b} - S_b \hat{I}_{x,b} \right)^2 \tag{4.8}$$

where x indexes spectral measurements or pixels, b indexes spectral bands, and $\hat{\mathbf{I}}$ is the reconstructed (estimated) data corresponding to the reference data, $\mathbf{I}$. $\mathbf{S}^*$ is the vector of scaling factors for individual bands which minimizes the root-mean-squared error between the reference and estimated spectral data. Problem (4.8) has the solution:

$$\mathbf{S}^* = \begin{bmatrix} \left(\sum_x I_{x,1}\hat{I}_{x,1}\right)\left(\sum_x \hat{I}_{x,1}\hat{I}_{x,1}\right)^{-1} \\ \left(\sum_x I_{x,2}\hat{I}_{x,2}\right)\left(\sum_x \hat{I}_{x,2}\hat{I}_{x,2}\right)^{-1} \\ \vdots \\ \left(\sum_x I_{x,B}\hat{I}_{x,B}\right)\left(\sum_x \hat{I}_{x,B}\hat{I}_{x,B}\right)^{-1} \end{bmatrix} \tag{4.9}$$

4.4.2 Colour Error

In addition to spectral error metrics, we computed RGB error metrics. In the following error metric formulas, c denotes an image colour channel (Red, Green, or Blue), and N is the number of pixels. I and $\hat{I}$ denote the true and estimated colour images, respectively.

MRAE,

$$MRAE_c = \frac{\sum_x \left| \frac{I_{x,c} - \hat{I}_{x,c}}{I_{x,c}} \right|}{N} \tag{4.10}$$

RMSE,

$$RMSE_c = \sqrt{\frac{\sum_x \left(I_{x,c} - \hat{I}_{x,c} \right)^2}{N}} \tag{4.11}$$

SSIM [123],

$$SSIM_c = SSIM \left(I_c, \hat{I}_c \right) \tag{4.12}$$

The above RGB error metrics are analogs of the spectral error metrics given in Section 4.4.1. They are by no means a complete list of error metrics, however. In particular, we omitted peak signal-to-noise ratio (PSNR) and colour peak signal-to-noise ratio (CPSNR), popular error metrics in demosaicing literature [79]. In our view, PSNR and CPSNR are redundant because they are closely-related to RMSE. PSNR is problematic, because it is difficult to define an appropriate peak value when evaluating high-dynamic range images. We find CPSNR even more problematic, as, while it is convenient to obtain one number for all colour channels, different colour channels are incomparable.

A better way to evaluate error across colour channels would be to use error metrics defined in standard colour spaces. Error metrics such as CIE ΔE are designed to be perceptually relevant [31]. Unfortunately, our evaluation according to such metrics would be affected by errors in colour space conversion. Therefore, we evaluate our RGB error metrics in the colour space of the raw camera responses, not in standard colour spaces.

4.4.3 Residual Chromatic Aberration

Motivation

The preceding error metrics are useful for synthetic experiments, where ground truth images are available. Unfortunately, they do not indicate whether the estimated images have been adequately corrected for chromatic aberration, especially if the ground truth images contain chromatic aberration. To evaluate chromatic aberration, we need metrics that do not require reference images.

If we had a metric for measuring chromatic aberration without a reference image, such as the image quality metric of Lamb and Khambete [63], presumably we could perfectly correct chromatic aberration by using the metric as an objective function. We argue that no such metric is valid, however, because if a scene contained rainbow patterns, they could be mistaken for chromatic aberration. The same problem would arise if one captured an image of a display of an image containing chromatic aberration. In general, we cannot rely on the image data alone to assess or correct chromatic aberration. We should instead say that we are interested in correcting images for physical chromatic aberration, as opposed to correcting colour fringes in images. In contrast to our method, uncalibrated (purely image-based) methods for correcting chromatic aberration do not distinguish between the two tasks.

To measure chromatic aberration, not colour fringing, we need a way to determine which colour fringes are caused by chromatic aberration. We can do so given some knowledge of the scene being imaged. In other words, instead of an image that provides pixel-level ground truth, we infer approximate ground truth from known scene characteristics. The most common approach used in literature is to take images of achromatic scenes, and then measure achromaticity in the image. For example, Chang *et al.* [18] use the Variance of Colour Differences metric, which is the mean squared difference between Red or Blue intensities and their corresponding Green intensities. We have not followed their approach, as achromaticity is a perceptual property of colours, and does not have a well-defined spectral analog, given that human vision has low spectral resolution. We seek a metric that applies to both colour images and spectral images. On this point, Rudakova and Monasse [99] took a more flexible approach by measuring the spreading of colours in a local image region away from a local achromatic line in RGB space. Their approach is valid for any regions that contain only mixtures of two spectral power distributions. One example of such a region is a single spectral reflectance subject to illumination with varying attenuation from shading. Another example is an edge blending two different spectral reflectances, under a constant illuminant. Combinations of two or more spectral reflectances with varying illumination

will produce a colour distribution which deviates from a line in RGB space or in the space of spectral power distributions, however, so their evaluation technique will not work in general. Furthermore, their evaluation technique does not account for errors in the local colour line itself, and so will not penalize chromatic aberration correction methods which make global changes to image colours, such as desaturating colours.

Aside from colour analysis, chromatic aberration can be measured using geometrical information. For instance, Helou *et al.* [45] designed a calibration pattern with a slanted edge for measuring longitudinal chromatic aberration using edge spread functions. From edge spread functions, they obtained point spread functions by assuming the point spread functions have symmetrical Gaussian forms. They then compared point spread function radii between colour channels as the calibration target was moved to different depths relative to the camera. In contrast, Rudakova and Monasse [99] measured longitudinal chromatic aberration using residual misalignment between disks detected from their disk calibration pattern. Their evaluation approach is therefore much simpler, and is also more relevant to our work, as we have not yet included longitudinal chromatic aberration in our image formation model. Unfortunately, as we use the disk calibration pattern approach from Rudakova and Monasse [99] to calibrate our models of chromatic aberration, our optimization problem (3.1) is directly minimizing such a measure of chromatic aberration. We need an evaluation criteria which is independent of our method.

Edge-Based Measurement of Residual Chromatic Aberration

Our proposed method for evaluating chromatic aberration correction relies on a scene with sharp edges bordering homogenous regions. In contrast to the works mentioned above [45, 99], it does not require a specific calibration pattern, and can operate on colour images or spectral images. It uses a synthetic image to provide ground truth pixel intensities for the regions bordering the edges, created by sampling pixels deeper inside the homogenous regions. In our experiments, we used our X-Rite ColorChecker chart as a scene suitable for creating the synthetic image.

Our evaluation procedure consists of several steps. First, we sample the colours in the centers of the ColorChecker patches in a captured image of the scene. As we assume the centers of the patches have little colour variation, we can sample their colours without demosaicing. From these samples, we create a synthetic colour image in which each colour sample is replicated to the entire area of the corresponding patch. We also measured the spectral reflectance of each patch (Section 4.6.1) to create a synthetic spectral image.

The contours of each patch in the synthetic images are those that we marked by hand, which was possible given the simple geometry of the image. We took care to use the appropriate images for labelling the contours, to account for dispersion: For evaluating direct colour image estimation (Section 3.9.1), we marked the contours in the green channel of the image captured under unfiltered light. For evaluating spectral image estimation, we marked the contours in the image captured under light filtered by the 600 nm bandpass filter. In either case, the edges we marked are in the locations that they should be after chromatic aberration correction, because we chose the Green channel, or the 600 nm spectral band, as zero dispersion points in our colour or spectral models of dispersion, respectively.

Next, to quantitatively evaluate residual chromatic aberration, we measure the error between estimated images and the synthetic images in narrow regions (12 pixels wide) inside the patches, abutting the hand-marked contours. We use the same error metrics as listed in Sections 4.4.1 and 4.4.2. When computing spectral error metrics, we first register the true and estimated images in the spectral domain using per-spectral band scaling factors which best align the images. To treat all ColorChecker patches equally, we compute the scaling factors using a modified version of (4.8):

$$\mathbf{S}^* = \underset{\mathbf{S}}{\operatorname{argmin}} \sum_x \sum_b w_x \left(I_{x,b} - S_b \hat{I}_{x,b} \right)^2 \tag{4.13}$$

where w_x is a weight with a magnitude inversely proportional to the number of pixels in the ColorChecker patch containing the pixel with position x.

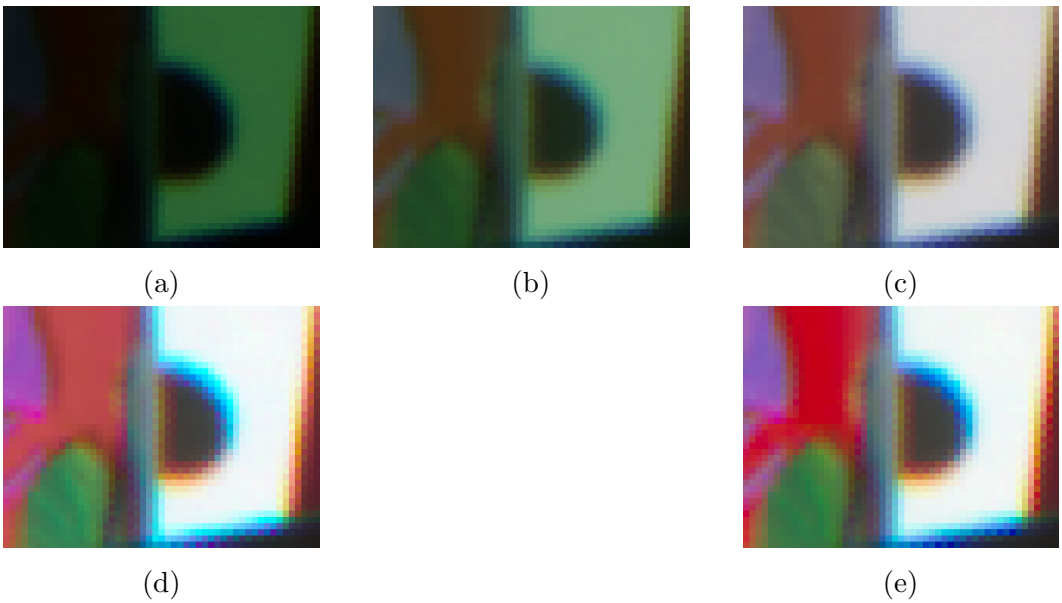
Problem (4.13) has the solution:

$$\mathbf{S}^* = \begin{bmatrix} \left(\sum_x w_x I_{x,1}\hat{I}_{x,1}\right)\left(\sum_x w_x \hat{I}_{x,1}\hat{I}_{x,1}\right)^{-1} \\ \left(\sum_x w_x I_{x,2}\hat{I}_{x,2}\right)\left(\sum_x w_x \hat{I}_{x,2}\hat{I}_{x,2}\right)^{-1} \\ \vdots \\ \left(\sum_x w_x I_{x,B}\hat{I}_{x,B}\right)\left(\sum_x w_x \hat{I}_{x,B}\hat{I}_{x,B}\right)^{-1} \end{bmatrix} \tag{4.14}$$

Note that we optimize problem (4.13) over the entire area of all patches, not only over the patch border regions.

In our evaluation, we do not assess regions outside the patches, because these regions do not have standardized colours, in contrast to the patches of the ColorChecker. We do assess the central regions of the patches, however, to verify that the error is lower far from image edges, as expected. As another sanity check, we can evaluate the chromatic aberration present in the captured colour image, to see if it is higher than the chromatic aberration present in our reconstructed images. Note that we can evaluate the captured colour image without demosaicing, by evaluating each colour channel at the appropriate locations in the colour-filter array.

The colour and spectral errors we measure account for both lateral and longitudinal chromatic aberration, as well as other image estimation artifacts, such as oversmoothing or ringing. Furthermore, they also flag chromatic aberration correction methods which tightly match colour channels or spectral bands, but do so at the cost of colour desaturation. Of course, the error measurements are also affected by our having synthesized the reference images. Therefore, to make the synthetic images as realistic as possible, rather than assuming the ColorChecker patches are perfectly homogenous, we apply a non-uniform shading pattern that we calibrate from the Green channel of the colour image of the scene. We fit the model of shading variation to the homogenous portions of the frame surrounding the colour checker patches, using the same process as described in Section 3.2.2.



(a) (b) (c)

(d) (e)

Figure 4.16: An image region subject to different colour processing methods: (a) The raw colour channel responses of our BlackFly Flea3 colour camera to a scene lit by our Optoma EP739 DLP projector, reinterpreted as sRGB colours. (b) The image subject to the sRGB gamma correction. (c) The image subject to white-balancing, followed by sRGB gamma correction. (d) Colours obtained by the root-polynomial regression technique of Finlayson *et al.* [29]. (e) Colours obtained by the colour homography technique of Finlayson *et al.* [31].

4.5 Visual Evaluation

When correcting chromatic aberration to reduce the perceivable artifacts of an imaging system, human observers must evaluate the results. A formal user study would provide the most comprehensive evaluation, but is beyond the scope of our work. We would need a significant number of participants to evaluate results on a wide variety of cameras and lenses before suggesting a particular algorithm be used as a general method for image quality enhancement, because image quality depends on many physical, computational, and perceptual processes. As an example showing why visual assessment is challenging, consider Figure 4.16.

In Figure 4.16, the same image is presented using different colour mapping algorithms. The original image (a) contains colour fringes from chromatic aberration, but they are barely visible. After gamma correction (b), colour

fringes are visible, but the colours of the image are unnatural. With both white-balancing, using a Bradford cone response model [67], and gamma correction, in (c), the colours of the image seem more natural, but are desaturated. White balancing only corrects achromatic portions of the scene, however, in contrast to full colour balance methods [57]. In (d) and (e), we show the results of the full colour balance methods of Finlayson *et al.* [29] and Finlayson *et al.* [31], respectively, both of which improve colour saturation, but consequently exaggerate the colour fringes.

The colour processing applied to the image, which depends on scene illumination, the spectral response of the camera, and user preferences, affects our choice of image restoration algorithm. Referring again to Figure 4.16, with a stronger gamma correction, we might favour algorithms which perform well in the darker regions of an image. If the colour correction increases saturation, then small differences between colour channels are amplified, and we might prefer algorithms which penalize misregistration between colour channels, even if they cause some colour desaturation, or oversmooth edges.

To produce colour-corrected results in this work, we selected the colour homography full colour balance method of Finlayson *et al.* [31]. There are many full colour balance methods, but we chose their method because it is recent, and because its results differed only slightly from those of the popular root-polynomial method of Finlayson *et al.* [29]. To experiment with both methods, we used the implementations provided by Han Gong.[3] While white-balancing may yield colours with a simpler relationship with the camera's raw responses, we prefer full colour balancing because it would be used to produce images for human observers. We rely on our quantitative evaluation metrics to assess image reconstruction accuracy in the physically-relevant raw colour space of the camera, so there is no need to show viewers a compromise between raw and perceptually-corrected colours.

Although we used Han Gong's implementation of the colour homography method, we still needed to set up its input data, and then post-process its output. First, whereas a colour homography can be computed without correcting

[3]https://github.com/hangong/ch

the colours in the image for non-uniform shading, we used the RANSAC-based variant of the method which takes shading-corrected colours for validation [31]. For shading correction, we used a polynomial model of intensity variation (Section 3.2.2), calibrated from the frame surrounding the patches of our ColorChecker colour chart. We then used the patches of the ColorChecker chart to fit the colour homography, excluding the white patch, because it was saturated at all exposures. Rather than using the standard XYZ colours of the ColorChecker chart as the target colours for homography computation, we synthesized XYZ colours for the patches from the measured spectral reflectances of the patches, according to the ASTM E308 standard [6]. (The measured spectral reflectances are those we obtained using an Analytical Spectral Devices spectrometer, mentioned in Section 4.6.1.) Following colour balancing, we apply a chromatic adaptation step to convert the XYZ colours, which were effectively measured under an equal energy radiator, to the D65 whitepoint, before computing sRGB colours for display [67].

All colour correction methods that operate on colour channels are approximate, since true colours can only be generated from spectral images. Regardless, we obtain colour-corrected images corresponding to our estimated spectral images by first converting the spectral images to the raw colour space of the camera, and then applying the same colour correction that we use on RGB images. We do not synthesize corrected colour images from spectral images by directly mapping spectral radiances to the XYZ colour space, for three reasons: First, we do not have a uniform reflectance standard to provide an accurate estimate of the illuminant whitepoint to use during XYZ to sRGB colour space conversion. Second, we wish our colour-corrected results to reflect accuracy in the raw colour space of the camera, as our image estimation algorithms attempt to minimize error in this colour space. Third, using the same colour processing pipeline for both RGB and spectral images facilitates visual comparisons between the two types of images.

4.6 Comparison Methods

We compare the results generated by our algorithm with data from several other sources. In our real experiments, we relied on spectrometry to provide additional spectral data for validation. In all of our experiments, we also ran image reconstruction algorithms developed by other authors to compare our work with the state-of-the-art.

4.6.1 Spectrometry

Analytical Spectral Devices point spectrometer

We measured the spectral reflectance of each of the 24 patches of our X-Rite 24-patch ColorChecker CLASSIC colour calibration chart at the Core-Net Spectral Laboratory,[4] with the assistance of Dr. Jilu Feng, and the permission of Dr. Benoit Rivard. An Analytical Spectral Devices point spectrometer with a fiber optic probe captured the spectral reflectance of each patch under illumination from a quartz halogen lamp. Each spectral reflectance is the average of 40 measurements, taken relative to a Spectralon tile that served as a white reflectance standard. The spectrometer has a spectral resolution of 1 nm and a range from 350 to 2500 nm. Given its high accuracy, we used it as ground truth data for evaluating chromatic aberration correction at the edges of the ColorChecker patches, as described in Section 4.4.3, and for evaluating spectral accuracy at the centers of the patches.

GoSpectro mobile spectrometer

The GoSpectro device, produced by Goyalab,[5] can be placed over a smartphone camera to disperse light entering a slit into a rainbow on the image sensor. The GoSpectro mobile software application can then extract spectral measurements from the images, after calibration. Calibration consists first of wavelength calibration, in which the user must manually mark the positions of the four strong spectral peaks of a fluorescent light. Second, spectral intensi-

[4] http://coresensing.net/index.html

[5] https://www.goyalab.com/product/hand-spectrometer-gospectro/

ties can then be calibrated from broad-spectrum light (*e.g.* sunlight) reflected from a white target. For our intensity calibration, we used the light from a halogen lamp reflected by the white patch of our ColorChecker chart, because sunlight produced overexposed images (even with autoexposure enabled in the GoSpectro app).

As the GoSpetro spectrometer is portable, we thought of using it to measure the spectral signatures of identifiable locations in the scenes we imaged inside our light-proof box (Section 4.2.1). Unfortunately, even though we used a fiber optic probe attached to the spectrometer to facilitate measuring the spectra reflected by small objects, we still could not hold the probe steady within the confines of our apparatus, without interfering with the light propagating throughout the space. Moreover, we measured the spectral reflectances of our ColorChecker's colour patches under a halogen light, for comparison with the Analytical Spectral Devices spectrometer's measurements. By comparing the two sets of measurements (Section 5.4.2), we determined that the GoSpectro setup does not have the accuracy or precision necessary to provide ground truth spectral measurements for our experiments.

4.6.2 RGB-to-Spectral Algorithms

Compact single-shot hyperspectral imaging using a prism

Our spectral reconstruction algorithm is inspired by the first stage of the method of Baek *et al.* [7]. They calibrated the spectral dispersion produced by a triangular prism attached to the front of the camera lens, and then formulated a three-stage global optimization algorithm to reconstruct spectral images from the RGB image captured by the camera.

We compare our method with elements of their method. To simulate the first stage of their method, we use its prior terms in our optimization problem, and omit our non-negativity constraint. The resulting optimization problem corresponds to the `L1L1` condition in Table 3.1, and is provided below:

$$\mathbf{I}^* = \underset{\mathbf{I}}{\operatorname{argmin}} \|\mathbf{M}\left(\mathbf{\Omega}\left(\mathbf{\Phi}\left(\mathbf{B}\left(\mathbf{I}\right)\right)\right)\right) - \mathbf{I}_{\text{raw}}\|_2^2 + w_1 \left\|\nabla_{xy}\mathbf{I}\right\|_1 + w_2 \left\|\nabla_\lambda \nabla_{xy}\mathbf{I}\right\|_1 \quad (4.15)$$

We selected the weights in problem (4.15) automatically, as described in Section 3.6, whereas they used fixed values of $w_1 = 1 \times 10^{-5}$ and $w_2 = 1 \times 10^{-1}$.

In evaluating their prior terms, it is important to note that there are key differences in our data-fitting term. First, our data-fitting term projects the spectral image to a raw mosaiced image, whereas it is unclear whether they demosaiced the image as a preprocessing step. They would have needed some form of demosaicing, assuming their image sensor had a colour-filter array, in order to shrink their images to the 600×860 resolution they used for image reconstruction, as this resolution is not related to the original resolution by integer scaling factors (therefore precluding pixel subsampling for image downscaling). Second, they estimate images with 31 spectral bands, at 10 nm increments from 400 nm to 700 nm, whereas, in our synthetic experiments, we use 8 equally-spaced spectral bands in the range from 418 to 667 nm (Section 3.3.2), and use Gaussian interpolation between spectral bands (Section 3.3.3).

Our intention was to determine if their spectral-spatial prior term, the second prior term in (4.15), benefits spectral reconstruction. Unfortunately, we can conclusively evaluate the prior term only in simulated experiments, by selecting the weights w_1 and w_2 that minimize the error with respect to the true spectral image.

High-quality hyperspectral reconstruction using a spectral prior

Another prior term we chose to evaluate is the data-driven prior term developed by Choi *et al.* [19]. Their prior term is a convolutional autoencoder, trained on spectral images from published datasets [17, 128]. We retrieved their trained model, and the implementation that they provided of their algorithm.[6] We then modified their implementation so that it performed RGB-to-spectral reconstruction as opposed to spectral reconstruction from coded aperture snapshot spectral imaging data, as described in Section 3.8.2. With respect to their parameters, we used a weight of 1×10^{-8} on their spatial regularization term, following their description of using their method for de-

[6] https://github.com/KAIST-VCLAB/deepcassi

mosaicing, and left their other parameters with default values.

We wanted to compare our method with data-driven spectral reconstruction methods, and selected their method among others because their convolutional autoencoder serves as an image prior, in contrast to the learned mappings from low- to high-spectral resolution images presented in other works. The advantage of their approach is that their autoencoder does not need to be re-trained following changes to the relationship between spectral images and RGB input images [19]. Furthermore, their method can be adapted to accept mosaiced RGB images as input. For these reasons, their method may be more useful in practice than many data-driven spectral reconstruction methods. Nevertheless, we could only evaluate their data-driven prior in synthetic experiments, because it is trained on reflectance data. In synthetic experiments, we can include an illuminant spectral power distribution mapping spectral reflectances to spectral radiances in the data-fitting term of their optimization problem, whereas in our real experiments, we did not have the equipment needed to characterize the scene illumination.

4.6.3 Demosaicing Algorithms

While plain demosaicing was not our primary objective, it is useful to compare our results with those of demosaicing algorithms to determine whether our method results in similar artifacts and comparable image sharpness.

Bilinear interpolation

Bilinear interpolation is one of the most basic demosaicing algorithms mentioned in literature. Missing pixel values in a given colour channel are computed by bilinear interpolation of the four nearest known pixels. In a multichannel image, bilinear interpolation normally produces suboptimal results by ignoring correlations between colour channels [125]. When these correlations are absent, however, such in areas of highly saturated colours [126], demosaicing algorithms that rely on them fail. Therefore, we selected bilinear interpolation as a robust baseline algorithm for demosaicing.

Adaptive Residual Interpolation (ARI)

Adaptive Residual Interpolation [79] is the most recent demosaicing algorithm in the family of residual interpolation algorithms for demosaicing. We selected this method for its demonstrated high performance on standard demosaicing datasets, and also because it does not rely on training data, so it may perform well on unusual images, such as the RGB images we simulate from multispectral images (Section 4.2.4). In fact, the authors have evaluated Adaptive Residual Interpolation on multispectral colour-filter arrays [79], suggesting that the algorithm has the flexibility we desire. We use the authors' implementation of Adaptive Residual Interpolation,[7] with their default parameter settings, in our experiments.

4.6.4 Chromatic Aberration Correction Algorithms

Precise correction of lateral chromatic aberration in images

Rudakova and Monasse [99] described the technique for modelling lateral chromatic aberration using disk keypoints interpolated by bivariate polynomials that we adapted to model spectral dispersion (Section 3.2). We implemented their technique for correcting lateral chromatic aberration, which is similar to our direct colour image estimation algorithm (Section 3.9.1), except that it warps the image channels as opposed to reconstructing them by solving a global optimization problem. Their description of their method does not specify the type of interpolation they used to resample the image during warping, although the interpolation method is important, as it determines the effective demosaicing algorithm applied to the input mosaiced image. Therefore, we tested demosaicing with either of the two demosaicing methods listed above (bilinear interpolation, and adaptive residual interpolation) before image warping. We hypothesize that bilinear interpolation is closer to their intent, as bilinear interpolation for demosaicing followed by bilinear interpolation during image warping is equivalent to a single bilinear interpolation step when directly warping the mosaiced image to compute the full-colour warped image.

[7] http://www.ok.sc.e.titech.ac.jp/res/DM/RI.html

Cross-channel information transfer (CCT)

Sun *et al.* [111] presented an image-based method for correcting both lateral and longitudinal chromatic aberration. In contrast, our method, and the method of Rudakova and Monasse [99], both require calibration, and both correct only lateral chromatic aberration. We selected the method of Sun *et al.* [111] in order to compare our results with those from a recent, calibration-free, chromatic aberration correction method. Additionally, we chose their method because it does not rely on training data. Therefore, it may perform well regardless of the imaging environment we are using, as suggested by their high-quality results for both diffractive, and refractive, imaging systems [111].

Their method combines a blind deblurring preprocessing step [61], applied to the Green channel of the image, with subsequent alignment and deblurring of the Red and Blue channels with respect to the Green channel [111]. The resulting algorithm has many parameters, although only a subset of the parameters require tuning. For the blind deblurring step [61], we used the implementation provided by Krishnan *et al.*,[8] and left all parameters with their default values, except for the λ weight in their blur kernel estimation procedure, which we set to 850, and the blur kernel size, which we set to 9. We set the blur kernel size to be an upper bound on the displacements between spectral bands caused by spectral dispersion.

We selected the value of the λ parameter that minimized the residual chromatic aberration measured as described in Section 4.4.3. Specifically, we cropped the image of our ColorChecker chart (Figure 4.15e), captured under unfiltered light from our DLP projector, to contain only the ColorChecker chart. We corrected the image for vignetting (Section 3.2.2), calibrating the vignetting model using the ColorChecker chart's frame, and then used Adaptive Residual Interpolation [79] to demosaic the image. We then ran the blind deblurring algorithm on the Green channel of the resulting colour image, and evaluated the MRAE between its result, and the ideal image created by copying colours from the centers of the ColorChecker patches to their edges. (We

[8] https://dilipkay.wordpress.com/blind-deconvolution/

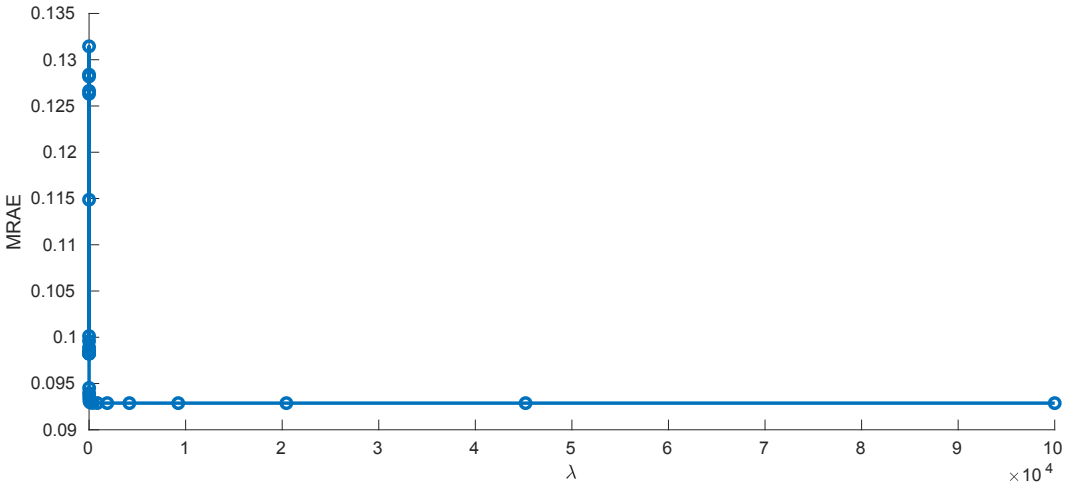


Figure 4.17: Residual chromatic aberration of the Green channel of an image processed by the method of Krishnan *et al.* [61], evaluated in terms of MRAE with respect to reference Green channel values. We selected 850 as the value of the λ parameter (of their blur kernel estimation step), corresponding to the "knee" of the plot, in our subsequent experiments.

only evaluated the MRAE within 12-pixel wide strips inside the borders of the patches.) The value of λ corresponding to the "knee" of the MRAE plot (Figure 4.17) was approximately 850. In other words, our method for choosing the λ parameter's value is inspired by the L-curve method [38].

Similarly, for the subseqent cross-channel information transfer step [111], we used the implementation provided by the authors.[9] In this case, there were two parameters to tune: The size of the overlapping windows used for inter-channel point spread function estimation, and the size of the overlapping windows used for cross-channel transfer. We used the same evaluation technique as described above to select values for these parameters, except that the MRAE is now evaluated over three colour channels, instead of only the Green channel.

As shown in Figure 4.18, the method of Sun *et al.* [111] performs best for a small cross-channel transfer window size, and a somewhat larger point spread function estimation window size. In fact, the rapid deterioration of performance with increasing cross-channel transfer window sizes suggests that the

[9] https://github.com/evanypeng/ICCV2017_RevisitCCIT_code

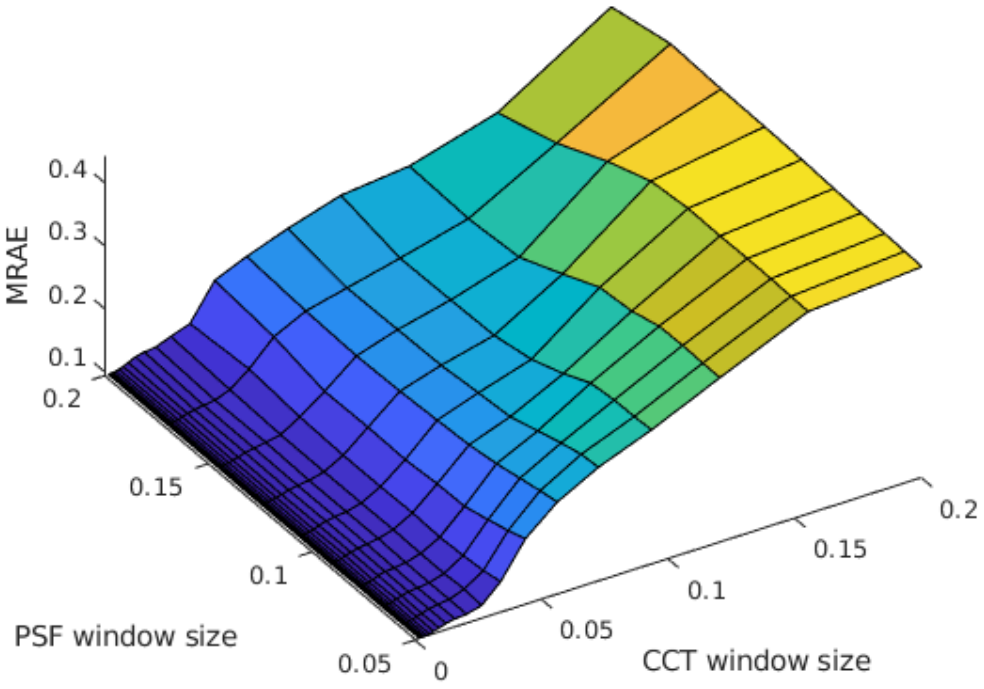


Figure 4.18: Residual chromatic aberration of an image processed by the method of Sun *et al.* [111], evaluated in terms of MRAE with respect to reference colours. From the evaluation, we selected a cross-channel transfer window size of 5 pixels, and a point spread function estimation window size of 0.17 times the larger dimension of the image. In this experiment, the image has dimensions 1455×2257 (cropped from a 2048×2448 image). The window sizes labelled in the figure are expressed as fractions of the largest of the (cropped) image's dimensions.

cross-channel transfer algorithm itself is sub-optimal. Their method is relatively insensitive to the point spread function estimation window size, whereas larger values of the cross-channel transfer window size result in colour desaturation, as they discuss in their article [111]. We are pleased that our evaluation technique detects this colour desaturation, rather than allowing it as a side-effect of chromatic aberration correction. Based on the results of our parameter tuning experiment, we chose a cross-channel transfer window size of 5 pixels, and a point spread function estimation window size of 0.17 times the larger dimension of the image sensor (usually on the order of 400 pixels).

Unfortunately, chromatic aberration varies with position on the image plane, whereas the implementation of the method of Krishnan *et al.* [61] estimates a spatially-invariant point spread function. When we ran the blind deblurring program on whole images, it tended to estimate very sharp point spread functions, suggesting that the average chromatic aberration over the entire image is negligible, likely because of the symmetry of lateral chromatic aberration, and because the Green channel is usually given priority during camera focusing. Rather than reimplement the blind deblurring algorithm to perform spatially-varying deconvolution, we simply omitted blind deblurring of the Green channel when running the cross-channel information transfer algorithm on whole images. For visual evaluations of small image patches, however, we show results both with and without the blind deblurring step. When generating results for small patches, we provide only the patches as input to the blind deblurring algorithm, so that it can compute an optimal point spread function for each patch individually.

Chapter 5

Results

5.1 Overview

In this chapter, we present the results of our image reconstruction experiments, on both synthetic and real input raw mosaiced RGB images. We compare image reconstruction algorithms, and suggest explanations for the differences between their results. We also evaluate our techniques for modelling dispersion. While we briefly review the context of each experiment, we focus on critical assessment of our experimental results, having described our experimental methods in detail in Chapter 4.

To our knowledge, we are the first to synthesize multispectral images using an RGB camera in combination with optical bandpass filters. Other authors have implemented multispectral imaging with either controlled illumination (*e.g.* Goel *et al.* [36]) or by filtering light entering the camera (*e.g.* Choi *et al.* [19]), but in either context, they have used monochromatic cameras. The novelty of our experiments warranted a more detailed description of our experimental methods, which we placed in a separate chapter from our results, to give our discussion a modular structure. The experiments we present in this chapter draw on different, but overlapping, content from the previous chapter (Chapter 4). For clarity, we now review the connections between the two chapters, and summarize the flow of ideas in this chapter.

Our first experiment relies on third-party spectral images, from the KAIST dataset (Section 4.3.1). We use this dataset to compare our image estimation method against other works tested on this dataset, specifically the method of

Choi *et al.* [19], and also to validate our image estimation method independently of our own dataset of images. As the images from the KAIST dataset are of higher spectral resolution than the images which we can obtain using our apparatus, we use the KAIST dataset to evaluate which variations of our method result in the highest spectral reconstruction accuracy.

The KAIST dataset lacks information concerning spectral dispersion, so our remaining experiments focus on our own captured images, where we can evaluate chromatic aberration correction. We then must evaluate both our models of spectral dispersion, and our image reconstruction method. In Section 5.3, we validate our models of dispersion by assessing whether they are robust to variations in our calibration procedure. In Section 5.4, we qualitatively and quantitatively assess chromatic aberration correction in RGB images captured under broadband illumination. We also evaluate spectral reconstruction accuracy relative to spectrometry data. Our novel experimental apparatus provides us with a more conclusive evaluation: Not only can we evaluate spectral reconstruction from RGB images, but we can also compare our multispectral images to the spectrometry data to provide a practical upper bound on spectral reconstruction accuracy.

Unfortunately, spectrometry data is only available in homogenous patches, but we seek to better understand how our method treats edges in the image, where we observe colour artifacts. Therefore, in Section 5.5, we evaluate chromatic aberration correction and spectral reconstruction in synthetic RGB images created from our multispectral images. In this setting, while the input images differ from true RGB images, we can evaluate spectral reconstruction at all image locations. We can also compare chromatic aberration correction using only our model of dispersion, in combination with the true multispectral images, to chromatic aberration correction from the synthetic RGB images using both our model of dipsersion, and our image reconstruction algorithm.

In Figure 5.1, we present a graphical summary of the preceding overview, explicitly showing the dependencies of our experimental results on portions of our experimental methods.

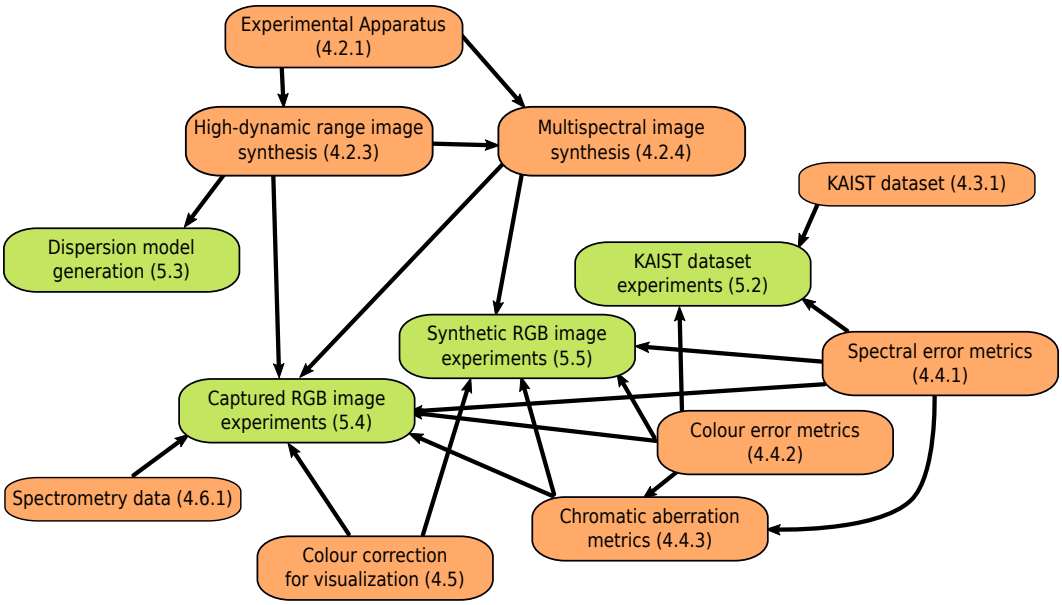


Figure 5.1: A graph of connections between our experimental methods (Chapter 4, orange shapes) and our experimental results (green shapes). Arrows indicate dependencies between topics.

5.2 Reconstruction of Images from the KAIST Dataset

In our first experiment on synthetic input images, we selected images from the KAIST dataset, as described in Section 4.3.1. We tested a large number of algorithms, including the method of Choi *et al.* [19], during the development of which Choi *et al.* created this dataset. Our intention was to narrow the selection of algorithms to be tested in subsequent experiments. As the dataset does not allow for reliable calibration of spectral dispersion, however, we were conservative in excluding algorithms from subsequent experiments, given that a model of dispersion may affect image estimation accuracy. Recall that, without a model of dispersion, our image estimation algorithms operate as described in Section 3.9.2.

5.2.1 Comprehensive Evaluation

In this section, we compare our image estimation algorithm with those of other authors, and show quantitative results for all variants of our algorithm.

In subsequent sections, we make finer comparisons between individual variants of our algorithm.

Spectral Error

Spectral evaluation metrics, averaged over the four images in our subset of the KAIST dataset, are provided below in Table 5.1.

Table 5.1: Spectral error metrics averaged across images from the KAIST dataset. Rows are sorted by MRAE. The first column identifies the combination of penalty terms used in our algorithms, as listed in Table 3.1. The second column indicates whether or not we imposed a non-negativity constraint on the estimated image. The third column identifies the criterion used to select regularization weights (Section 3.6).

			$MRAE$	$RMSE_{\text{pixel}}$	$RMSE_{\text{global}}$	GOF	$SSIM_{\text{mean}}$
Penalties	NNeg	Criterion					
L1L1	yes	MSEC	**0.229**	$\mathbf{1.81}\times10^{-4}$	$\mathbf{2.43}\times10^{-4}$	0.850	**0.99950**
L2	yes	MSEC	0.230	1.89×10^{-4}	2.61×10^{-4}	0.833	0.99942
L1	yes	MSEC	0.231	1.85×10^{-4}	2.57×10^{-4}	**0.852**	0.99946
L2L2	yes	MSEC	0.232	1.84×10^{-4}	2.50×10^{-4}	0.834	0.99947
L2L2	no	MSEC	0.244	1.89×10^{-4}	2.58×10^{-4}	0.835	0.99943
L1	no	DMC	0.245	1.87×10^{-4}	2.56×10^{-4}	0.849	0.99943
L2	no	MSEC	0.248	1.94×10^{-4}	2.69×10^{-4}	0.834	0.99939
L1	no	MSEC	0.254	1.91×10^{-4}	2.65×10^{-4}	0.844	0.99941
L1L1	no	MSEC	0.257	1.88×10^{-4}	2.59×10^{-4}	0.839	0.99942
L1	yes	DMC	0.259	2.11×10^{-4}	3.13×10^{-4}	0.843	0.99917
L1L1	yes	DMC	0.260	2.12×10^{-4}	3.12×10^{-4}	0.837	0.99915
L2	yes	DMC	0.298	2.17×10^{-4}	3.12×10^{-4}	0.834	0.99916
L2L2	yes	DMC	0.298	2.17×10^{-4}	3.13×10^{-4}	0.834	0.99916
Choi *et al.* [19]			0.304	2.32×10^{-4}	3.62×10^{-4}	0.829	0.99888
L2L2	yes	MDC	0.322	2.19×10^{-4}	3.07×10^{-4}	0.835	0.99919
L1L1	yes	MDC	0.330	2.19×10^{-4}	3.03×10^{-4}	0.837	0.99916
L2	yes	MDC	0.350	2.45×10^{-4}	3.70×10^{-4}	0.831	0.99885
L2L2	no	MDC	0.352	2.13×10^{-4}	2.92×10^{-4}	0.833	0.99927
L1	yes	MDC	0.362	3.23×10^{-4}	6.26×10^{-4}	0.811	0.99707
L1L1	no	MDC	0.375	2.20×10^{-4}	3.01×10^{-4}	0.832	0.99917
L1L1	no	DMC	0.377	2.17×10^{-4}	3.09×10^{-4}	0.830	0.99915
L2L2	no	DMC	0.394	2.23×10^{-4}	3.08×10^{-4}	0.829	0.99918
L2	no	DMC	0.394	2.23×10^{-4}	3.09×10^{-4}	0.829	0.99918
L2	no	MDC	0.426	2.39×10^{-4}	3.39×10^{-4}	0.829	0.99902
L1	no	MDC	0.632	3.15×10^{-4}	5.25×10^{-4}	0.829	0.99785

Note that none of the algorithms listed in Table 5.1 used a penalty on the

image Laplacian, and so the results do not include all combinations of penalty terms listed in Table 3.1. Unfortunately, penalties on the image Laplacian exhibited poor convergence. Since they were time-consuming to optimize, we only evaluated them on the first image from the dataset (Figure 4.14a). Results on the first image, for all combinations of penalty terms, are provided in Table A.1 (Appendix A).

From the spectral evaluation results, we observe that the method of Choi *et al.* [19] performs well, but, depending on the penalty terms we include, has inferior performance to our algorithms. Differences in the numbers of iterations each method performed, and the lack of an automatic procedure in Choi *et al.* [19] for selecting weights on their prior terms make the comparison inconclusive. Regardless, our experiment suggests that a data-driven prior is not necessary to obtain good results.

Our version of the first stage of the method of Baek *et al.* [7] (discussed in Section 4.6.2), corresponding to the `L1L1` condition without a non-negativity constraint in Tables 5.1 and A.1, does not perform as well as when we augment it with a non-negativity constraint. Overall, regardless of whether we use it with or without a non-negativity constraint, its performance is not significantly different from that of a single penalty term, on the spatial gradient of the image (*i.e.* the `L1` or `L2` conditions).

Since the intention behind their mixed-gradient penalty term was to align edges between spectral bands, it is difficult to assess their penalty on this dataset. In the absence of a model of spectral dispersion, but with input images such as these containing some spectral dispersion, the data-fitting term will discourage full alignment between spectral bands. Therefore, the mixed-gradient penalty will be in conflict with the data-fitting term, and will not be as effective. The spectral dispersion in the KAIST dataset is reasonably small, however, especially in comparison to the experimental setting of Baek *et al.*, where a prism was in front of the lens [7]. Therefore, we conjecture that it is possible to achieve good performance without the spectral-spatial image gradient penalty, regardless of whether dispersion from a lens is accounted for in the data-fitting term.

Colour Error

When evaluating colour image reconstruction, we can now include results from a larger set of algorithms, such as demosaicing algorithms. Using a format analogous to our presentation of spectral reconstruction results, Table 5.2 displays error metrics averaged over the four images from the KAIST dataset, whereas Table A.2 (Appendix A) displays error metrics for the first image (Figure 4.14a), but includes results for penalties on the image Laplacian.

Table 5.2: RGB error metrics averaged across images from the KAIST dataset. Rows are sorted by MRAE in the Green channel ($MRAE_G$). The first column distinguishes spectral image estimation algorithms from colour image estimation algorithms. The second column identifies the combination of penalty terms used in our algorithms, as listed in Table 3.1. The third column indicates whether or not we imposed a non-negativity constraint on the estimated image. The fourth column identifies the criterion used to select regularization weights (Section 3.6).

				MRAE Red	*MRAE* Green	*MRAE* Blue	*RMSE* Red	*RMSE* Green	*RMSE* Blue	*SSIM* Red	*SSIM* Green	*SSIM* Blue
Spectral	Penalties	NNeg	Criterion									
no	ARI [79]			0.0118	**0.00506**	**0.0107**	**0.00105**	**0.00069**	**0.00131**	**0.99963**	**0.99982**	**0.99958**
no	Bilinear interpolation			**0.0113**	0.00509	0.0110	0.00121	0.00099	0.00147	0.99955	0.99978	0.99951
no	L2	no	DMC	0.0143	0.00511	0.0133	0.00149	0.00100	0.00188	0.99932	0.99978	0.99924
no	L2L2	no	DMC	0.0152	0.00511	0.0140	0.00163	0.00099	0.00208	0.99920	0.99978	0.99913
no	L2	no	MSEC	0.0149	0.00516	0.0137	0.00158	0.00100	0.00201	0.99926	0.99978	0.99917
no	L1	no	DMC	0.0151	0.00521	0.0139	0.00164	0.00099	0.00210	0.99921	0.99978	0.99913
no	L2L2	no	MSEC	0.0146	0.00535	0.0124	0.00155	0.00108	0.00181	0.99924	0.99971	0.99926
no	L2	yes	MSEC	0.0424	0.00607	0.0418	0.00175	0.00102	0.00232	0.99877	0.99977	0.99786
no	L1L1	no	MSEC	0.0150	0.00608	0.0131	0.00128	0.00092	0.00163	0.99946	0.99980	0.99942
no	L1	no	MSEC	0.0174	0.00631	0.0149	0.00160	0.00096	0.00206	0.99922	0.99979	0.99914
no	L2L2	yes	MSEC	0.0157	0.00635	0.0202	0.00131	0.00098	0.00174	0.99943	0.99978	0.99922
yes	L1L1	yes	DMC	0.0168	0.00735	0.0142	0.00171	0.00128	0.00249	0.99910	0.99954	0.99882
no	L2	yes	DMC	0.0627	0.00736	0.0622	0.00182	0.00101	0.00248	0.99823	0.99976	0.99626
no	L2L2	yes	DMC	0.0679	0.00753	0.0677	0.00180	0.00099	0.00248	0.99806	0.99977	0.99578
yes	L2L2	yes	DMC	0.0186	0.00759	0.0149	0.00181	0.00123	0.00209	0.99900	0.99967	0.99918
yes	L2	yes	DMC	0.0186	0.00759	0.0149	0.00181	0.00123	0.00209	0.99900	0.99967	0.99918
yes	L2L2	no	DMC	0.0185	0.00760	0.0149	0.00181	0.00123	0.00209	0.99900	0.99967	0.99917
yes	L2	no	DMC	0.0185	0.00760	0.0149	0.00181	0.00123	0.00209	0.99900	0.99967	0.99917
yes	L1	yes	DMC	0.0198	0.00804	0.0152	0.00260	0.00163	0.00334	0.99802	0.99871	0.99770
yes	L2L2	no	MSEC	0.0182	0.00963	0.0206	0.00192	0.00172	0.00275	0.99834	0.99851	0.99755

Table 5.2: (continued)

Spectral	Penalties	NNeg	Criterion	$MRAE$ Red	$MRAE$ Green	$MRAE$ Blue	$RMSE$ Red	$RMSE$ Green	$RMSE$ Blue	$SSIM$ Red	$SSIM$ Green	$SSIM$ Blue
yes	L2L2	yes	MSEC	0.0195	0.01032	0.0196	0.00155	0.00150	0.00230	0.99912	0.99923	0.99861
yes	L1L1	yes	MSEC	0.0232	0.01114	0.0221	0.00208	0.00301	0.00529	0.99864	0.99642	0.99483
yes	L1L1	no	MSEC	0.0230	0.01114	0.0173	0.00193	0.00185	0.00302	0.99891	0.99930	0.99867
no	L1L1	no	DMC	0.0247	0.01191	0.0192	0.00145	0.00089	0.00182	0.99917	0.99974	0.99912
no	L1L1	yes	MSEC	0.0206	0.01258	0.0180	0.00187	0.00113	0.00238	0.99894	0.99959	0.99890
no	L1L1	yes	DMC	0.0273	0.01260	0.0222	0.00342	0.00142	0.00497	0.99758	0.99950	0.99704
yes	L1	yes	MSEC	0.0294	0.01352	0.0286	0.00380	0.00410	0.00737	0.99636	0.99355	0.99058
yes	L2	no	MSEC	0.0317	0.01437	0.0332	0.00347	0.00242	0.00420	0.99418	0.99528	0.99273
yes	L2	yes	MSEC	0.0301	0.01448	0.0299	0.00232	0.00204	0.00345	0.99773	0.99761	0.99583
yes	L1	no	DMC	0.0348	0.01523	0.0220	0.00203	0.00185	0.00387	0.99862	0.99897	0.99757
yes	L1	no	MSEC	0.0293	0.01556	0.0308	0.00256	0.00483	0.00867	0.99786	0.99033	0.98730
yes	L1L1	no	DMC	0.0355	0.01726	0.0211	0.00165	0.00139	0.00263	0.99894	0.99944	0.99881
no	L2	yes	MDC	0.0342	0.01956	0.0295	0.00347	0.00326	0.00467	0.99713	0.99841	0.99683
no	L2	no	MDC	0.0343	0.01957	0.0296	0.00347	0.00326	0.00467	0.99713	0.99841	0.99683
no	L2L2	yes	MDC	0.0337	0.01970	0.0292	0.00341	0.00329	0.00461	0.99722	0.99838	0.99690
no	L2L2	no	MDC	0.0337	0.01970	0.0293	0.00341	0.00328	0.00462	0.99721	0.99838	0.99690
no	L1	yes	MSEC	0.0391	0.02054	0.0298	0.00288	0.00169	0.00373	0.99755	0.99922	0.99740
no	L1	yes	DMC	0.0444	0.02405	0.0333	0.00293	0.00170	0.00377	0.99725	0.99905	0.99710
yes	Choi *et al.* [19]			0.0459	0.03321	0.0358	0.00204	0.00289	0.00336	0.99754	0.99515	0.99579
yes	L2L2	no	MDC	0.1128	0.05851	0.0661	0.00883	0.00927	0.00936	0.98365	0.98904	0.98883
yes	L2L2	yes	MDC	0.1131	0.05855	0.0662	0.00884	0.00928	0.00937	0.98362	0.98901	0.98881
yes	L2	yes	MDC	0.1154	0.05948	0.0673	0.00919	0.00947	0.00970	0.98288	0.98863	0.98832
yes	L2	no	MDC	0.1154	0.05955	0.0674	0.00920	0.00948	0.00971	0.98286	0.98862	0.98830
yes	L1	yes	MDC	0.1224	0.06342	0.0713	0.00945	0.00822	0.00902	0.97799	0.98351	0.98416

Table 5.2: (continued)

Spectral	Penalties	NNeg	Criterion	$MRAE$ Red	$MRAE$ Green	$MRAE$ Blue	$RMSE$ Red	$RMSE$ Green	$RMSE$ Blue	$SSIM$ Red	$SSIM$ Green	$SSIM$ Blue
yes	L1L1	yes	MDC	0.1279	0.06853	0.0751	0.00916	0.00830	0.00891	0.97856	0.98336	0.98442
yes	L1	no	MDC	0.1450	0.07371	0.0798	0.00943	0.00847	0.00923	0.97808	0.98316	0.98376
yes	L1L1	no	MDC	0.1425	0.07697	0.0801	0.00924	0.00840	0.00899	0.97814	0.98301	0.98415
no	L1	no	MDC	0.3929	0.19792	0.2389	0.01292	0.01080	0.01403	0.94610	0.96648	0.95699
no	L1	yes	MDC	0.3932	0.19810	0.2391	0.01293	0.01082	0.01407	0.94606	0.96646	0.95696
no	L1L1	no	MDC	0.3228	0.26205	0.1922	0.01112	0.01517	0.01253	0.94804	0.95333	0.96446
no	L1L1	yes	MDC	0.3333	0.27273	0.2026	0.01166	0.01545	0.01276	0.95654	0.95140	0.96329

Regarding the algorithm variants listed in Tables 5.2 and A.2, recall that the colour versions of our algorithm recover latent images in the space of colour channels, rather than the space of spectral bands. We explained the difference between our colour and spectral image estimation algorithms in Section 3.9.1. In the tables, when we indicate that a colour image estimation algorithm uses a "spectral" prior (*e.g.* `L1SpectralLap2`), we mean that there is a penalty on the cross-channel gradient of the image spatial gradient. Specifically, we calculate the cross-channel penalty by subtracting the Red channel's spatial gradient from the Green channel's spatial gradient, and the Green channel's spatial gradient from the Blue channel's spatial gradient. For spectral image estimation, such a "spectral" penalty is truly spectral, because it is on the spectral gradient of the image spatial gradient. We reused the label of the penalty term, "spectral", for colour image estimation algorithms, because the penalty term has the same mathematical form, although it has a different physical interpretation.

The colour versions of the dataset images are recovered with much higher accuracy than their spectral versions, as is evident when comparing the errors reported in Tables 5.1 and A.1 to the errors reported in Tables 5.2 and A.2. This result is to be expected, because colour image estimation is a better-posed problem than spectral image estimation. We observe that algorithms that operate in the space of colour channels often perform better than spectral image estimation algorithms, possibly indicating that regularization penalties should be applied in the same space (*i.e.* colour channels or spectral bands) in which the image reconstruction error will be evaluated.

Comparing our algorithms to those of other authors, we draw the same conclusions as for the spectral error results. The method of Choi *et al.* [19] yields higher error than most variants of our algorithm in Table 5.2, again suggesting that simple smoothness priors with appropriate weights can be more effective than data-driven priors. With respect to the image spectral-spatial gradient prior of Baek *et al.* [7], we again observe that it offers no advantage. In contrast, demosaicing algorithms are clearly superior for colour image estimation. While Adaptive Residual Interpolation [79] does not perform as well

as some variants of our algorithm in Table A.2, most of these variants used the true image for regularization weight selection. Moreover, the results in Table A.2 are only for one image from the dataset, and may not reflect performance in general. On the other hand, bilinear interpolation performs almost as well as Adaptive Residual Interpolation, suggesting that these images are not challenging inputs for demosaicing algorithms. We note that the images from the KAIST dataset contain mostly diffuse edges, and so the results of different interpolation-based demosaicing algorithms will be similar.

5.2.2 Non-Negativity Constraint

In Table 5.3, we present a subset of the results from Table 5.1 to more clearly compare spectral image estimation with and without a non-negativity constraint on the estimated image. A non-negativity constraint improves the MRAE, often by a significant margin, for all algorithms, except for the `L1` condition with the DMC for regularization weight selection. A similar conclusion can tentatively be drawn from the results in Table A.1, for algorithms which used a penalty on the image Laplacian.

The same findings apply to colour error, as shown in Table 5.4, although for direct colour image estimation, a non-negativity constraint usually worsens the MRAE. For spectral image estimation, a non-negativity constraint likely reduces error by making the problem better posed, whereas for colour image estimation, a non-negativity constraint may worsen the bias introduced by the regularization penalties.

We note that Choi *et al.* [19] decided against using a non-negativity constraint because it would complicate their optimization algorithm and slow convergence. In our experiments, a non-negativity constraint sometimes improved convergence, such as for the `L1L1` spectral image estimation condition. We conclude that a non-negativity constraint is necessary because it improves spectral image estimation accuracy, and that its computational burden is justifiable.

Evaluating images with negative values can be done in several ways. On one hand, as negative intensities are not physically possible, our error met-

Table 5.3: MRAE averaged across images from the KAIST dataset, comparing algorithms with and without a non-negativity constraint. Rows are sorted by MRAE. The first column identifies the combination of penalty terms used in our algorithms, as listed in Table 3.1. The second column identifies the criterion used to select regularization weights (Section 3.6). Bold values indicate the better errors between the algorithm variants with or without a non-negativity constraint.

		$MRAE$ (NNeg)	$MRAE$ (without NNeg)
Penalties	Criterion		
L1L1	MSEC	**0.229**	0.257
L2	MSEC	**0.230**	0.248
L1	MSEC	**0.231**	0.254
L2L2	MSEC	**0.232**	0.244
L1	DMC	0.259	**0.245**
L1L1	DMC	**0.260**	0.377
L2	DMC	**0.298**	0.394
L2L2	DMC	**0.298**	0.394
L2L2	MDC	**0.322**	0.352
L1L1	MDC	**0.330**	0.375
L2	MDC	**0.350**	0.426
L1	MDC	**0.362**	0.632

rics are too tolerant of negative values. We might instead simply reject any algorithm that produces images with negative values. Alternatively, we could be more lenient, by thresholding images to remove negative intensities prior to evaluation. Thresholding would be a necessary post-processing step in a real application, for any algorithm that may produce negative intensity values. Thresholding after image reconstruction is similar to applying a non-negativity constraint, except without a guarantee of global optimality. Therefore, we prefer to incorporate a non-negativity constraint directly into our global optimization algorithm.

5.2.3 Selection of Image Priors

Different regularization penalties are each suited to different image textures, so it is difficult to make universal judgements concerning which penalties are appropriate. Selecting regularization penalties is also a matter of personal preference, as we illustrate by showing image patches later in Section 5.4.2.

Table 5.4: RGB MRAE averaged across images from the KAIST dataset, comparing algorithms with and without a non-negativity constraint. Rows are sorted by MRAE in the Green channel ($MRAE_G$). The first column distinguishes spectral image estimation algorithms from colour image estimation algorithms. The second column identifies the combination of penalty terms used in our algorithms, as listed in Table 3.1. The third column identifies the criterion used to select regularization weights (Section 3.6). Bold values indicate the better errors between the algorithm variants with or without a non-negativity constraint.

			with NNeg constraint			without NNeg constraint		
			$MRAE$ Red	$MRAE$ Green	$MRAE$ Blue	$MRAE$ Red	$MRAE$ Green	$MRAE$ Blue
Sp.	Pen.	Crit.						
no	L2	DMC	0.0627	0.00736	0.0622	**0.0143**	**0.00511**	**0.0133**
no	L2L2	DMC	0.0679	0.00753	0.0677	**0.0152**	**0.00511**	**0.0140**
no	L2	MSEC	0.0424	0.00607	0.0418	**0.0149**	**0.00516**	**0.0137**
no	L1	DMC	0.0444	0.02405	0.0333	**0.0151**	**0.00521**	**0.0139**
no	L2L2	MSEC	0.0157	0.00635	0.0202	**0.0146**	**0.00535**	**0.0124**
no	L1L1	MSEC	0.0206	0.01258	0.0180	**0.0150**	**0.00608**	**0.0131**
no	L1	MSEC	0.0391	0.02054	0.0298	**0.0174**	**0.00631**	**0.0149**
yes	L1L1	DMC	**0.0168**	**0.00735**	**0.0142**	0.0355	0.01726	0.0211
yes	L2L2	DMC	0.0186	**0.00759**	**0.0149**	**0.0185**	0.00760	0.0149
yes	L2	DMC	0.0186	**0.00759**	**0.0149**	**0.0185**	0.00760	0.0149
yes	L1	DMC	**0.0198**	**0.00804**	**0.0152**	0.0348	0.01523	0.0220
yes	L2L2	MSEC	0.0195	0.01032	**0.0196**	**0.0182**	**0.00963**	0.0206
yes	L1L1	MSEC	0.0232	**0.01114**	0.0221	**0.0230**	0.01114	**0.0173**
no	L1L1	DMC	0.0273	0.01260	0.0222	**0.0247**	**0.01191**	**0.0192**
yes	L1	MSEC	0.0294	**0.01352**	**0.0286**	**0.0293**	0.01556	0.0308
yes	L2	MSEC	**0.0301**	0.01448	**0.0299**	0.0317	**0.01437**	0.0332
no	L2	MDC	**0.0342**	**0.01956**	**0.0295**	0.0343	0.01957	0.0296
no	L2L2	MDC	**0.0337**	**0.01970**	**0.0292**	0.0337	0.01970	0.0293
yes	L2L2	MDC	0.1131	0.05855	0.0662	**0.1128**	**0.05851**	**0.0661**
yes	L2	MDC	**0.1154**	**0.05948**	**0.0673**	0.1154	0.05955	0.0674
yes	L1	MDC	**0.1224**	**0.06342**	**0.0713**	0.1450	0.07371	0.0798
yes	L1L1	MDC	**0.1279**	**0.06853**	**0.0751**	0.1425	0.07697	0.0801
no	L1	MDC	0.3932	0.19810	0.2391	**0.3929**	**0.19792**	**0.2389**
no	L1L1	MDC	0.3333	0.27273	0.2026	**0.3228**	**0.26205**	**0.1922**

With respect to spectral image estimation error, L1-norm penalties generally had an advantage for spectral image estimation when their weights were selected using the DMC (Table 5.1). In contrast, L2-norm penalties performed better than L1-norm penalties for the other weight selection criteria. Therefore, L2-norm penalties seem to be reliable over a broader range of weights. The spectral reconstruction error under penalties on the image Laplacian seems to be much higher than for penalties on other image gradients (Table A.1), while the penalty on the spectral-spatial image gradient does not significantly increase accuracy with respect to only a penalty on the spatial gradient (as discussed in Section 5.2.1).

Regarding colour image error, penalties on the image Laplacian perform well (Table A.2), perhaps because they are more flexible than first-order gradient penalties. It is not clear from the results whether L1 or L2-norm penalties produce better colour error.

Aside from image reconstruction accuracy, there are practical reasons for selecting certain regularization penalties over others. Most notably, image estimation is much faster with L2-norm penalties, which allow for least-squares optimization. First-order gradient penalties are also easier to optimize than second-order gradient penalties, because the former penalties induce constraints on smaller neighbourhoods of pixels than the latter penalties.

A second practical consideration is the difficulty selecting a regularization weight for the spectral-spatial gradient penalty of Baek *et al.* [7]. Tuning weights for this prior requires images without spectral dispersion, because the prior penalizes misalignment between spectral bands. In contrast, priors on the image spatial gradient penalize spatial roughness, and the desired level of spatial roughness can be set by examining aberrated images, as done by the DMC (Section 3.6).

We also note that the spectral-spatial gradient penalty is not sufficient for demosaicing, as it does not enforce smoothness between neighbouring pixels (Figure 3.8). As it must be used in combination with a spatial gradient penalty, image estimation becomes computationally expensive. We recommend against using the spectral-spatial gradient penalty, as its disadvantages outweigh its

slight benefit to image estimation accuracy.

5.2.4 Weights on Regularization Penalties

Whereas different regularization penalties often resulted in similar image reconstruction error, there is a clear distinction between criteria used to select weights for the regularization penalties. The results presented in Section 5.2.1 show that the MDC produces higher spectral and colour error than other criteria, by a large margin. The MSEC generally leads to the most accurate images, although the DMC sometimes gives comparable accuracy, and performs well with respect to colour error, even if spectral images are being estimated. That the MSEC, which minimizes the image reconstruction error, does not always outperform the DMC happens for two reasons: First, we only select regularization weights from a single patch in each image, whereas we evaluate image reconstruction error across the entire image. Second, as mentioned in Section 3.6, the grid search we use to optimize regularization weight selection criteria may not find a global optimum.

Consistent with the second explanation, we observed that the MSEC yielded highly variable regularization weights across images, whereas the MDC, for which there are proven convergence results [108], yielded the most stable regularization weights. It may be that the MDC is simply insensitive to the image data, however. The type of regularization penalty also determined how variable the weights were, with the spectral-spatial gradient penalty having the most variable weights (including in our experiments on other image datasets).

Overall, our results reflect the conjecture by Song *et al.* that "the choice of the differential operator is not so crucial since the criterion also includes the data fitting term and the trade-off between the data fitting and regularization terms is controlled by the regularization parameters." They assert that "only the choice of the regularization parameters really matters" [108].

We can verify their statement by examining colour versions of reconstructed spectral images (Figure 5.2). The MDC selects high weights, resulting in severe oversmoothing (Figure 5.2a). In contrast, the MSEC sometimes selects weights which are too low. As such, undersmoothing is apparent in Figure 5.2b. Colour

(a) (b) (c)

Figure 5.2: A comparison of images estimated under different criteria for selecting weights on regularization penalty terms (introduced in Section 3.6): (a) Regularization weight selection using our version of the MDC [108]. (b) Regularization weight selection using the MSEC, which minimizes the mean squared error with respect to the true spectral image. (c) Regularization weight selection using our DMC, which minimizes the mean squared error with respect to the Green channel of the demosaiced input image. All images were estimated with an L1-norm penalty on the image spatial gradient, and a non-negativity constraint. The image patches shown here are sRGB versions of the estimated spectral images, computed using the CIE 1931 colour matching functions, as described in Section 3.3.1. The input image is the scene shown in Figure 4.14d.

desaturation indicates that intensities in the colour-filter array are not being propagated to neighbouring pixels, as can be seen directly by magnifying the figure to observe a residual colour-filter array pattern. On this image, the DMC selects an appropriate level of smoothness, producing a reasonable demosaicing result (Figure 5.2c). Note that, despite the undesirable visual artifacts, the spectral MRAE and global RMSE (4.5) are lower for the MSEC result relative to the DMC result. If this were not the case, then the MSEC would not have been properly optimized during regularization weight selection. The visual artifacts reflect the higher colour MRAE and RMSE, for all colour channels, of the MSEC result relative to the DMC result.

Our comparison between the MSEC and DMC serves only to validate the DMC. We cannot recommend using the MSEC, as it cannot be used in practice, since the true image is unknown.

5.3 Models of Dispersion

While synthetic experiments help validate different image priors, we did not have realistic simulations of spectral dispersion to validate our methods for

modelling dispersion. We relied instead on real data, specifically images of a pattern of black disks printed on a white board, described in Section 4.2.1, to calibrate models of spectral dispersion. Although we do not know the true dispersion, we can still validate our methods for modelling dispersion. We do so by showing that we obtain consistent models of dispersion across image capture conditions, and between variants of our calibration methods (Appendix B).

For the Computar 07I lens, under light filtered by the optical bandpass filters, we obtain the patterns of dispersion shown in Figure 5.3. By comparison, without filtering the light, and calibrating dispersion between colour channels, as done by Rudakova and Monasse [99], we obtain the patterns shown in Figure 5.4. The images we used to calibrate the spectral and colour channel-based models of dispersion are the filtered-light and unfiltered-light versions, respectively, of the images shown in Figures 4.15b and 4.15c. For the filtered-light images, we used the high-dynamic range images produced under each optical bandpass filter, not individual band images from our multispectral images, as the latter images have been subject to further processing that is not necessary for calibrating dispersion.

The model of dispersion in terms of colour channels does not capture the full range of spectral dispersion: Figure 5.4 shows a peak displacement between the Blue and Red channels of approximately 2.5 pixels, whereas Figure 5.3 shows a peak displacement between the 400 and 700 nm bands of approximately 5 pixels.

To better illustrate the limitations of modelling dispersion between colour channels, in Figure 5.5, we show the models of dispersion between colour channels obtained from the high-dynamic range images captured under our 450 nm centre wavelength optical bandpass filter. Under the filtered light, we measure very little dispersion between the different colour channels, showing that models of dispersion between colour channels are specific to the spectral characteristics of the light reflecting from the calibration pattern.

We also note that the models in Figure 5.5 support our assumption that the optical filters in our experimental apparatus are properly filtering the light

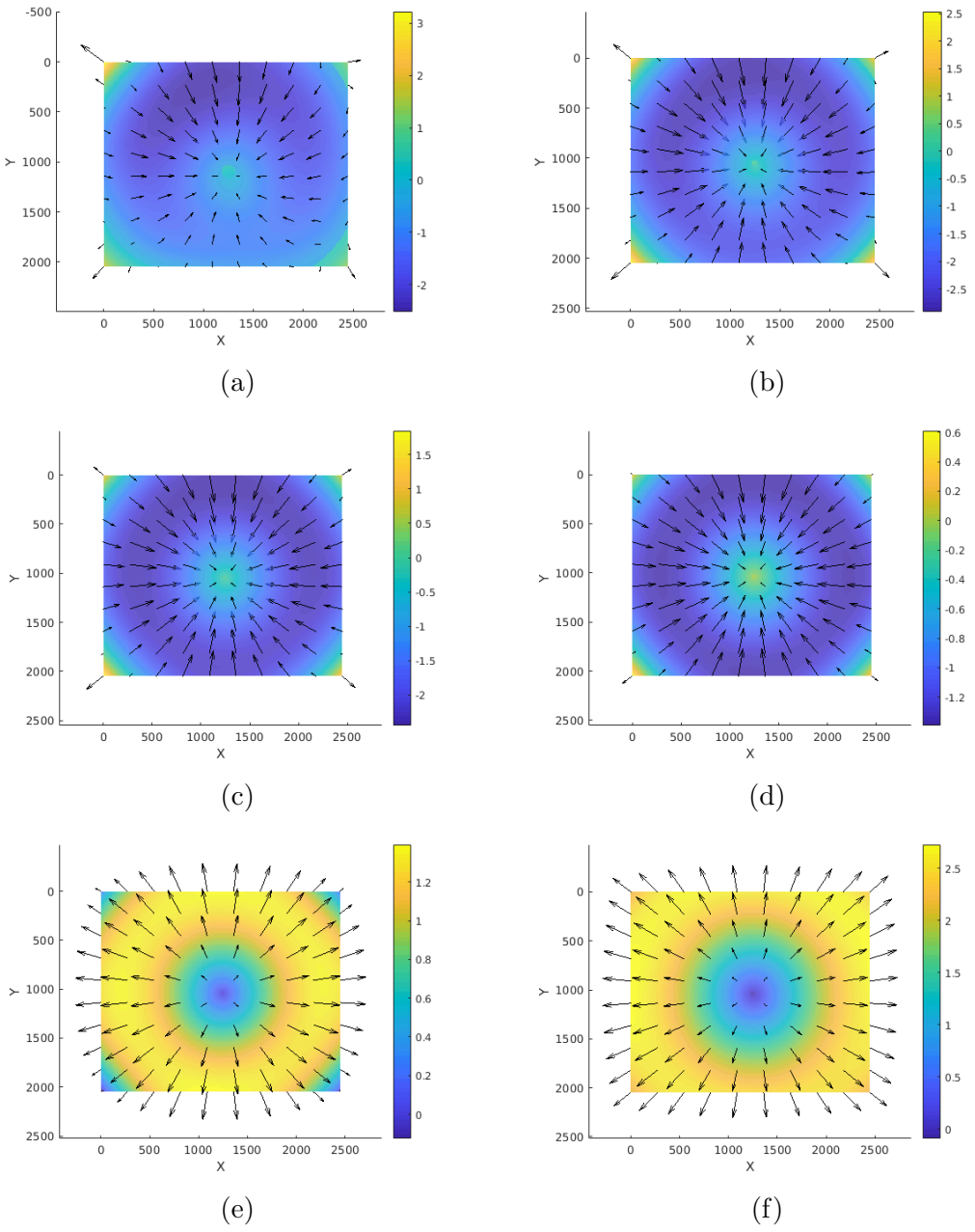
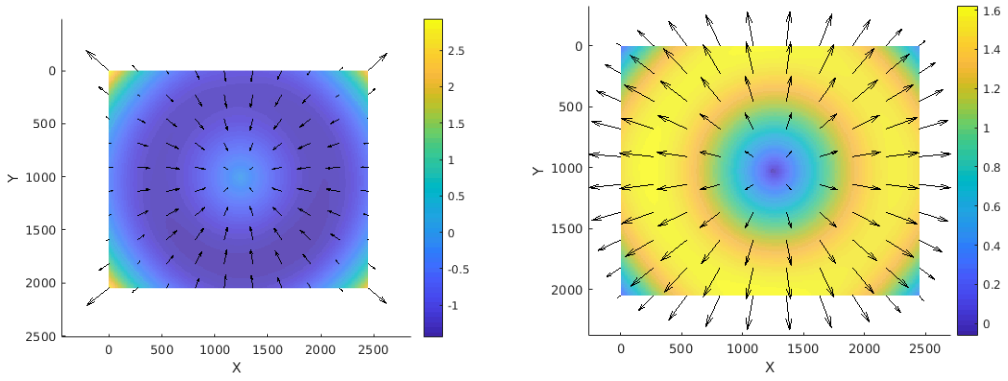


Figure 5.3: Trivariate polynomials in x, y, and λ modeling dispersion for a Computar 07I lens mounted on a Point Grey BlackFly Flea3 camera. From left to right, top to bottom, the plots show aberration relative to 600 nm for spectral bands centered at 400, 450, 500, 550, 650, and 700 nm, respectively. Positive dispersion magnitudes indicate that the dispersion vectors point away from the image center. Arrows in the figures show the dispersion direction.



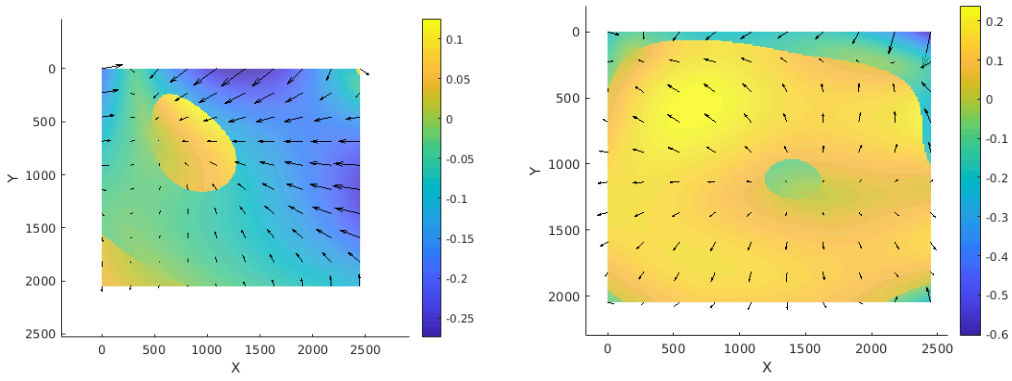
(a) (b)

Figure 5.4: Bivariate polynomials in x and y modeling dispersion for a Computar 07I lens mounted on a Point Grey BlackFly Flea3 camera. The aberration of the Blue channel relative to the Green channel is shown in (a), whereas that for the Red channel relative to the Green channel is shown in (b). Positive dispersion magnitudes indicate that the dispersion vectors point away from the image center. Arrows in the figures show the dispersion direction. Note that dispersion is a spectral phenomenon, so measuring it between colour channels in this manner (as in Rudakova and Monasse [99]) only approximates the results shown in Figure 5.3.

from our DLP projector to narrowband ranges of wavelengths. If the filters transmitted large amounts of light outside of its specified passband, we would have obtained larger dispersion vectors, resulting in patterns more similar to those in Figure 5.4.

In both Figure 5.3 and Figure 5.4, the dispersion patterns are approximately radially symmetrical, as we expected based on literature (*e.g.* Matsuoka *et al.* [75]). The magnitude of dispersion is not monotonic with respect to distance from the image center, however. We investigated several possible explanations for this phenomenon, to determine if it is an artifact of our method for calibrating dispersion.

Our findings, provided in Appendix B, suggest that the non-monotonicity is a physical property of the lenses we used, because it is present in the models of dispersion generated by different variants of our calibration method. Therefore, in our subsequent experiments with spectral and colour image estimation, we presume that any artifacts we observe in our results are unlikely to orig-



(a) (b)

Figure 5.5: Bivariate polynomials in x and y modeling dispersion for a Computar 07I lens mounted on a Point Grey BlackFly Flea3 camera. The models were calibrated from images of a dot pattern captured under light filtered by our 450 nm optical bandpass filter. The aberration of the Blue channel relative to the Green channel is shown in (a), whereas that for the Red channel relative to the Green channel is shown in (b). Positive dispersion magnitudes indicate that the dispersion vectors point away from the image center. Arrows in the figures show the dispersion direction.

inate from our models of dispersion. Rather, artifacts more likely arise from our optimization method and its associated regularization penalties.

5.4 Image Reconstruction from Raw RGB Images

Our initial synthetic experiment on spectral images from the KAIST dataset (Section 5.2) evaluated spectral reconstruction, but not chromatic aberration correction. In this section, we use captured RGB images to evaluate chromatic aberration correction, and to a lesser extent, spectral reconstruction.

5.4.1 Experiment Details

Our input images, shown in Figure 4.15, were captured using our BlackFly Flea3 colour camera with a Computar 07I lens. With this camera and lens combination, we used the model of spectral dispersion shown in Figure 5.3 for spectral image estimation, and the model of colour dispersion shown in Figure 5.4 for direct colour image estimation. Our reconstructed spectral

images have 8 control bands, with centre wavelengths from 386 nm to 707 nm (inclusive), spaced 45.9 nm apart. We selected this spectral sampling scheme as described in Section 5.4.3.

Selection of Algorithms Evaluated

Based on the results of our simulated experiment (Section 5.2.4), we dropped the MDC (3.22) for regularization weight selection, and used only similarity with bilinear interpolation (3.24), the DMC, as our criterion for regularization weight selection. We also excluded the spectral-spatial gradient penalty (3.17), for the reasons discussed in Section 5.2.3.

As we had images for calibrating models of dispersion in our dataset (Figures 4.15b and 4.15c), we were able to compare image reconstruction with, and without, a model of dispersion. For a more conclusive comparison, we selected regularization weights in the absence of models of dispersion, and then used these regularization weights for image estimation both with and without a model of dispersion.

In fact, there are advantages of selecting regularization weights without a dispersion model, even when incorporating the dispersion model into the optimization algorithm during image reconstruction. First, as we select regularization weights only from a single image patch, the regularization weights will not depend on the magnitude of dispersion predicted by a dispersion model. Therefore, the regularization weights are likely to generalize better to patches with other magnitudes of dispersion. Second, selecting regularization weights is faster in the absence of a model of dispersion, because the data-fitting term of the image reconstruction optimization problem is simpler to evaluate (as discussed in Section 3.9.2).

5.4.2 Image Estimation Evaluation

While our dataset of captured raw RGB images is small, it has a sufficient variety of image features to visually differentiate image estimation algorithms. In Figures 5.6, 5.7, 5.8, and 5.9, we compare our spectral image estimation algorithm to the image estimation algorithms from the related works mentioned

in Section 4.6.

Figure 5.6 shows a patch with sharp black and white edges, serving as a typical test case for deblurring and chromatic aberration correction algorithms. We observe that this patch is challenging for demosaicing algorithms ((a) and (f)), which leave some pixellation artifacts visible at edges, in particular in the white '+' symbol near the top left of the patch. The plain demosaicing results also preserve strong colour fringes from chromatic aberration, as shown in the inset. These fringes are muted by the CCT algorithm of Sun *et al.* [111] ((b) and (g)), but are not eliminated completely. Unfortunately, when the Green channel is pre-processed by the blind deconvolution method of Krishnan *et al.* [61], before the CCT algorithm, the image is sharpened, but colour fringes become more vibrant, and ringing appears around image edges ((c) and (h)). In contrast, colour channel warping using the method of Rudakova and Monasse [99] preserves both image blurriness, and demosaicing artifacts, but reduces colour fringing more effectively ((d) and (i)). Our method, with the `Lap2` penalty term, (e), slightly sharpens the image, and leaves subtle colour fringes which are opposite in hue to the original fringes from chromatic aberration. In contrast, our method with the `L1` penalty term, (j), produces an image which is sharper, and has less fringing. The image in (j) appears sharper, and with reduced colour artifacts, than the images (d) and (i) produced by the method of Rudakova and Monasse [99]. Note that neither of our algorithms ((e) and (j)) produce artifacts resembling those of the demosaicing algorithms.

For a less conventional visual test, we present an image patch with bright colours and a specular highlight in Figure 5.7. Whereas demosaicing algorithms lose the edges of the small red dots, and blind deconvolution [61] with CCT [111] alters the colour of the dots, our method restores their edges without altering their colour. Our method also sharpens the specular highlight with minimal ringing, especially with the `L1` penalty term in (j), in contrast to the method of Krishnan *et al.* [61] ((c) and (h)), which introduces strong ringing into the highlight.

Aside from the ringing artifacts it may introduce, deblurring may also undesirably sharpen an image. Consider Figure 5.8, which shows an image patch

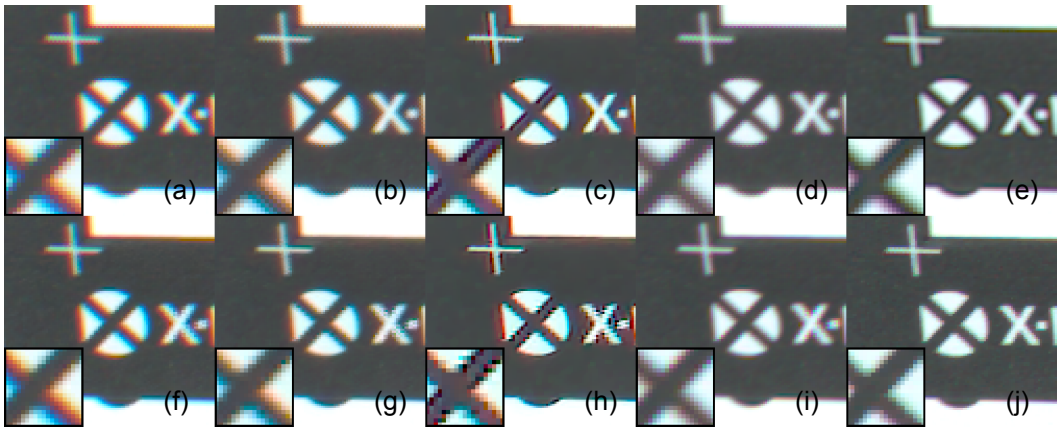


Figure 5.6: A comparison of our image estimation algorithm with related works, on a patch from the image shown in Figure 4.15a. (a) Demosaicing by bilinear interpolation; (b) Demosaicing by bilinear interpolation followed by CCT [111] without blind deconvolution of the Green channel; (c) Demosaicing by bilinear interpolation followed by blind deconvolution of the Green channel [61] and then CCT [111]; (d) Demosaicing by bilinear interpolation followed by colour channel warping [99]; (e) Our spectral image estimation algorithm with the `Lap2` combination of penalties (Table 3.1) and a non-negativity constraint; (f)–(i) Same as (a)–(d), but with ARI [79] as the demosaicing method; (j) Our spectral image estimation algorithm with the `L1` combination of penalties (Table 3.1) and a non-negativity constraint.

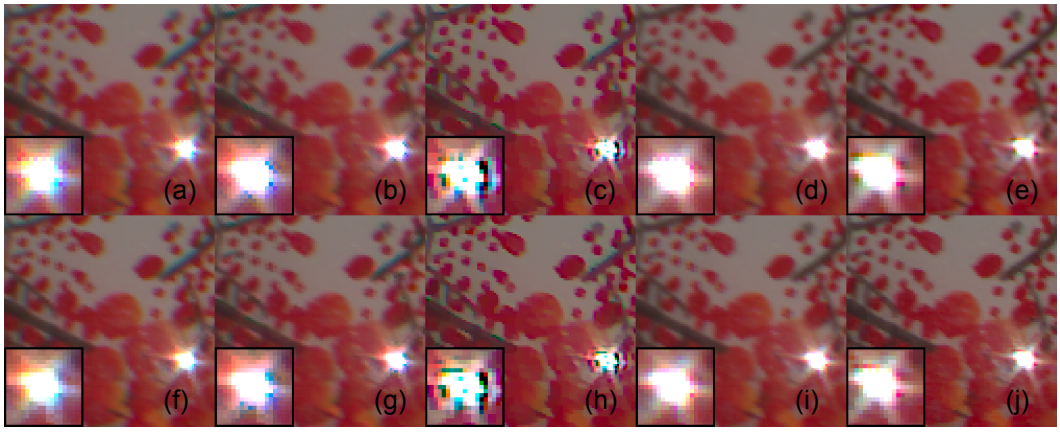


Figure 5.7: A comparison of our image estimation algorithm with related works, on a patch from the image shown in Figure 4.15d. The different conditions shown in the figure are the same as those in Figure 5.6.

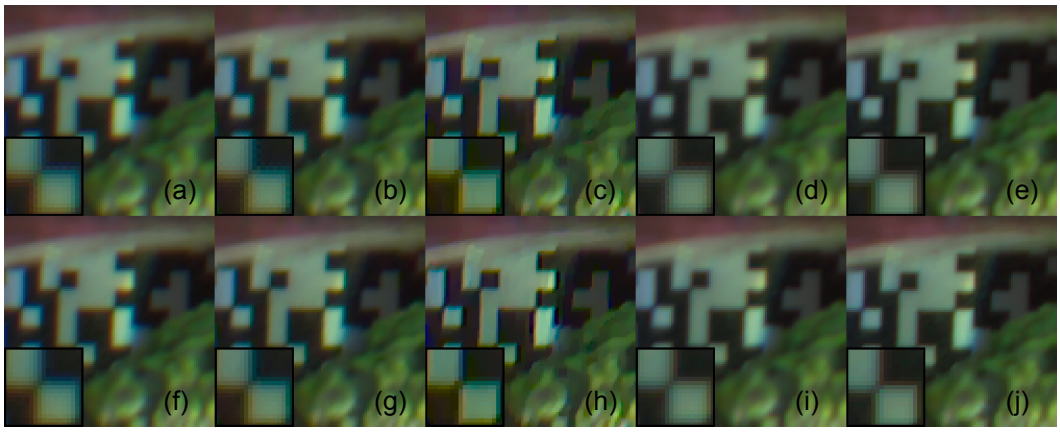


Figure 5.8: A comparison of our image estimation algorithm with related works, on a patch from the image shown in Figure 4.15g. The different conditions shown in the figure are the same as those in Figure 5.6.

containing both sharper (black and white) and blurrier (green) regions. In photography, defocus blur is often introduced for visual effect. Our method supports such an intention, by only eliminating blur caused by lateral chromatic aberration. In Figure 5.8 (j), for example, the entire image patch is noticeably sharper compared to (a) and (f). One can observe the sharpening effect in the inset, and also along the edge between the green region and the black and white region in (j). Yet our method preserves the defocus of the green region. In contrast, blind deconvolution [61] with CCT [111] forcibly sharpens the green region, attempting to remove all blur, not only blur caused by chromatic aberration.

In Figure 5.9, we differentiate image-based from calibrated chromatic aberration correction algorithms using an image patch containing rainbows from a dispersive object. There is little colour fringing from chromatic aberration in this image, as can be seen from the edges of the black dot in the demosaicing results ((a) and (f)). Our method, and that of Rudakova and Monasse [99], produce images similar to those from demosaicing, but alter colour fringing and sharpness at the edges of the dot. In contrast, the method of Sun *et al.* [111] introduces colour artifacts in both the edges of the dot, and in the rainbows ((b) and (g), and especially (c) and (h)). As their method is image-based, it treats the rainbows as chromatic aberration.

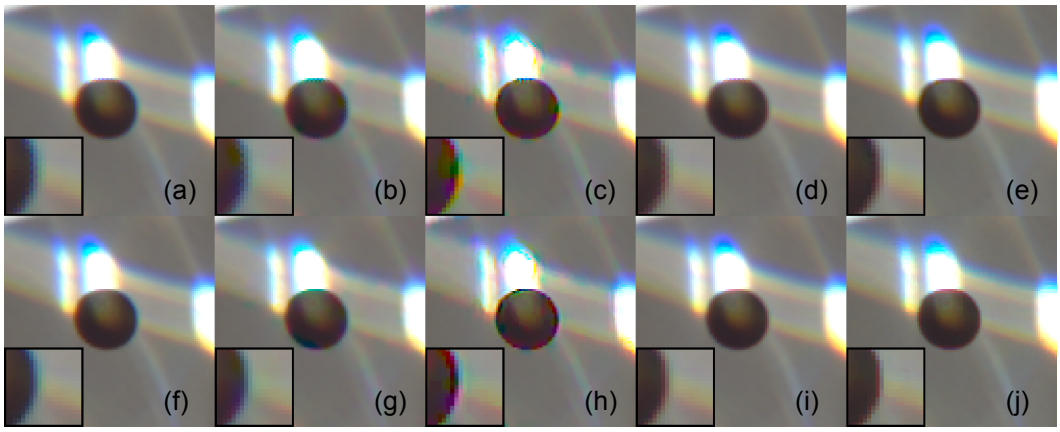


Figure 5.9: A comparison of our image estimation algorithm with related works, on a patch from the image shown in Figure 4.15f. The different conditions shown in the figure are the same as those in Figure 5.6.

In Figures 5.10 and 5.11, we compare different versions of our algorithm. Regardless of the regularization penalties chosen, direct colour image estimation (Section 3.9.1) produces blurrier images than spectral image estimation, and also shows some pixellation artifacts reminiscent of demosaicing artifacts ((a)–(e)). We attribute the sharpening effect of spectral image estimation to correction of within-channel chromatic aberration. Therefore, spectral image estimation has advantages over colour image estimation that warrant the more complicated calibration procedure. We note that such a deblurring effect would be difficult to achieve with conventional deblurring techniques, as they would not distinguish between defocus blur and chromatic aberration.

Comparing the different regularization penalties, in Figures 5.10 and 5.11, we observe that a penalty on the image Laplacian tends to produce some noise in the result, especially when the penalty is subject to the L1-norm ((g) compared with (f), (h), (i), and (j)). A penalty on the first-order spatial gradient, in contrast, smoothes edges when subject to the L2-norm (i), but preserves sharp edges, without introducing noise, when subject to the L1-norm (j). Our preferred combination of image priors is therefore the `L1` condition in Table 3.1, an L1-norm penalty on the image spatial gradient. The `Lap2` condition, an L2-norm penalty on the image spatial Laplacian, also produces reasonable results, although they are slightly blurrier.

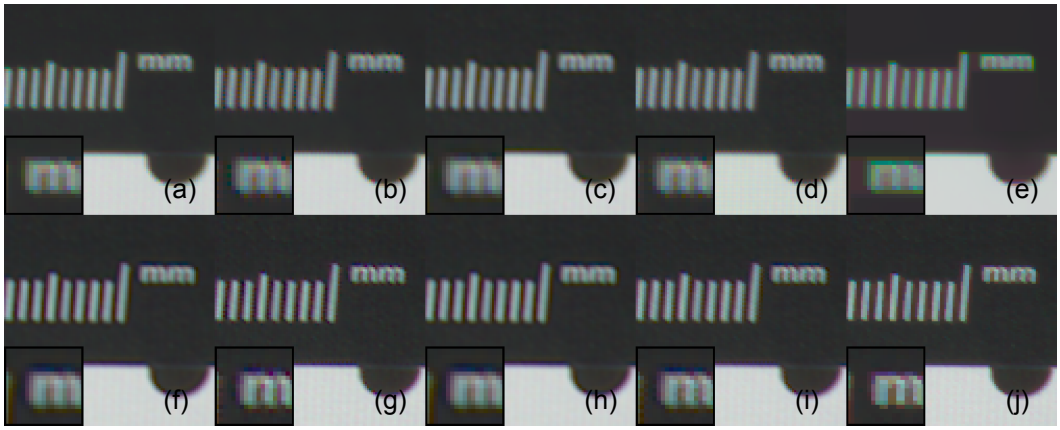


Figure 5.10: A comparison of variants of our image estimation algorithm, on a patch from the image shown in Figure 4.15a. The top row shows the results of direct colour image estimation (Section 3.9.1), whereas the bottom row shows the results of spectral image estimation. The columns compare different regularization penalties (from the list in Table 3.1), all of which were used with a non-negativity constraint on the estimated image: `L1SpatialLap2` ((a) and (f)), `Lap1` ((b) and (g)), `Lap2` ((c) and (h)), `L2` ((d) and (i)), and `L1` ((e) and (j))

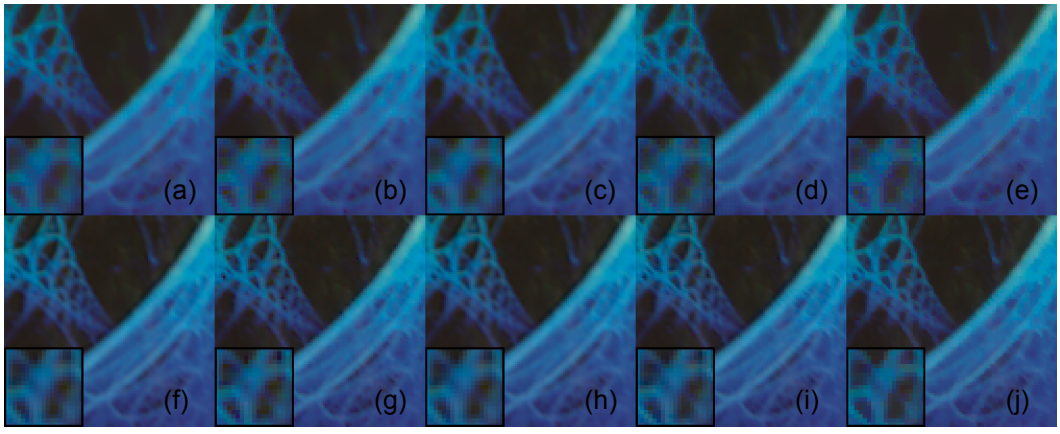


Figure 5.11: A comparison of variants of our image estimation algorithm, on a patch from the image shown in Figure 4.15d. The different conditions shown in the figure are the same as those in Figure 5.10.

Table 5.5: Spectral MRAE and RMSE (4.4) in the 12 pixel-wide regions within ColorChecker patch edges, averaged across patches. Entries are sorted by MRAE. All of our methods used a non-negativity constraint.

Dispersion-aware	Penalties	$MRAE$	$RMSE_{\text{pixel}}$
yes	L1	**0.31156**	0.10466
yes	Lap1	0.31417	0.10428
yes	Lap2	0.31426	0.10467
yes	L1SpatialLap2	0.31430	0.10466
no	Lap2	0.31827	0.10602
no	L1SpatialLap2	0.31831	0.10600
yes	L2	0.32566	**0.10261**
no	L2	0.32693	0.10464
no	L1	0.33265	0.10651
no	Lap1	0.33414	0.10539

The image patches shown in this section illustrate the strengths of different algorithms, but lead to subjective conclusions about which algorithm is most effective. To objectively evaluate chromatic aberration correction, we used our technique described in Section 4.4.3, and obtained the spectral and colour error results shown in Tables 5.5 and 5.6, respectively. Note that chromatic aberration correction by blind deconvolution [61] followed by CCT [111] is not listed in Table 5.6, because we only performed blind deconvolution on image patches, as discussed in Section 4.6.4.

Table 5.5 presents residual chromatic aberration in the spectral domain. From the results, we observe that the "dispersion-aware" methods, which included a model of spectral dispersion in the data-fitting term of the optimization problem (3.1), outperform their variants which excluded a model of dispersion from the data-fitting term (described in Section 3.9.2). Therefore, attempting to correct chromatic aberration does reduce the residual chromatic aberration as measured by our evaluation technique.

To compare our method with previous works on chromatic aberration correction, we must evaluate images in the RGB domain, as presented in Table 5.6. The comparison is more interesting than in the spectral domain, because we can also compare our spectral image estimation algorithm to its colour equivalent (described in Section 3.9.1). Spectral image estimation algorithms have

Table 5.6: MRAE and RMSE in each colour channel in the 12 pixel-wide regions within ColorChecker patch edges, averaged across patches. Entries are sorted by error in the Green channel, then in the Red channel. Note that Rudakova and Monasse [99] and Sun *et al.* [111] (without Krishnan *et al.* [61] as a preprocessing step) do not alter the Green channel, and so have the same error in Green as the demosaicing methods they follow. All of our methods used a non-negativity constraint. The non-spectral variants of our method (Section 3.9.1) reconstruct colour images directly, instead of estimating spectral images.

			$MRAE \times 10^2$			$RMSE \times 10^2$		
Dispersion-aware	Spectral	Penalties	Red	Green	Blue	Red	Green	Blue
yes	yes	L1SpatialLap2	8.82	**8.13**	8.77	1.069	1.912	0.846
yes	yes	Lap2	8.82	8.13	8.77	1.069	**1.912**	0.846
yes	yes	L2	8.93	8.26	8.92	1.074	1.920	0.848
yes	yes	L1	8.85	8.36	8.60	1.083	1.936	**0.840**
yes	yes	Lap1	8.88	8.41	8.76	1.066	1.933	0.851
no	no	L1	9.26	8.54	10.07	1.245	2.153	1.241
yes	no	L1	9.21	8.57	8.65	1.226	2.161	1.005
ARI [79]			**8.49**	8.65	9.38	1.073	2.137	1.116
ARI [79] & warping [99]			9.12	8.65	**8.49**	1.175	2.137	0.925
ARI [79] & CCT [111]			9.18	8.65	9.47	1.146	2.137	1.056
no	no	L1SpatialLap2	9.05	8.66	9.94	1.175	2.145	1.191
no	no	Lap2	8.66	8.66	9.61	1.094	2.134	1.130
yes	no	Lap2	9.05	8.66	8.58	1.163	2.134	0.930
no	no	Lap1	9.04	8.68	9.79	1.131	2.145	1.141
yes	no	Lap1	9.18	8.68	8.63	1.178	2.145	0.938
yes	no	L1SpatialLap2	9.18	8.70	8.70	1.201	2.156	0.977
Raw input image			8.51	8.77	9.49	1.081	2.148	1.140
Bilinear			8.73	8.77	9.62	1.098	2.147	1.130
Bilinear & CCT [111]			9.26	8.77	9.59	1.163	2.147	1.060
Bilinear & warping [99]			9.29	8.77	8.70	1.182	2.147	0.938
no	no	L2	9.96	8.78	10.71	1.193	2.149	1.181
yes	no	L2	9.40	8.82	8.92	1.188	2.158	0.956
no	yes	L1SpatialLap2	8.58	9.46	10.94	1.023	2.359	1.316
no	yes	Lap2	8.58	9.46	10.94	**1.023**	2.359	1.317
no	yes	L2	8.84	9.55	11.31	1.050	2.366	1.299
no	yes	Lap1	8.83	9.56	10.97	1.060	2.376	1.307
no	yes	L1	8.90	9.62	11.39	1.122	2.399	1.309

a clear advantage over colour image estimation algorithms, although the comparison may also be affected by the need to mark different ideal edge locations for spectral vs. colour image evaluation, as mentioned in Section 4.4.3. As in Table 5.5, accounting for dispersion during image estimation yields better results, but primarily only in the Green channel, and primarily only for spectral image estimation algorithms. Comparing our method to algorithms created by other authors, we find that Adaptive Residual Interpolation for demosaicing [79] performs well, even though this algorithm is not designed for chromatic aberration correction. Surprisingly, post-processing demosaicing results with cross-channel information transfer (CCT) [111] or colour channel warping [99] worsens results in the Red channel, but does improve results in the Blue channel.

We note that our results in tables 5.5 and 5.6 reflect performance on one image, and, while we sampled 24 patches from the ColorChecker in that image, our conclusions are only tentative because of the limited amount of test data. In particular, we can draw distinctions between broad classes of algorithms, but feel that there is insufficient data to recommend a particular combination of regularization penalties over others. For now, we recommend selecting regularization penalties based on personal preferences concerning the appearance of the estimated images.

Residual Colour Fringes

A subtle defect in our results, such as shown in Figures 5.6e and 5.6j, is the presence of residual colour fringes. Typically, the fringes are opposite to those in the input images, in that the blue and red portions of a residual fringe are swapped relative to the fringe in the input. The residual fringes are less intense than the colour fringes in the input image, and we note that colour correction has accentuated their saturation (as demonstrated earlier in Figure 4.16). Still, their presence suggests that our algorithm has not properly corrected chromatic aberration.

One may suppose that the residual colour fringes arise from our simplified model of chromatic aberration. Specifically, we have neglected changes

in focusing distance with wavelength, and have modelled dispersion only as a wavelength-dependent spatial warp of the image. If that were the case, we would expect our results to look more like the results of colour channel warping [99] (*e.g.* Figures 5.6d and 5.6i), in which there are somewhat purple fringes that extend on both sides of thin structures. In contrast, our results have red and blue fringes on opposite sides of thin structures. We also refute this explanation based on further experiments, in Section 5.5.

Another possible explanation for the residual fringes is that our regularization penalties have different effects on different spectral bands. Consider the diagrams shown in Figure 5.12, and the explanation in the caption.

The diagrams in Figure 5.12 convey the idea that regularization may produce spectra which are metamers of the true image. Moreover, the diagrams predict that this problem may occur regardless of whether or not dispersion is taken into account during image estimation, because the model of dispersion simply shifts the spectral bands, and does not change their spatial gradients. Therefore, regularization will have the same effect before as after dispersion is applied to the spectral bands.

We were able to verify this prediction in practice, as shown in Figure 5.13. In Figure 5.13, we find that images estimated without a model of dispersion, then corrected for dispersion by warping, are almost identical to images estimated with a model of dispersion as part of the optimization algorithm. The only difference is that the images estimated without a model of dispersion are slightly blurrier. We suggest their relative blurriness comes from image warping with bilinear interpolation for resampling, whereas, when image warping is part of the data-fitting term of the optimization problem (3.1), the bilinear interpolation is inverted (as in deconvolution), sharpening the estimated image.

Residual colour fringes are symmetrical about thin structures in the results from direct colour image estimation ((a), (b), (e), and (f)), and are asymmetrical about thin structures in the results from spectral image estimation ((c), (d), (g), and (h)), regardless of whether or not dispersion was included in the image estimation optimization problem. Therefore, the residual fringes must not be

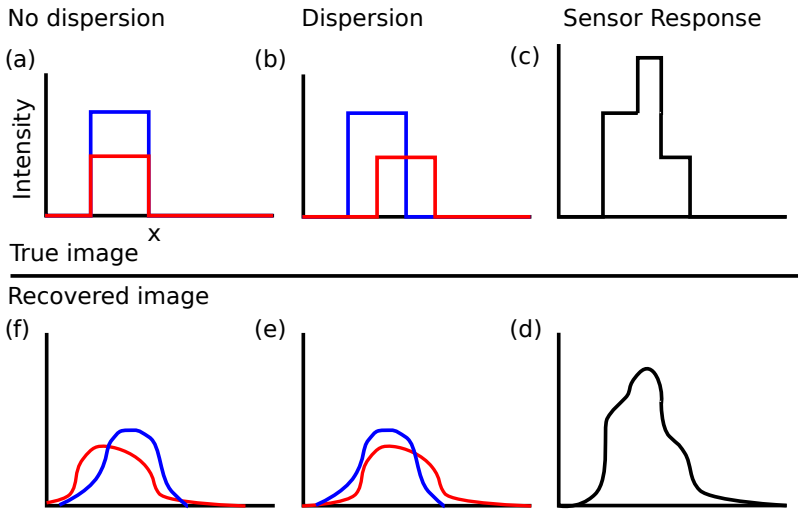


Figure 5.12: A thought experiment to explain the appearance of residual colour fringes in estimated spectral images. The diagrams show image intensities in two spectral bands (red and blue) ((a), (b), (e) and (f)), or one colour channel (black) ((c) and (d)). Intensities are plotted along a line in an image. The plots on the left show intensities in the ideal image (a), or in the chromatic abberration-corrected image (f). The plots in the middle and right columns show intensities in the aberrated ((b) and (c)) and uncorrected ((d) and (e)) versions of the image. In (a), the ideal image is shown, with sharp edges perfectly aligned between spectral bands. (b) shows the light reaching the sensor, which has been altered by spectral dispersion in the lens. (c) is the pattern of intensity recorded by the sensor, which responds to light in both spectral bands. (c) is the input image for the image reconstruction algorithm. The image reconstruction algorithm produces an image, (d), which corresponds to a smoothed version of the original intensity pattern. Smoothing is applied in the space of spectral bands, however, and has produced the spectral band image shown in (e). While (e) projects to (d), and (d) matches the input image (c) reasonably well, (e) deviates substantially from the actual spectral intensities in (b). The reason for the difference between (e) and (b) is that, in (b), the spectral band with higher intensity (blue) also has larger spatial gradients. In an image with strong edges, there will be very low intensity regions with similar intensities across spectral bands, and high-intensity regions with intensities that generally differ between spectral bands. The spectral bands with higher intensities will have larger gradients at the strong edges. The image estimation algorithm, in minimizing gradients, will therefore find a metamer which has more similar intensities in different spectral bands in the high-intensity regions of the image. The redistribution of intensities between spectral bands appears as a spatial spreading of image edges when chromatic aberration is corrected (f) (*i.e.* a residual colour fringe).

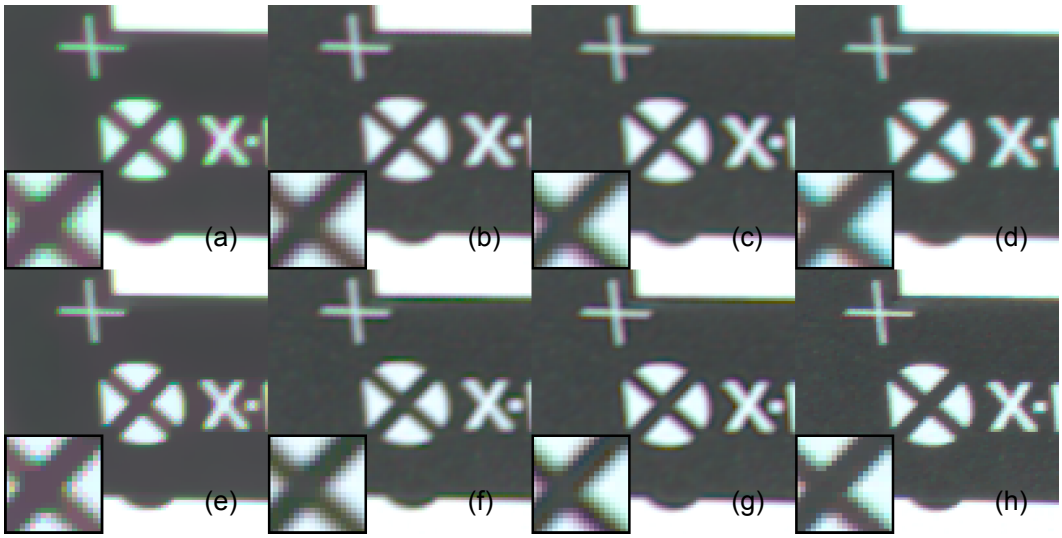


Figure 5.13: A comparison of correcting for dispersion at different times, on a patch from the image shown in Figure 4.15a. The first two columns show the results of direct colour image estimation (Section 3.9.1), whereas the last two columns show the results of spectral image estimation. The columns also compare two different regularization penalties (from the list in Table 3.1), all of which were used with a non-negativity constraint on the estimated image: `L1` ((a), (d), (e), and (h)), and `Lap2` ((b), (c), (f), and (g)). The images in the top row were obtained by estimating the image without a model of dispersion (Section 3.9.2), and then correcting for dispersion afterwards by warping the image. The images in the bottom row were estimated with models of dispersion included in the data-fitting term of the optimization problem (3.1), as usual.

caused by an interaction between the dispersion model and the optimization process. Instead, they likely arise from either our regularization penalties, or from limitations in our models of dispersion, but we needed to perform further experiments to determine which explanation is correct (Section 5.5).

Spectral Reconstruction Evaluation

Aside from assessing spectral reconstruction near edges, as a measure of residual chromatic aberration (Table 5.5), we can assess spectral reconstruction in the centres of the ColorChecker patches. Doing so allows us to compare our method with the GoSpectro spectrometer, which we used to measure the spectral reflectances of the ColorChecker patches as described in Section 4.6.1. We use the Analytical Spectral Devices data (Section 4.6.1) as ground truth spec-

Table 5.7: Spectral MRAE and RMSE (4.4) in the centres of ColorChecker patches, averaged across patches. Entries are sorted by MRAE. All of our methods used a non-negativity constraint. The "filtered" result is the multispectral image created using images captured under narrowband optical filters (Section 4.2.4). We show two replicates for the GoSpectro spectrometer, which was re-calibrated in-between replicates.

Penalties	$MRAE$	$RMSE$
Filtered	**0.19437**	**0.06682**
Lap2	0.25760	0.09120
L1SpatialLap2	0.25762	0.09121
L2	0.25790	0.09091
Lap1	0.25920	0.08755
L1	0.26056	0.08656
GoSpectro-1	0.46633	0.10451
GoSpectro-2	0.53425	0.12325

tral reflectances, and registered the measurements obtained by other methods to this data as described in Section 4.4.1. Note that we did not correct spectral images for intensity non-uniformity (*e.g.* vignetting), because in an arbitrary scene, there would not necessarily be a suitable object for calibrating intensity non-uniformity. For the GoSpectro spectrometer, we captured two replicates of the spectral measurements, re-calibrating the device in-between replicates.

The results are shown in Table 5.7. Interestingly, the GoSpectro spectrometer has lower accuracy than our spectral image estimation algorithm. The multispectral image created using narrowband optical filters has the best accuracy, even though it has spectrally-varying intensity non-uniformity across the image plane. We may consider our multispectral image as the highest quality spectral data that we can collect using our RGB camera. Therefore, there is room for improvement, but our image estimation algorithm is approaching the best spectral reconstruction accuracy we can expect in practice.

Sample plots of spectral reflectances, for patches 13, 14, and 15 of the ColorChecker, representing the primary colours Blue, Green, and Red, are shown in Figures 5.14, 5.15, and 5.16, respectively. In Table 5.7 and in the figures, it is evident that the regularization penalties we use only have a small influence on the spectral reconstruction accuracy. Of course, our regularization

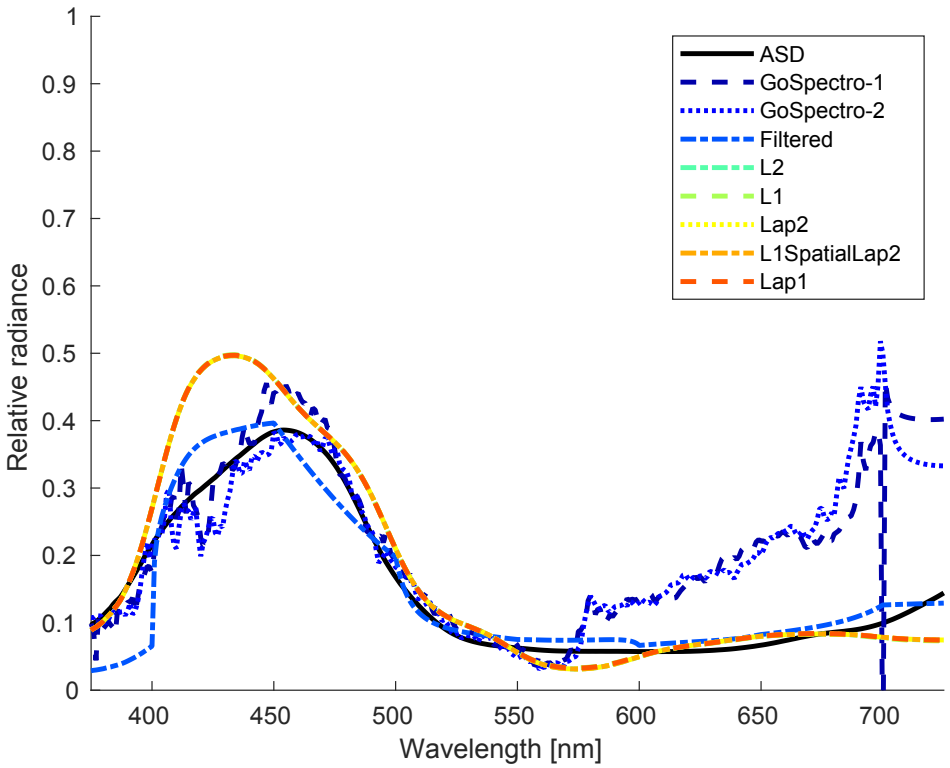


Figure 5.14: A comparison of spectral reflectances measured using spectrometers, or extracted from spectral images, for Patch 13 (Blue) of our X-Rite 24-patch ColorChecker CLASSIC colour calibration chart. "ASD" refers to the Analytical Spectral Devices spectrometer ground truth data (Section 4.6.1), whereas "Filtered" refers to our multispectral image synthesized from images captured under narrowband optical bandpass filters. All of our spectral image estimation algorithm variants have overlapping curves on the plot.

penalties operate on the image spatial gradient, and so would have similar effects in homogenous image regions, such as the ColorChecker patch centres being evaluated.

In summary, our spectral image estimation method performs remarkably well, as it operates on images from an unmodified RGB camera, yet it outperforms the GoSpectro spectrometer, which modifies an RGB camera to increase spectral resolution.

5.4.3 Spectral Image Representation

One detail we have glossed over until now is why we selected 8 spectral control bands for our estimated spectral images. We tested varying numbers of

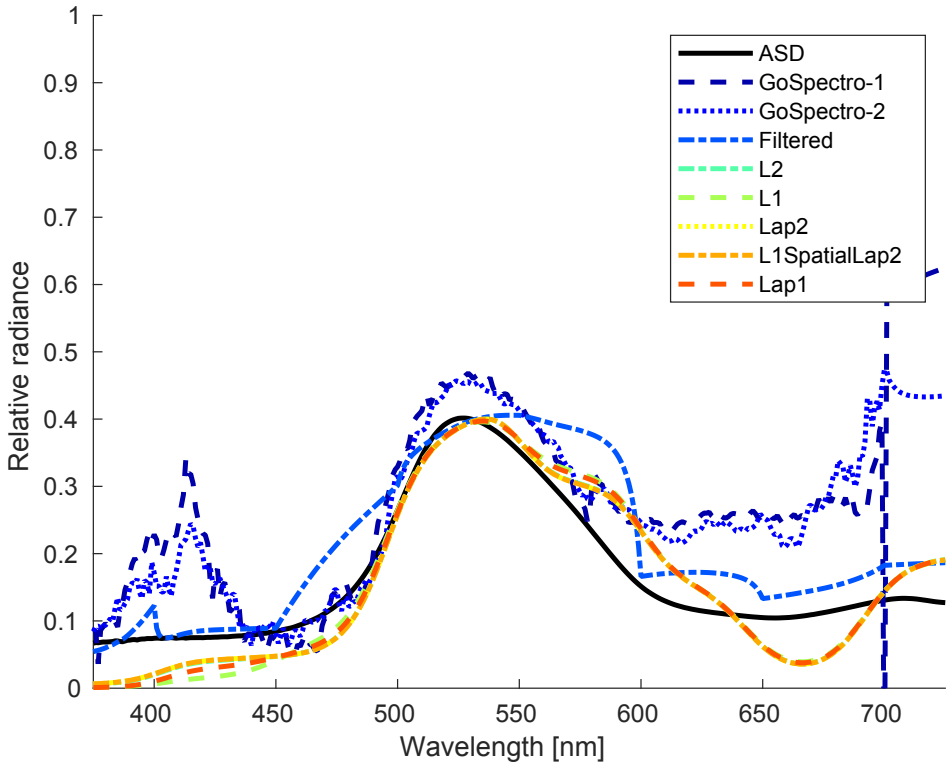


Figure 5.15: A comparison of spectral reflectances measured using spectrometers, or extracted from spectral images, for Patch 14 (Green) of our X-Rite 24-patch ColorChecker CLASSIC colour calibration chart. The curves on the plot have the same interpretation as those in Figure 5.14.

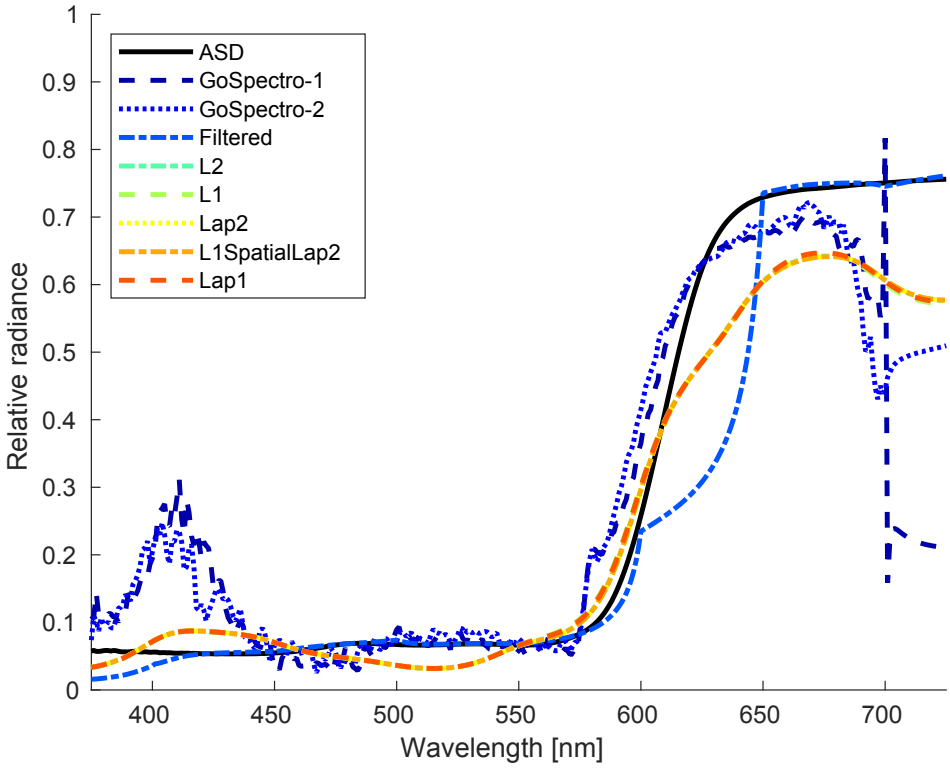


Figure 5.16: A comparison of spectral reflectances measured using spectrometers, or extracted from spectral images, for Patch 15 (Red) of our X-Rite 24-patch ColorChecker CLASSIC colour calibration chart. The curves on the plot have the same interpretation as those in Figure 5.14.

Table 5.8: Spectral MRAE and RMSE (4.4) in the centres of ColorChecker patches, averaged across patches, for spectral images having different sets of control bands. Entries are sorted by MRAE. The "filtered" result is the multispectral image created using images captured under narrowband optical filters (Section 4.2.4). For our spectral image estimation algorithm, all sets of control bands had equally-spaced centre wavelengths, with the first band centered at 386 nm, and the last band centered at 707 nm. We show two replicates for the GoSpectro spectrometer, which was re-calibrated in-between replicates.

Condition	$MRAE$	$RMSE$
Filtered	**0.19437**	**0.06682**
12 bands	0.33525	0.12911
10 bands	0.34141	0.13233
8 bands	0.34643	0.13674
6 bands	0.36850	0.14479
4 bands	0.42854	0.17843
3 bands	0.45908	0.18113
GoSpectro-1	0.46633	0.10451
GoSpectro-2	0.53425	0.12325

spectral bands, in combination with the `L1` set of regularization penalties (Table 3.1), and a non-negativity constraint. The results are listed in Table 5.8, which is analogous to Table 5.7. All sets of spectral control bands that we tested had equally-spaced centre wavelengths, with the first band centered at 386 nm, and the last band centered at 707 nm. The first and last band centre wavelengths were chosen by finding the extreme wavelengths at which any colour channel of our Point Grey FL3-GE-50S5C camera had at least 5% of its peak sensitivity. (The spectral responses of the sensor are plotted in Figure 3.6 for reference.)

In this experiment, we did not observe any degradation in spectral reconstruction accuracy at higher numbers of bands, which is surprising given that the data-fitting operator of our optimization problem (3.1), $\mathbf{M}(\mathbf{\Omega}(\mathbf{\Phi}(\mathbf{B}(\mathbf{I}))))$, is a rank-deficient matrix when there are just two or more spectral control bands. It is also surprising because our regularization penalty in this experiment, a penalty on the image spatial gradient, does not enforce any smoothness in the spectral domain. As shown in Figures 5.17, 5.18, and 5.19, estimated spectra are still smooth at the highest number of control bands we tested.

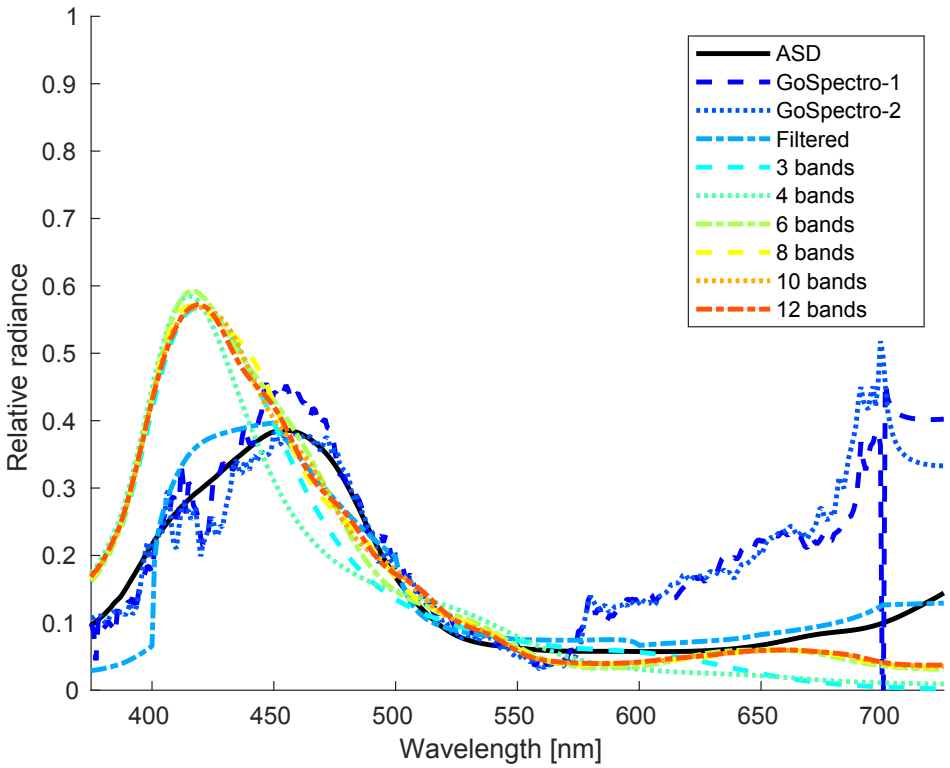


Figure 5.17: A comparison of spectral reflectances measured using spectrometers, or extracted from spectral images, for Patch 13 (Blue) of our X-Rite 24-patch ColorChecker CLASSIC colour calibration chart. "ASD" refers to the Analytical Spectral Devices spectrometer ground truth data (Section 4.6.1), whereas "Filtered" refers to our multispectral image synthesized from images captured under narrowband optical bandpass filters. Our spectral image estimation algorithms differ based on their numbers of spectral control bands in the range 386 to 707 nm.

These figures plot the spectral reflectances for patches 13, 14, and 15 of the ColorChecker, representing the primary colours Blue, Green, and Red, respectively, and are analogous to Figures 5.14, 5.15, and 5.16.

We chose 8 bands in our other experiments as a compromise between accuracy and resource consumption (computation time, and computer memory and storage). We hypothesize that spectral reconstruction accuracy will eventually plateau at high numbers of control bands, and then may possibly degrade.

At low numbers of control bands, we obtain poor results, both in terms of spectral reconstruction accuracy (Table 5.8), and visually. As shown in Figures 5.20 and 5.21, low numbers of bands produce some pixellation artifacts

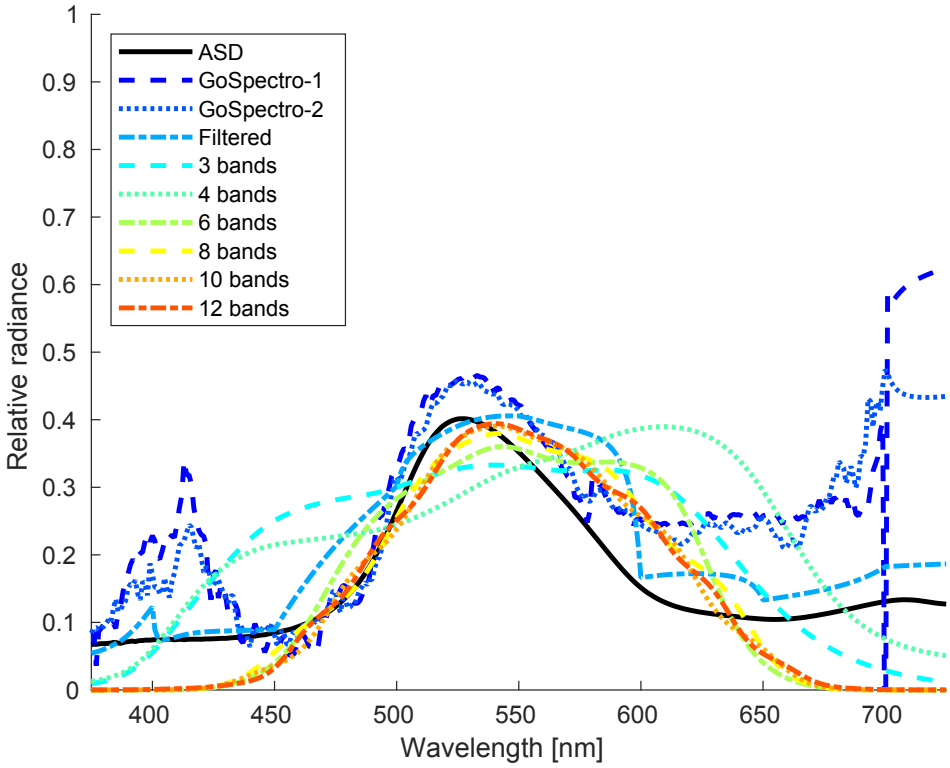


Figure 5.18: A comparison of spectral reflectances measured using spectrometers, or extracted from spectral images, for Patch 14 (Green) of our X-Rite 24-patch ColorChecker CLASSIC colour calibration chart. The curves on the plot have the same interpretation as those in Figure 5.17.

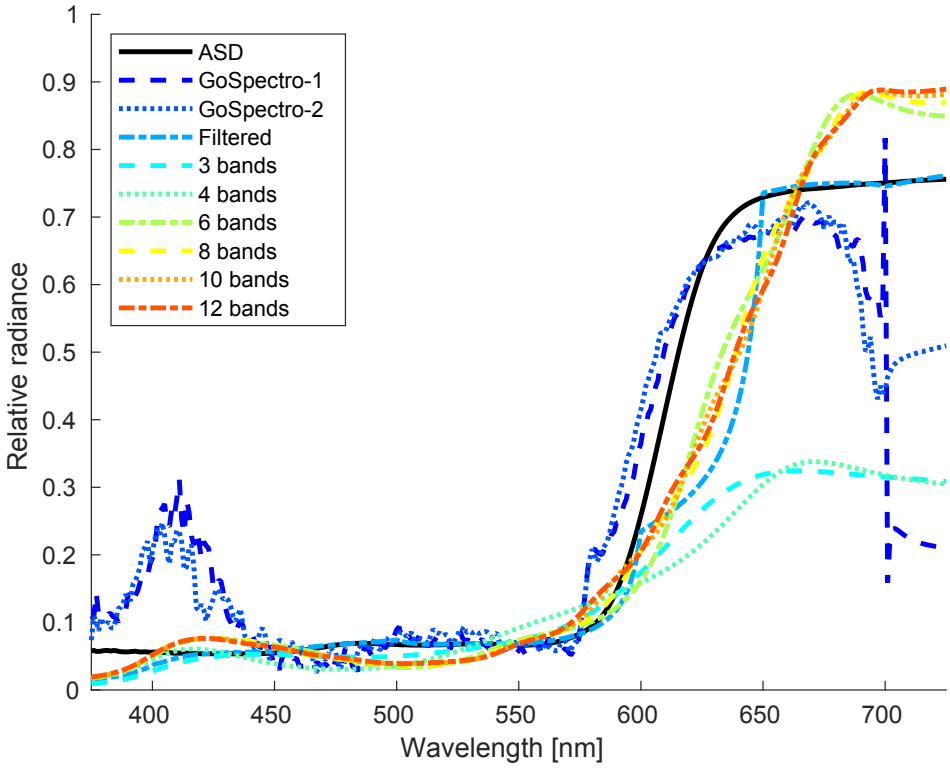


Figure 5.19: A comparison of spectral reflectances measured using spectrometers, or extracted from spectral images, for Patch 15 (Red) of our X-Rite 24-patch ColorChecker CLASSIC colour calibration chart. The curves on the plot have the same interpretation as those in Figure 5.17.

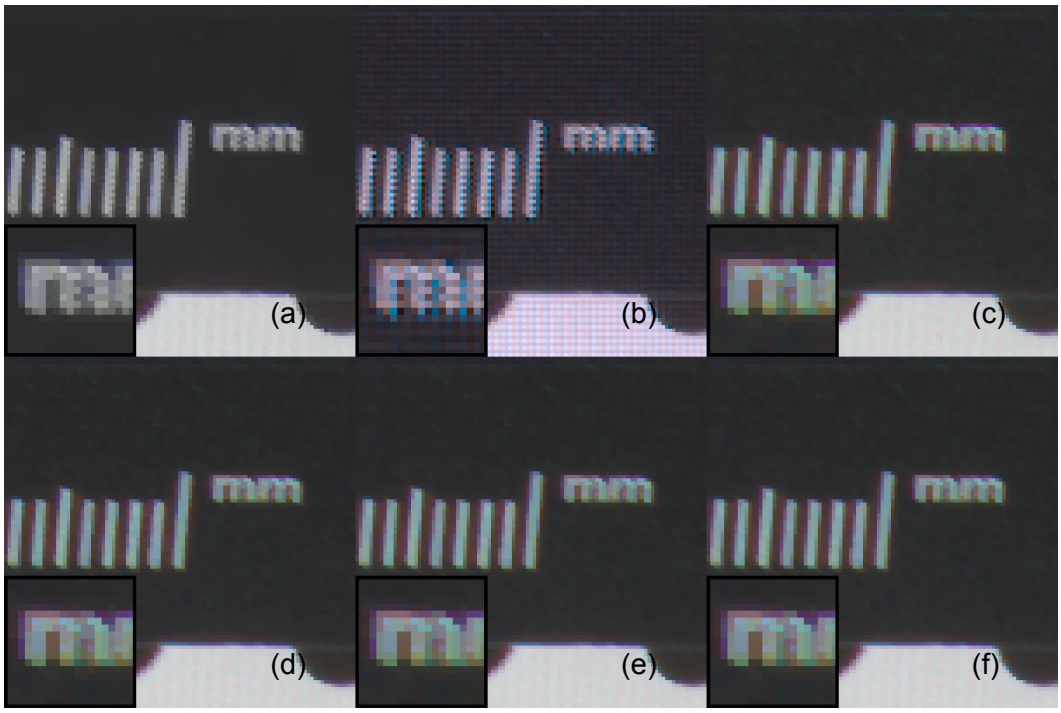


Figure 5.20: A comparison of images produced using our spectral image estimation algorithm with the `L1` combination of penalties (Table 3.1) and a non-negativity constraint, for different numbers of spectral control bands in the range 386 to 707 nm: (a) 3 bands, (b) 4 bands, (c) 6 bands, (d) 8 bands, (e) 10 bands, and (f) 12 bands.

in black and white regions of the image. In contrast, in regions with saturated colours, low numbers of bands produce severe colour distortion, as shown in Figure 5.22. Even though 3 control bands provides the same number of degrees of freedom as a (demosaiced) colour image, such a small number of bands does not allow for adequate colour reproduction, perhaps because the estimated image is non-negativity constrained.

Interestingly, Oh *et al.* [87] determined the space of colour channel sensitivity functions across cameras to be approximately 8-dimensional. Therefore, by choosing an eight-control band spectral image representation, we are using the highest-dimensional representation of spectral radiances that is likely to be recoverable from RGB cameras (although, technically, from multiple RGB cameras simultaneously imaging the same scene to provide at least 8 colour channels) [87].

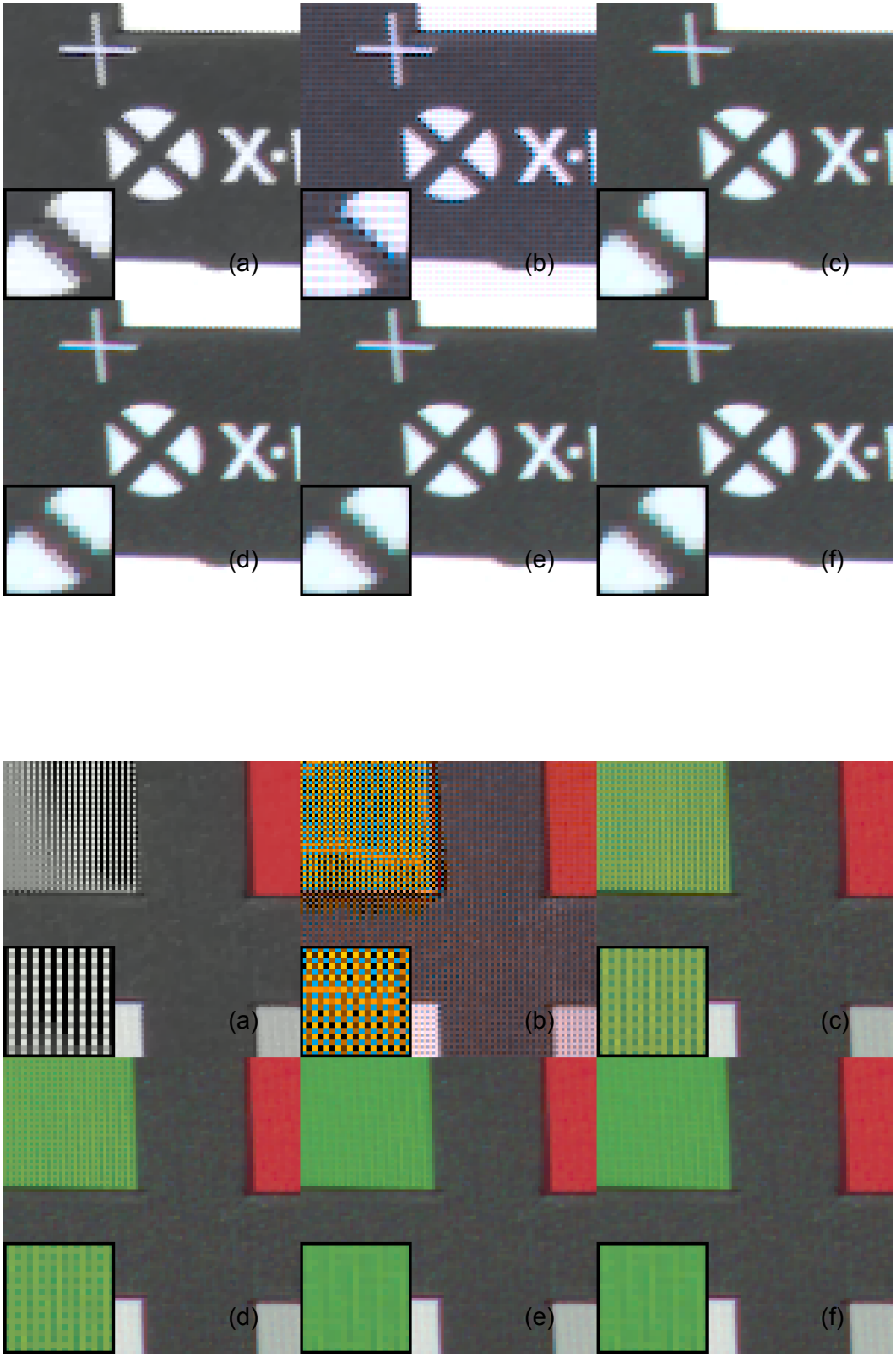


Figure 5.22: A version of Figure 5.20 for an image patch with saturated colours

5.4.4 Weights on Regularization Penalties

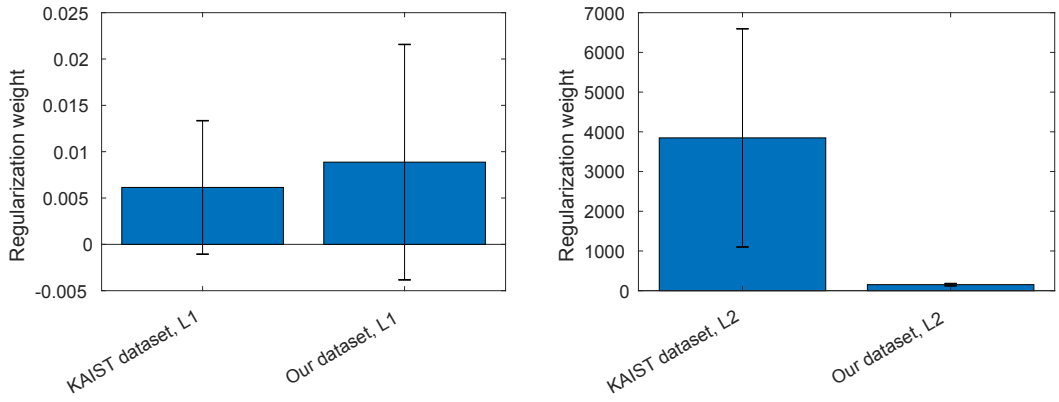
Another pertinent question about our experiments is whether or not our regularization weights should be fixed. We illustrated in Section 5.2.4 that different regularization weight selection criteria produce significant differences in image estimation accuracy. Is it feasible, however, to select regularization weights in advance, rather than from the current image's content?

Unfortunately, we do not have a clear answer to this question, because of the limited number of images in our experiments, with regularization weights selected from one patch for each image. An experiment in which we select regularization weights from many patches across many images, and then evaluate image estimation under each set of regularization weights, is prohibitively time-consuming.

In Figure 5.23, we show that regularization weights vary significantly, potentially with both the input image patch, and with image capture conditions that differ between datasets (*e.g.* image sensor resolutions). The variability of regularization weights depends on the regularization penalties for which the weights are being selected, as we see by comparing Figures 5.23a and 5.23b.

5.5 Image Reconstruction from Synthetic RGB Images

While the experiments in Section 5.5 illustrate how our method may perform in practice, our quantitative evaluation was limited because we did not have ground truth spectral images. In this section, we evaluate chromatic aberration correction and spectral reconstruction on our dataset of multispectral images, which we synthesized as described in Section 4.2.4. We use our multispectral images as ground truth, and use raw colour images that we simulated from them as input to the image estimation algorithms. We then configured the operators in our data-fitting term in our optimization problem (3.1) to match the ones used to simulate the raw input images. For the spectral control bands of our estimated spectral images, we use the same 7 spectral bands as the multispectral images (400, 450, 500, 550, 600, 650, and 700 nm).



(a) (b)

Figure 5.23: A comparison of regularization weights selected using our DMC between the KAIST dataset, and our own dataset of mosaiced RGB input images. (a) Regularization weights selected for the `L1` set of regularization penalties (Table 3.1), in combination with a non-negativity constraint. (b) Regularization weights selected for the `L2` set of regularization penalties (Table 3.1), in combination with a non-negativity constraint. The bars in the graphs illustrate the mean weights across the 4 images for the KAIST dataset, or the 7 images from our dataset, where each weight was selected using one 128×128 patch per image. Error bars are one standard deviation in length.

Given that our multispectral images are still of relatively low spectral resolution, and were captured under illumination conditions that we are unable to fully validate (as discussed in Section 4.2.1), we feel that our results on RGB images captured under unfiltered light (Section 5.5) better characterize real-world performance. In this section, we instead strive to understand the numerical behaviour of our method, now that we can clearly observe it under a known spectral-to-colour image formation model. In particular, our input images are unaffected by image sensor nonlinearity and pixel crosstalk, effects that we do not account for in our image estimation algorithm. For studying performance under these ideal conditions, we felt it is unnecessary to examine all variants of our algorithm. We selected only the `L2` set of regularization terms (Table 3.1), in combination with a non-negativity constraint, as this variant of our algorithm runs relatively quickly, and the effect of its regularization term is simple to interpret. We used two criteria for selecting regularization weights: Error with respect to the true spectral or colour image, MSEC (3.23), and similarity with bilinear interpolation, DMC (3.24).

5.5.1 Ideal Chromatic Aberration Correction

For our models of dispersion in this experiment, we used the same models as in Section 5.4, shown in Figures 5.3 and 5.4, for spectral image and colour image estimation, respectively. While the spectral-to-colour conversion component of colour image formation is known in this experiment, the true spectral dispersion is unknown. Moreover, the model of colour channel warping (Figure 5.4) was calibrated from images captured under unfiltered light, not from colour images simulated from the multispectral images. We decided against calibrating the model of colour channel warping from simulated colour images, to avoid artifacts from the non-uniform illumination present in the spectral bands of the multispectral images.

We need to justify that both our colour and spectral models of dispersion are appropriate for our multispectral image dataset. In Figure 5.24 (b) and (d), we show image unwarping using the colour and spectral models of dispersion, respectively. The colour model of dispersion produces reasonable results with barely visible purple fringing, but the spectral model of dispersion produces the sharpest edges, and does not leave any colour fringes. The same observations are reflected in our quantitative evaluation of chromatic aberration correction, shown in Tables 5.9 and 5.10 (analogous to Tables 5.5 and 5.6, respectively, presented in Section 5.4.2). In Table 5.10, correction using the colour model of dispersion ("True image, colour-unwarping") reduces the error relative to the original image, but not as much as does correction using the spectral model of dispersion ("True image, spectral-unwarping"). We are pleased that our visual evaluation agrees with our quantitative evaluation, and that our spectral model of dispersion outperforms our colour model of dispersion. Moreover, because our colour model of dispersion does not leave significant colour fringes (Figure 5.24 (b)), we feel that it has been properly calibrated. In summary, our results validate our method for calibrating dispersion, in particular by showing that dispersion is best calibrated in the spectral domain. Our results also validate our method for quantitatively evaluating residual chromatic aberration, because it leads to the same conclusions as our

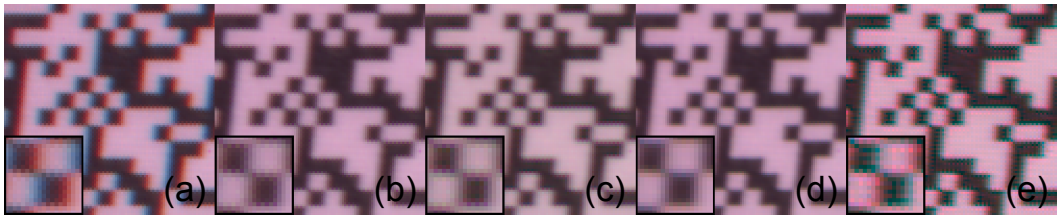


Figure 5.24: A comparison of our image estimation algorithm with chromatic aberration correction by warping the true image. We perform the comparison on a patch from the image shown in Figure 4.15g. (a) The input colour image (before mosaicing); (b) Unwarping colour channels in the true colour image using the colour channel-based model of dispersion shown in Figure 5.4; (c) Direct colour image estimation (Section 3.9.1) with the regularization weight selected using the DMC (3.24); (d) Unwarping bands in the true spectral image using the model of spectral dispersion shown in Figure 5.3; (e) Spectral image estimation with the regularization weight selected using the DMC (3.24)

Table 5.9: Spectral MRAE and RMSE (4.4) in the 12 pixel-wide regions within ColorChecker patch edges, averaged across patches. Entries are sorted by MRAE.

Condition	$MRAE$	$RMSE_{\text{pixel}}$
True image, spectral-unwarping	**0.3328**	**0.1021**
True image, uncorrected	0.3400	0.1080
DMC	0.4467	0.1588
MSEC	0.4768	0.1727

visual evaluation.

5.5.2 Image Estimation Evaluation

As shown in Table 5.10, our colour image estimation algorithm ("Colour, MSEC") and ("Colour, DMC") corrects chromatic aberration almost as well as when the model of colour dispersion is applied to the true colour image ("True image, colour-unwarping"). The similarity between the two results holds across all colour channels.

In contrast, while our spectral image estimation algorithm, with regularization weights selected using the DMC ("Spectral, DMC" in Tables 5.9 and 5.10) does restore the Green channel, its errors on the other two colour channels are high, as is its spectral error (Table 5.9). With the MSEC, it performs even

Table 5.10: MRAE and RMSE in each colour channel in the 12 pixel-wide regions within ColorChecker patch edges, averaged across patches. Entries are sorted by error in the Green channel, then in the Red channel. Note that Rudakova and Monasse [99], and Sun *et al.* [111] (without Krishnan *et al.* [61] as a preprocessing step) do not alter the Green channel, and so have the same error in Green as the demosaicing methods they follow. The colour variant of our method (Section 3.9.1) reconstructs colour images directly, instead of estimating spectral images.

Condition	$MRAE \times 10^2$			$RMSE \times 10^2$		
	Red	Green	Blue	Red	Green	Blue
True image, spectral-unwarping	10.16	**7.92**	**8.88**	0.297	**0.700**	**0.274**
Spectral, DMC	23.58	8.15	14.46	0.543	0.723	0.353
ARI [79]	10.29	8.55	9.98	0.317	0.816	0.372
ARI [79] & warping [99]	10.75	8.55	9.16	0.342	0.816	0.311
ARI [79] & CCT [111]	11.73	8.55	10.93	0.357	0.816	0.378
Colour, MSEC	10.86	8.59	9.31	0.342	0.816	0.318
True image, colour-unwarping	10.68	8.62	9.25	0.338	0.817	0.313
Bilinear	10.49	8.62	10.19	0.323	0.817	0.378
Bilinear & warping [99]	10.87	8.62	9.34	0.343	0.817	0.318
Bilinear & CCT [111]	11.53	8.62	10.60	0.351	0.817	0.369
Colour, DMC	10.99	8.65	9.31	0.345	0.819	0.320
Simulated raw image	**9.15**	8.84	10.27	0.303	0.825	0.400
True image, uncorrected	10.21	9.34	11.58	**0.294**	0.894	0.452
Spectral, MSEC	16.57	18.58	21.05	0.398	1.341	0.435

worse, even though this criterion is designed to minimize the error with respect to the true spectral image. It is possible that the regularization weight search converged on a non-global minimum of the MSEC, or that other image patches would have produced better regularization weights, but such explanations do not adequately explain why the MSEC performed poorly in all images, not only on the image evaluated in the tables.

With both regularization weight selection criteria, we observe residual colour fringes and residual colour-filter array patterns in the output images, as shown in Figure 5.25. The regularization weights selected using the MSEC were higher than those selected using the DMC, and the residual fringes for the MSEC tend to be thicker than those for the DMC. Our experiment supports our hypothesis from Section 5.4.2 that residual colour fringes are artifacts from regularization, for two reasons: First, different weights on the regularization terms alter the thickness of the fringes. Second, we are using the ground truth spectral-to-colour image conversion model during image estimation, and our models of dispersion produce good results on ground truth images (Figure 5.25 (a) and (d)). In other words, there are no sources of error in the data-fitting term of our optimization problem (3.1) that could account for the residual fringes.

To illustrate how the residual fringes arise, in Figure 5.26, we plot the spectral intensities of an image along a line passing from white to black, and then to white again. In Figure 5.26a, we observe that the largest error is an underestimation of intensity in the 500 nm band, which is the band to which the Green channel is most sensitive compared to the other colour channels. All other spectral bands are given higher intensities to compensate. As a result, our algorithm has recovered spectra that are metamers of the true spectra. Dispersion reveals the metamers through colour fringes, because redistribution of spectral intensities between bands changes the spatial distribution of intensity in the dispersion-corrected image. The residual fringes are visible as a shift to the right of the estimated Red channel relative to the ideal Red channel in Figure 5.26b.

Aside from the global metamerism in Figure 5.26, the image edges have

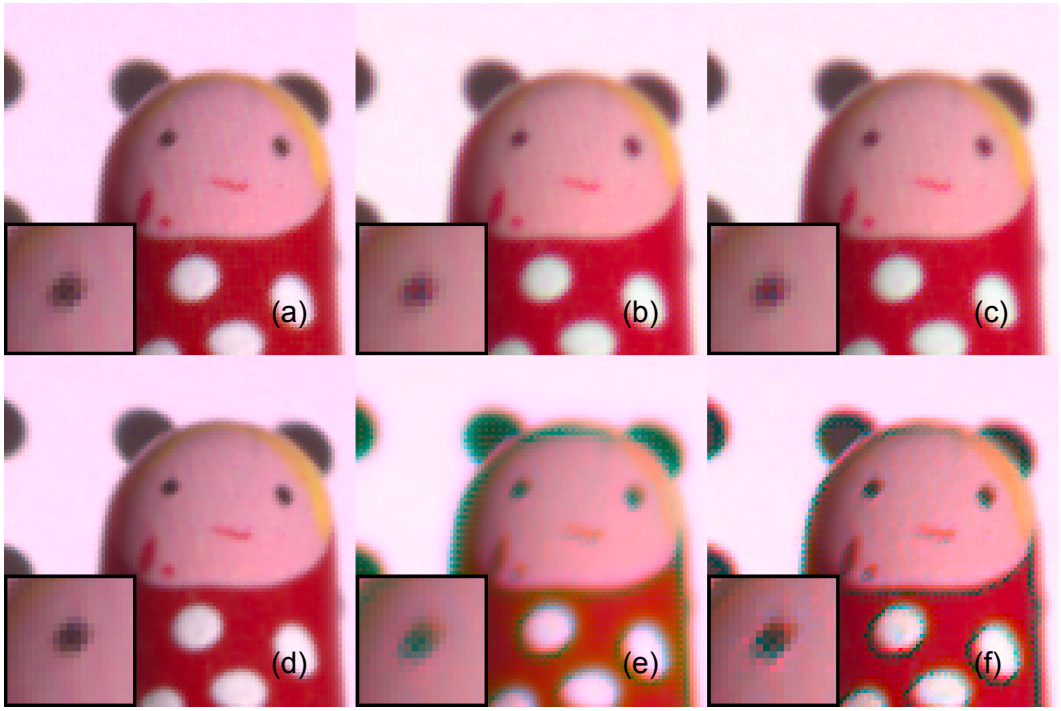
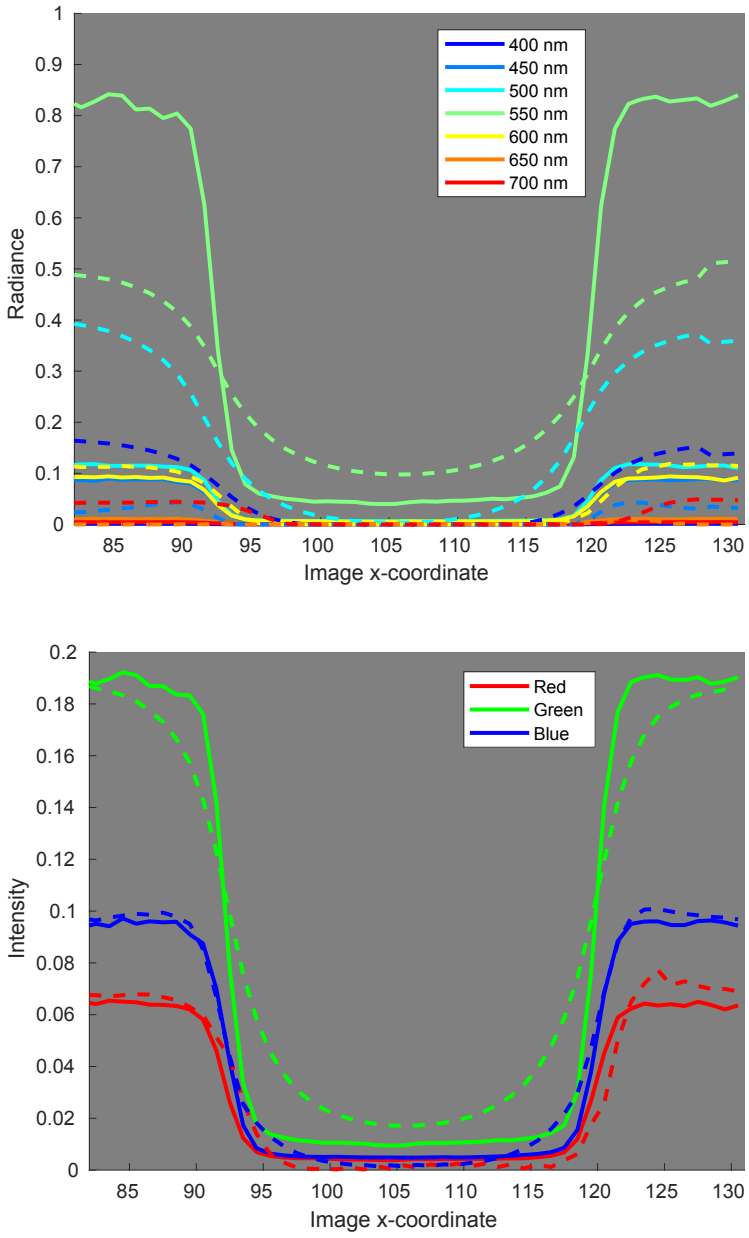


Figure 5.25: A further comparison of our image estimation algorithm with ideal image warping, on a patch from the image shown in Figure 4.15d. (a) Unwarping colour channels in the true colour image using the colour channel-based model of dispersion shown in Figure 5.4; (b) Direct colour image estimation (Section 3.9.1) using the MSEC, or (c) the DMC, for selecting regularization weights; (d) Unwarping bands in the true spectral image using the model of spectral dispersion shown in Figure 5.3; (e) Spectral image estimation using the MSEC, or (f) the DMC, for selecting regularization weights.

been oversmoothed by the regularization term, because of its excessive weight. The degree of smoothing is intensity-dependent, unfortunately, which further exacerbates the residual fringes that appear in the image.

The results shown in this section, for the `L2` set of regularization terms (an L2-norm penalty on the image spatial gradient) illustrate how regularization of the image spatial gradient produces predictable artifacts in the estimated spectral image: First, high-intensity spectral bands are underestimated, because lower intensities tend to yield lower image gradients, favoured by the regularization penalty. Second, sharp edges are oversmoothed. The estimated spectral image still approximately projects to the input image, but the estimated spectral image a metamer of the true spectral image. Unfortunately, even though we understand how the metamers differ from the true spectral radiances, and could predict what metamers would occur given the true spectral radiances, nothing in the structure of the metamers allows us to do the reverse and predict what the true spectral radiances should be. To overcome metamerisms, we would need some additional information, such as known structure in the scene serving as a reference for edge locations in the image.



(b)

Figure 5.26: Image intensities produced by our spectral image estimation algorithm (dashed curves), compared with the true multispectral image (solid curves). The true image was corrected for chromatic aberration using the model of spectral dispersion shown in Figure 5.3. Our algorithm used the `L2` regularization penalty (Table 3.1), as well as a non-negativity constraint. The MSEC (3.23) was used to select the weight on the penalty. Image intensities are plotted for a line which crosses from a white background to a black disk, and then back to the white background. (a) Spectral intensities for the seven spectral bands in the multispectral image (corresponding to the seven optical bandpass filters); (b) The corresponding colour channel intensities.

Chapter 6

Discussion

In Chapter 5, we presented experimental results to validate our method for simultaneous demosaicing and chromatic aberration correction through spectral reconstruction. Our analysis focused on specific details of our method, whereas in this chapter, we discuss some broader questions pertaining to our approach.

6.1 Chromatic Aberration and Image Quality

An important question to revisit is whether there is value in attempting to correct chromatic aberration. There are two parts to this question. First, is chromatic aberration detrimental to image quality to the point where it should be corrected? The effect of chromatic aberration depends on both the scene being viewed, and on the imaging system. The lenses we tested had a moderate amount of chromatic aberration, which produced highly noticeable colour fringes, depending upon the location in the image plane (*e.g.* Figure 5.6 (a) and (f)). For colour fringes to be observed, however, the scene must possess strong edges, and the image resolution must be sufficiently high. Finally, the image must be viewed by a human observer, or processed by a machine vision system, at the level of its smallest details. Otherwise, chromatic aberration is unimportant. Additionally, for human observers, the colour correction applied to the image determines whether chromatic aberration appears more as blurred edges, or as colour fringes, as we demonstrated in Figure 4.16. We consider colour fringes to be the more objectionable artifact.

Given the conditions under which chromatic aberration should be corrected, we can consider the second part of the question: Does our algorithm for correcting chromatic aberration achieve a sufficient benefit in relation to its drawbacks? Our algorithm is computationally expensive, because it uses global optimization. There is little benefit in relation to the time spent running it on large images that are to be used at low magnification, because chromatic aberration is not visible under these conditions. Moreover, for our spectral image estimation algorithm, the final image may have a global colour cast because of slight colour misestimation caused by our regularization terms (Section 5.5.2). When images are to be used under high magnification, however, we argue that our method is worthwhile, as it improves visual quality both by deblurring slightly, and by reducing colour fringing. Our results are also free from artifacts such as zippering, common to demosaicing algorithms [79], and ringing, common to deblurring algorithms [111]. Moreover, the computational cost will be reasonable if only small portions of the image are to be reconstructed for further processing or for display.

6.2 Alternative Models of Dispersion

We chose to model only lateral chromatic aberration, which is a spatial warp between images formed under different wavelengths of light (Section 1.1.2). Therefore, we ignore longitudinal (axial) chromatic aberration, which encompasses all other optical aberrations that depend on wavelength, but which is usually modelled as a relative blur between images formed under different wavelengths [45].

There are several reasons why we chose to ignore longitudinal chromatic aberration. First, in contrast to lateral chromatic aberration, longitudinal chromatic aberration depends on the lens aperture, and a small aperture will reduce it considerably [60]. Second, lateral chromatic aberration also manifests as blur, because of the superposition of wavelengths within each colour channel, so modelling longitudinal chromatic aberration is not necessary to correct some of the blur in the image. Third, we consider longitudinal chromatic aberration

to be much more difficult to model and calibrate. Fourth, it is also difficult to remove the effects of blur from an image (such as blur from longitudinal chromatic aberration), without introducing artifacts such as ringing [81].

With respect to our third reason above, calibrating blur requires point spread function estimation, which is an ill-posed problem [82].[1] Blur is, in general, spatially-varying, and arises from longitudinal chromatic aberration, other aberrations [10], and geometric defocus [73]. Even if one could calibrate longitudinal chromatic aberration independently of other sources of blur, to correct longitudinal chromatic aberration, one would need to sample it at arbitrary locations in the image. To do so, one would need a model of how the point spread function varies in space. To be well-behaved, such a model would require choosing a parametric form for the point spread function [50], therefore introducing representation bias.

We note that our present algorithm must sample the model of chromatic aberration at arbitrary wavelengths, not only at arbitrary positions in space, to accommodate any spectral basis we use to represent our estimated images. Our models of lateral chromatic aberration are well-behaved under sampling at arbitrary spatial-spectral positions, because the x and y-components of dispersion vectors are smooth functions of spatial-spectral coordinates, as we showed in Section 5.3. Achieving likewise with a model of longitudinal chromatic aberration would require designing a smooth spatial-spectral parametric representation for point spread functions. Finding such a representation that also adequately models the aberrations of a variety of lenses may be a challenge.

6.3 Spectral Response Calibration

We highlighted the difficulty of camera spectral response calibration in Section 3.3.1. An important question is how sensitive our method is to errors in the camera spectral response functions. We distinguish between two types of

[1] Delbracio *et al.* [23] argued that it is a well-posed problem, given an appropriate calibration pattern, but Mosleh *et al.* [82], and Jemec *et al.* [51] respectively observe that noise, and moderate amounts of blur, render the calibration pattern of Delbracio *et al.* [23] ineffective.

errors:

(i) Incorrect relative spectral responses between colour channels at a given wavelength.

(ii) Incorrect scale of all colour channel response functions at a given wavelength. In other words, all colour channels response functions are scaled by the same factor relative to their true values.

Error of type (i) will severely degrade our results, because our algorithm will shift intensities between wavelengths in the estimated spectral image, changing the chromatic aberration correction. Unfortunately, type (i) error is quite common, because camera spectral response functions are often determined only up to a per-channel unknown scaling factor. Such is the case, for example, for the spectral sensitivities measured by Darrodi *et al.* [22] (Figure 3.5). While we used the Nikon D5100 spectral sensitivities measured by Darrodi *et al.* [22], our results are fortunately not affected by the scale uncertainty, as we only used these spectral sensitivities to simulate RGB images from known spectral images (Section 4.3.1).

Error of type (ii) is less detrimental. If the channel response functions are all scaled by a given factor at a given wavelength, the spectral image can be scaled by the inverse factor at that wavelength in order to produce the same colour image as without any error in the channel response functions. While the spectral image will now be less accurate, there will be no change in the colour version of the estimated image. Chromatic aberration correction will be the same as without the error, because it depends only on the product of the camera's spectral response with the intensity of the spectral image at each wavelength, rather than depending directly on the spectral image.

In reality, error of type (ii) will affect our estimated colour images, because the action of our regularization terms depends on the relative intensities of the spectral image across the visible spectrum. Specifically, spectral bands with higher intensities tend to have larger spatial gradient magnitudes, and therefore are smoothed more aggressively by our regularization terms. We would want the spectral image to have a uniform intensity across the spectrum

in order for our penalties on the image spatial gradient to penalize edges equally in all spectral bands.[2] We might therefore want to express the spectral image as relative to a reference spectral intensity, such that the spectral image is maximally uniform across the spectrum. One approach is to compute the average RGB colour of the input image, and set the reference spectral intensity to be a smooth spectral intensity projecting to this RGB colour. For example, we might solve the following optimization problem:

$$\underset{\mathbf{s}}{\operatorname{argmin}} \|\boldsymbol{\Omega}(\mathbf{s}) - \mathbf{c}\|_2^2 + w\nabla_\lambda \mathbf{s}\,, \quad \mathbf{s} \geq 0 \tag{6.1}$$

where $\mathbf{s}$ is the reference spectral intensity, $\mathbf{c}$ is the mean RGB colour of the image, and w is the weight on the spectral regularization penalty term. We leave a detailed evaluation of this approach to future work.

6.4 Comparison with "Compact single-shot hyperspectral imaging using a prism"

In Section 2.3.1, we mentioned that our algorithm is inspired by the work of Baek *et al.* [7], who reconstructed spectral information from an RGB image distorted by a triangular prism. Their method has five stages:

(i) Estimate an initial spectral image, by solving the following global optimization problem using the Alternating Direction Method of Multipliers method [1]:

$$\mathbf{I}_1 = \underset{\mathbf{I}}{\operatorname{argmin}} \|\boldsymbol{\Omega}(\boldsymbol{\Phi}(\mathbf{I})) - \mathbf{I}_{\text{raw}}\|_2^2 + w_1 \|\nabla_{xy}\mathbf{I}\|_1 + w_2 \|\nabla_\lambda \nabla_{xy}\mathbf{I}\|_1 \tag{6.2}$$

where $\boldsymbol{\Phi}$ is an operator modelling the effect of dispersion from a prism placed in front of the camera lens, and $\boldsymbol{\Omega}$ is an operator converting spectral radiances to the raw colour space of the camera. It is not clear from their article whether the input image, $\mathbf{I}_{\text{raw}}$, has been demosaiced.

(ii) Obtain a colour version of the estimated image, $\boldsymbol{\Omega}(\mathbf{I}_1)$, and detect edges in the colour image using the method of Dollár and Zitnick [25].

[2] Setting per-band weights on the spatial gradient penalties would be an alternative solution.

(iii) Solve an optimization problem to reconstruct the spatial gradient of the spectral image from the spatial gradient of the input image:

$$\mathbf{g}_{xy} = \underset{\mathbf{g}_{xy}}{\operatorname{argmin}} \|\mathbf{\Omega}\left(\mathbf{\Phi}\left(\mathbf{g}_{xy}\right)\right) - \nabla_{xy}\mathbf{I}_{\text{raw}}\|_2^2 + w_3 \left\|\nabla_{\lambda}\mathbf{g}_{xy}\right\|_1 + w_4 \left\|\nabla_{xy}\mathbf{g}_{xy}\right\|_2^2 \tag{6.3}$$

Only pixels in the edges identified in Step (ii) have their spectral gradients reconstructed. As in Step (i), they solve the optimization problem using the Alternating Direction Method of Multipliers method [1].

(iv) Reconstruct the spectral image from its spatial gradient, by solving a third optimization problem:

$$\mathbf{I}_4 = \underset{\mathbf{I}}{\operatorname{argmin}} \|\mathbf{\Omega}\left(\mathbf{\Phi}\left(\mathbf{I}\right)\right) - \mathbf{I}_{\text{raw}}\|_2^2 + w_5 \left\|\mathbf{W}_{xy} \odot \left(\nabla_{xy}\mathbf{I} - \mathbf{g}_{xy}\right)\right\|_2^2 + w_6 \left\|\Delta_{\lambda}\mathbf{I}\right\|_2^2 \tag{6.4}$$

where $\Delta_{\lambda}\mathbf{I}$ is the spectral Laplacian gradient of the image, and $\mathbf{W}_{xy}$ is a matrix of confidence weights that depends on the image gradient magnitude, and on the direction of dispersion. As (6.4) is a least-squares problem, they solve it using a conjugate gradient method.

(v) Produce the final spectral image by guided filtering [40] of the result from (6.4) with $\mathbf{\Omega}\left(\mathbf{I}_1\right)$ as the guidance image.

Baek *et al.* [7] present excellent results, but unfortunately did not provide their implementation of their method. Our image reconstruction algorithm solves a problem similar to (6.2), but we tested a variety of prior terms (Section 3.5), as well as a non-negativity constraint on the estimated spectral image. Had we incorporated the remaining steps of their method, we might have achieved better results, such as without the residual colour fringes that we discussed in Section 5.4.2.

There are several reasons we decided against testing their full method. For one, the time required to implement and run their method would have been considerable, given the number of global optimization steps it entails. More importantly, their method is severely challenged by images that either lack strong edges, or have edges that are separated by less than the spatial extent

of dispersion [7]. They also mention that fine structures in the image tend to be lost during the optimization steps —They included the final guided filtering step (v) in an attempt to restore image details [7]. We are more interested in restoring high-frequency content in the image than in evaluating spectral reconstruction in broad patches of colour, so we consider this last problem with their method to be a severe limitation.

Regardless, their approach remains inspiring, because they made spectral reconstruction from RGB images a better-posed problem. In particular, they did so in a more flexible way than a spectrometer, or the system of Cao *et al.* [16], for example, which are "brute force" methods that prevent the spectra of different regions in the scene from overlapping on the light sensor. Instead, Baek *et al.* use edge locations (Step (ii)) to obtain a one-to-many mapping from edge pixels in the ideal spectral image to pixels in the dispersion region around the edge in the input image. This one-to-many mapping compensates for the many-to-one mapping of spectral radiances to RGB colours, therefore making spectral reconstruction well-posed.

In our view, the primary disadvantage of their edge-based spectral reconstruction is the need to make hard decisions about which pixels belong to edges. A second disadvantage is the need to reconstruct images in multiple stages, which may accumulate error in comparison to a single joint optimization [43, 115]. We wonder if a careful choice of image priors, or of a representation for the spectral image, may make explicit edge identification unnecessary, and allow for a single stage algorithm. For instance, Pan *et al.* [89] presented a method for blind deblurring that uses $L0$-regularization to avoid relying on sharp edge detection.

Chapter 7

Future Work

In this chapter, we continue our discussion, from Chapter 6, of ideas for future work, but extend it from ideas closely-connected with our algorithm to more general research directions.

7.1 Calibration

7.1.1 Camera Spectral Response Calibration

We mentioned in Section 3.3.1 that camera spectral response functions are difficult to calibrate, especially without professional equipment. Calibration methods which allow broadband illumination to be used are limited in accuracy because the space of spectral reflectances of real objects is approximately six- to nine-dimensional [30]. Whereas the visible spectrum is infinite-dimensional, we use low-dimensional representations of spectral images, as discussed in Section 3.3.2. Therefore, we recommend experiments to calibrate camera spectral response functions only to the spectral resolution required for reliable correction of chromatic aberration by our image estimation algorithm. Recovery of spectral response functions with six to nine degrees of freedom, matching the dimensionality of real-world reflectances, should be sufficient, as we used a number of spectral bands in this range (as justified in Section 5.4.3). We believe that many calibration methods should be able to do so without sophisticated equipment. Such low-cost camera response calibration methods will allow our image estimation algorithms to be used more widely.

Aside from reducing the equipment and time required for camera spectral

response calibration, we might also want to augment spectral response calibration by also calibrating crosstalk. As we mentioned in Section 2.1, correcting crosstalk is an under-studied problem, and calibrating crosstalk is even less-commonly discussed in computer vision literature. Still, modelling crosstalk may be important for accurate spectral reconstruction from RGB images, and warrants further investigation.

7.1.2 Dispersion

Generalization across Imaging Conditions

In our experiments, we were careful to use the same focus and aperture settings to capture all images. Light propagation through the lens depends on all geometric and material properties of the lens, and therefore changes to the lens settings will alter dispersion. Future work is needed to quantify the sensitivity of dispersion to variation in lens settings, and to create models of dispersion which can accommodate different lens settings.

The point spread function estimation method of Shih *et al.* [105] might be adapted to produce flexible models of dispersion. Provided they had access to the lens design, they calibrated the point spread function of the lens at one lens setting, and used the point spread function to adjust the parameters of the lens design to better model the physical lens. From the adjusted lens design, they could then simulate point spread functions for other lens settings.

Generalization across Scenes

Aside from lens settings, dispersion is also affected by the depth of objects in the scene. We conducted preliminary raytracing experiments, outside the scope of this thesis, to model the effect of depth on dispersion in a simple lens. Based on the results of these raytracing experiments, we conjecture that the dependence of lateral chromatic aberration on depth weakens rapidly at depths beyond a few focal lengths. Similarly, Baek *et al.* [7] showed that dispersion caused by a triangular prism, which is much more severe than the dispersion in most lenses, is effectively constant with respect to depth at depths larger than 70 cm, and changes significantly with respect to depth only up to depths

of around 20 cm. Therefore, we doubt it is necessary to account for depth when modelling dispersion, except in applications, such as microscopy, where objects are very close to the lens.

Low-Spectral Resolution Calibration

As with camera spectral response calibration (Section 3.3.1), our dispersion calibration procedure also relies on specialized equipment, specifically a system to filter light to narrowband illumination (Section 4.2.1). As a lower-cost alternative to optical bandpass filters, we recommend testing LEDs, which others have used for inexpensive spectral imaging [36, 90]. As the illuminants become more broadband in spectrum, however, modifications to our dispersion calibration procedure may be necessary. We recommend calibrating point spread functions under each illuminant, from which one can likely extract the ranges of dispersion vectors corresponding to the spectral ranges of the illuminants. Any single vector, such as measured using a disk keypoint displacement (Section 3.2.1) will poorly characterize the dispersion pattern produced by a broadband illuminant. Unfortunately, mapping point spread functions to spectral dispersion vectors may be difficult because of variations in the direction of dispersion with wavelength. As we illustrated in Figure B.4, the projection of dispersion vectors along a given direction may be non-monotonic with respect to wavelength.

7.2 Spectral Reconstruction

7.2.1 Regularization

Adaptive Regularization

We cited the optimization-based demosaicing algorithm of Menon and Calvagno [77] in Section 2.1 in particular for their use of adaptive regularization. Presently, our regularization terms are given global weights across the image, although regularization which adapts to local image content may give better results. Whereas Menon and Calvagno [77] used the result of their first iteration to calculate local regularization weights, we might be able to set local

regularization weights in advance, using our models of dispersion. For example, we could try penalizing image gradients more strongly in the direction perpendicular to the direction of lateral chromatic aberration, with the strength of the penalty depending on the magnitude of the aberration. Adaptive regularization may also improve computational efficiency, by allowing us to achieve good results with L2-norm penalty terms, which simplify optimization relative to L1-norm penalty terms.

Alternative Image Priors

In our work, we used simple L1 and L2-norm priors on the image gradients (Table 3.1), but there is a large variety of possible priors that could be tested, including cross-band priors [42], L0-norm priors [89], and normalized sparsity priors [61]. Most of these priors would make the image estimation optimization problem more difficult to solve, unfortunately.

An approach that would potentially allow us to improve our results under our simple image priors may be intrinsic image decomposition, which has recently been extended to spectral images [49, 55, 56]. Presently, we apply our smoothness priors to the combination of reflectance and shading that forms the reconstructed spectral image. Yet the statistical properties of reflectance and shading differ [103], so the two should be subject to different regularization penalties, or to the same regularization penalty, but with different weights. Therefore, it may be beneficial to separate reflectance from shading by integrating intrinsic image decomposition into our method. In fact, our suggestion in Section 6.3 of expressing the spectral image as relative to a reference spectral intensity is equivalent to decomposing the image into a reflectance image, and a global illumination spectral power distribution, where the latter is given by the reference spectral intensity.

Alternative Image Representations

Our estimated images are represented in terms of pixels, and we use smoothness priors on the image spatial gradient to encourage continuity between pixels. In contrast, in the spectral domain, we have embedded smoothness into

our image representation by using a basis of Gaussian functions to represent spectral radiances (Section 3.3.2). As such, we do not need a regularization term penalizing spectral roughness, provided that we choose a low-dimensional or sparse basis.

We recommend experimenting with spatial basis representations as well as spectral basis representations. For example, Parmar *et al.* [91] used a 2D Haar wavelet basis to represent spatial information, and a learned KSVD basis to represent spectral information [91]. Their separate representations for spatial and spectral information might be reasonable considering that the spatial and spectral dimensions are incomparable. Still, combined spatial-spectral bases may also be worth investigating, such as the 3D sparse dictionary representation of Lin *et al.* [66].

7.2.2 Data-Driven Spectral Reconstruction

Aside from using training data to create representations of spectral images for regularization (Section 7.2.1), we may want to leverage learned mappings from RGB colours to spectral signatures. We note that data-driven spectral reconstruction algorithms (Section 2.3.2) are often single-pass algorithms, rather than iterative optimization algorithms, so integrating them could lead to a faster image reconstruction method.

One fast approach to chromatic aberration correction would be to use a data-driven spectral reconstruction method to obtain a spectral image, and then correct chromatic aberration in the spectral image by warping the image according to a model of dispersion. In addition to the difficulty of obtaining adequate training data, which we discussed in Section 3.3.2, such an approach would face two further challenges: First, the input image is a colour-filter array image, whereas most data-driven RGB-to-spectral reconstruction methods require full-colour images as input. If we used a demosaicing method as a preprocessing step, we would introduce artifacts from demosaicing, and potentially complicate chromatic aberration correction, as we discussed in Section 2.2. Alternatively, we could train the RGB-to-spectral mapping to operate on mosaiced images, but it may be difficult to learn a well-behaved mapping,

because colour-filter array images have significant high-frequency content.

Second, chromatic aberration mixes spectral radiances from neighbouring points in the scene. Consequently, at edges in the image, the corresponding spectral intensities no longer satisfy the assumptions made by many RGB-to-spectral methods that spectral radiances are well-modelled by a low-dimensional manifold [53] or by a sparse representation [4]. A possible solution may be to use a chromatic-aberration-corrected image as the input to the RGB-to-spectral mapping. We envision an iterative approach, wherein the output image of each iteration of chromatic aberration correction is the input to the next iteration. Alternatively, we could incorporate the RGB-to-spectral mapping within a global optimization algorithm, as done by Choi *et al.* [19], but then the computational cost may be significant (as discussed in Section 3.8.2). A more efficient approach may be to use the initial output of the RGB-to-spectral mapping as a fixed prior in our existing optimization algorithm, but weighted so that it has less influence at image edges.

7.3 Multispectral Imaging

Different image priors or learned mappings for spectral reconstruction all make implicit or explicit assumptions concerning the true spectral information. They will fail when these assumptions are violated. More generally, they cannot make the problem of spectral reconstruction well-posed, in contrast to improvements to the spectral resolution of the imaging system.

While we have formulated our approach in the context of conventional RGB cameras, it can be easily extended to multispectral cameras. Our method may prove useful in this area, as multispectral demosaicing algorithms are still under active research [78].

Another area where our algorithm may perform well is in multiple aperture imaging, assuming each aperture has unique colour channels. On this topic, Oh *et al.* [87] demonstrated spectral imaging by combining the images from multiple RGB cameras. The primary challenge with multiple aperture systems is that the images from each aperture or camera must be accurately

registered. While there are algorithms for image registration in multiple aperture settings [47, 52, 83], we could try to incorporate image registration into our image estimation algorithm, instead of performing image registration as a pre-processing step. Doing so would be novel, as we are not aware of image registration methods that account for chromatic aberration, and may potentially give improved results.

Chapter 8

Conclusion

We have shown that modeling lateral chromatic aberration in terms of wavelengths of light, and then estimating a spectral image consistent with our model, accurately corrects chromatic aberration in the corresponding colour image. We formulated image estimation as an optimization problem, and the algorithm we use to solve the optimization problem is guaranteed to converge towards a global optimum. Furthermore, solving an optimization problem instead of designing a direct algorithm for chromatic aberration correction eliminates demosaicing of the input colour-filter array image. In direct algorithms, demosaicing is either explicit, as a preprocessing step, or implicit, as in colour channel warping methods for chromatic aberration correction.

No previous works have relied on spectral information to correct chromatic aberration in colour images, even though chromatic aberration is elegantly modelled in the spectral domain. To illustrate this point, we show simulated patterns of lateral chromatic aberration in Figure 8.1. When modelled as warping between colour channels (top row), lateral chromatic aberration is calibrated with respect to the spectral characteristics of the calibration scene, and does not vary with the spectral characteristics of the scene for which the image is to be corrected. In contrast, when modelled as warping between spectral bands, lateral chromatic aberration appears as a spectrally-varying point spread function (bottom row).

To illustrate the spectral variation of lateral chromatic aberration more clearly, we show the Green channel point spread functions corresponding to

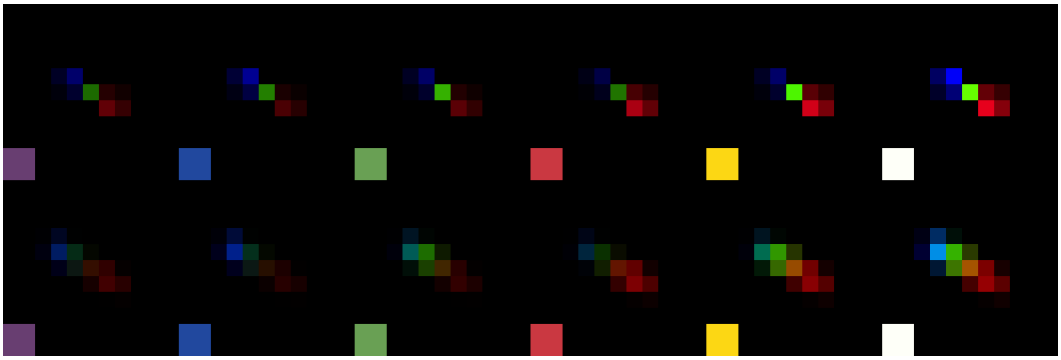


Figure 8.1: A comparison of point spread functions simulated from models of dispersion. The columns contain point spread functions simulated for the spectral radiances of patches 10 (purple), 13 (blue), 14 (green), 15 (red), 16 (yellow), and 19 (white), of our XRite ColorChecker Classic colour chart, lit by a CIE D65 illuminant. For reference, we provide samples of these colours in the large squares to the bottom right of each point spread function. In the top half of the figure, we show point spread functions generated using the colour channel-based model of dispersion from Figure 5.4, which ignores spectral variability. In the bottom half, we show point spread functions generated using our spectral model of dispersion (Figure 5.3). Note that the point spread functions shown in this figure have different overall intensities because they are not normalized to account for the differences in intensity of the different light sources.

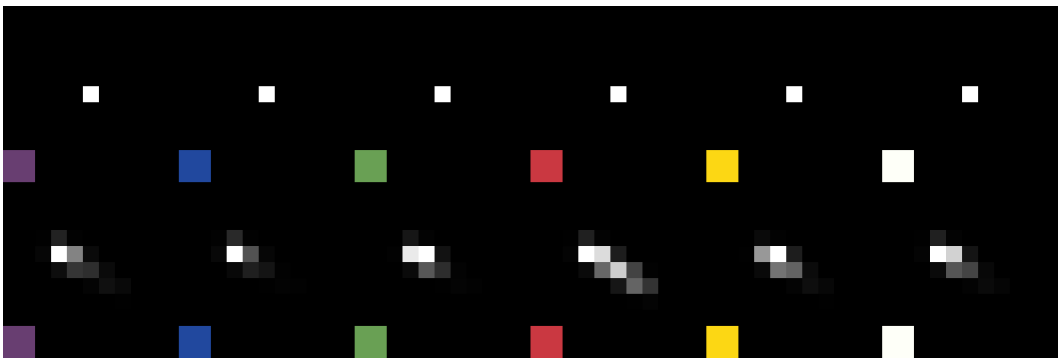


Figure 8.2: A comparison of point spread functions for the Green channel simulated from models of dispersion. The point spread functions shown are the Green channel components of those shown in Figure 8.1. For easier comparison between columns in the figure, the point spread functions have been normalized by their peak intensities.

Figure 8.1 in Figure 8.2. As the spectral radiance of the scene changes, the appropriate point spread function for correcting lateral chromatic abberation also changes. When operating in the colour domain, adequately correcting lateral chromatic aberration is therefore a nonlinear problem. In contrast, in the spectral domain, the point spread function is independent of spectral image intensity, so there is a linear relationship between the input raw image and the spectral image. Therefore, our spectral-domain approach to correcting chromatic aberration has the advantages of both physical validity, and of computational simplicity.

While chromatic aberration should be evaluated visually when designing algorithms for photography applications, visual evaluation is influenced by the colour correction that mapped the images to a standard colour space. To strengthen our conclusions, we therefore validated our correction of chromatic aberration quantitatively, using a novel evaluation technique that measures colour error without the need for demosaicing. Additionally, we validated our estimated spectral images, against a state-of-the-art spectral reconstruction method [19], in simulated experiments, and against inexpensive spectrometry equipment, in real experiments.

The many-to-one mapping from spectral radiances to colour image inten-

sities (*i.e.* metamerism) poses a challenge that warrants future investigation. Still, the slight residual fringes from metamers in our output images are acceptable, in comparison to the fringes or blur left by other methods for chromatic aberration correction. We have not added heuristics or post-processing steps to our algorithm to directly suppress residual fringes. Instead, we hope that our method may serve as a general, interpretable algorithm for others to build upon in further research into demosaicing, spectral reconstruction, and chromatic aberration correction.

References

[1] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 681–695, 2011. DOI: 10.1109/TIP.2010.2076294.

[2] N. Akhtar and A. S. Mian, "Hyperspectral recovery from RGB images using Gaussian processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 15, Oct. 2018. DOI: 10.1109/TPAMI.2018.2873729.

[3] A. Alvarez-Gila, J. V. D. Weijer, and E. Garrote, "Adversarial networks for spatial context-aware spectral image reconstruction from RGB," in *IEEE International Conference on Computer Vision (ICCV) Workshops*, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 480–490. DOI: 10.1109/ICCVW.2017.64.

[4] B. Arad and O. Ben-Shahar, "Sparse recovery of hyperspectral signal from natural RGB images," in *The European Conference on Computer Vision (ECCV)*, Oct. 2016. DOI: 10.1007/978-3-319-46478-7_2.

[5] B. Arad, O. Ben-Shahar, and R. Timofte, "NTIRE 2018 challenge on spectral reconstruction from RGB images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Jun. 2018.

[6] "Standard Practice for Computing the Colors of Objects by Using the CIE System," ASTM International, West Conshohocken, PA, Standard, 2017. DOI: 10.1520/E0308-17.

[7] S.-H. Baek, I. Kim, D. Gutierrez, and M. H. Kim, "Compact single-shot hyperspectral imaging using a prism," *ACM Transactions on Graphics*, vol. 36, no. 6, 2017. DOI: 10.1145/3130800.3130896.

[8] B. Bayer, "Color imaging array," U.S. Patent 3 971 065, Jul. 20, 1976.

[9] M. Belge, M. E. Kilmer, and E. L. Miller, "Efficient determination of multiple regularization parameters in a generalized l-curve framework," *Inverse Problems*, vol. 18, no. 4, pp. 1161–1183, 2002. DOI: 10.1088/0266-5611/18/4/314.

[10] B. Bitlis, P. A. Jansson, and J. P. Allebach, "Parametric point spread function modeling and reduction of stray light effects in digital still cameras," in *SPIE - The International Society for Optical Engineering*, vol. 6498, 2007. DOI: 10.1117/12.715101.

[11] M. Born and E. Wolf, *Principles of optics*, 4. ed. Oxford: Pergamon Press, 1970, ISBN: 9780080139876.

[12] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, Jan. 2011. DOI: 10.1561/2200000016.

[13] J. Brauers and T. Aach, "Longitudinal aberrations caused by optical filters and their compensation in multispectral imaging," in *International Conference on Image Processing (ICIP)*, 2008, pp. 525–528. DOI: 10.1109/ICIP.2008.4711807.

[14] ——, "Geometric calibration of lens and filter distortions for multispectral filter-wheel cameras," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 496–506, 2011. DOI: 10.1109/TIP.2010.2062193.

[15] J. Brauers, N. Schulte, and T. Aach, "Multispectral filter-wheel cameras: Geometric distortion model and compensation algorithms," *IEEE Transactions on Image Processing*, vol. 17, no. 12, pp. 2368–2380, 2008. DOI: 10.1109/TIP.2008.2006605.

[16] X. Cao, H. Du, X. Tong, Q. Dai, and S. Lin, "A prism-mask system for multispectral video acquisition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2423–2435, 2011. DOI: 10.1109/TPAMI.2011.80.

[17] A. Chakrabarti and T. Zickler, "Statistics of real-world hyperspectral images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2011, pp. 193–200. DOI: 10.1109/CVPR.2011.5995660.

[18] J. Chang, H. Kang, and M. G. Kang, "Correction of axial and lateral chromatic aberration with false color filtering," *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 1186–1198, 2013. DOI: 10.1109/TIP.2012.2228489.

[19] I. Choi, D. S. Jeon, G. Nam, D. Gutierrez, and M. H. Kim, "High-quality hyperspectral reconstruction using a spectral prior," *ACM Transactions on Graphics*, vol. 36, no. 6, 2017. DOI: 10.1145/3130800.3130810.

[20] C. Clason, B. Jin, and K. Kunisch, "A semismooth Newton method for L1 data fitting with automatic choice of regularization parameters and noise calibration," *SIAM Journal on Imaging Sciences*, vol. 3, no. 2, pp. 199–231, 2010. DOI: 10.1137/090758003.

[21] O. Cossairt and S. Nayar, "Spectral focal sweep: Extended depth of field from chromatic aberrations," in *IEEE International Conference on Computational Photography (ICCP)*, 2010, pp. 1–8. DOI: 10.1109/ICCPHOT.2010.5585101.

[22] M. M. Darrodi, G. Finlayson, T. Goodman, and M. Mackiewicz, "Reference data set for camera spectral sensitivity estimation," *Journal of the Optical Society of America A: Optics and Image Science, and Vision*, vol. 32, no. 3, pp. 381–391, 2015. DOI: 10.1364/JOSAA.32.000381.

[23] M. Delbracio, P. Musé, A. Almansa, and J.-M. Morel, "The non-parametric sub-pixel local point spread function estimation is a well posed problem," *International Journal of Computer Vision*, vol. 96, no. 2, pp. 175–194, 2012. DOI: 10.1007/s11263-011-0460-0.

[24] K. Dijkstra, J. van de Loosdrecht, L. R. B. Schomaker, and M. A. Wiering, "Hyperspectral demosaicking and crosstalk correction using deep learning," *Machine Vision and Applications*, vol. 30, no. 1, 2019. DOI: 10.1007/s00138-018-0965-4.

[25] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 8, pp. 1558–1570, 2015. DOI: 10.1109/TPAMI.2014.2377715.

[26] D. Eberly. (Jun. 19, 2016). Geometric Tools. version 3.0.0, [Online]. Available: https://www.geometrictools.com (visited on 07/06/2018).

[27] H. Fang, C. Luo, G. Zhou, and X. Wang, "Hyperspectral image deconvolution with a spectral-spatial total variation regularization," *Canadian Journal of Remote Sensing*, vol. 43, no. 4, pp. 384–395, 2017. DOI: 10.1080/07038992.2017.1356221.

[28] G. D. Finlayson, H. Gong, and R. B. Fisher, "Color homography color correction," in *IS and T/SID Color Imaging Conference*, vol. 0, 2016, pp. 310–314. DOI: 10.2352/ISSN.2169-2629.2017.32.310.

[29] G. D. Finlayson, M. MacKiewicz, and A. Hurlbert, "Color correction using root-polynomial regression," *IEEE Transactions on Image Processing*, vol. 24, no. 5, pp. 1460–1470, 2015. DOI: 10.1109/TIP.2015.2405336.

[30] G. Finlayson, M. M. Darrodi, and M. Mackiewicz, "Rank-based camera spectral sensitivity estimation," *Journal of the Optical Society of America A: Optics and Image Science, and Vision*, vol. 33, no. 4, pp. 589–599, 2016. DOI: 10.1364/JOSAA.33.000589.

[31] G. Finlayson, H. Gong, and R. B. Fisher, "Color homography: Theory and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, pp. 20–33, 2019. DOI: 10.1109/TPAMI.2017.2760833.

[32] FLIR. (Jan. 27, 2017). FLIR FLEA3 GigE Vision Imaging Performance Specification. version 1.1, [Online]. Available: https://www.ptgrey.com/support/downloads/10109 (visited on 05/08/2017).

[33] 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: Optics and Image Science, and Vision*, vol. 23, no. 10, pp. 2359–2372, 2006. DOI: 10.1364/JOSAA.23.002359.

[34] N. P. Galatsanos and A. K. Katsaggelos, "Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation," *IEEE Transactions on Image Processing*, vol. 1, no. 3, pp. 322–336, Jul. 1992, ISSN: 1057-7149. DOI: 10.1109/83.148606.

[35] M. E. Gehm, R. John, D. J. Brady, R. M. Willett, and T. J. Schulz, "Single-shot compressive spectral imaging with a dual-disperser architecture," *Optics Express*, vol. 15, no. 21, pp. 14 013–14 027, 2007. DOI: 10.1364/OE.15.014013.

[36] M. Goel, E. Whitmire, A. Mariakakis, T. S. Saponas, N. Joshi, D. Morris, B. Guenter, M. Gavriliu, G. Borriello, and S. N. Patel, "HyperCam: Hyperspectral imaging for ubiquitous computing applications," in *2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, Association for Computing Machinery, Inc, 2015, pp. 145–156. DOI: 10.1145/2750858.2804282.

[37] J. W. Goodman, *Introduction to Fourier optics*, 3rd ed. Englewood, Colorado: Roberts & Company, 2005, ISBN: 0974707724.

[38] P. Hansen, "Analysis of discrete ill-posed problems by means of the l-curve," *SIAM Review*, vol. 34, no. 4, pp. 561–580, 1992. DOI: 10.1137/1034115.

[39] T. Hastie, R. Tibshirani, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. New York: Springer, 2009, p. 745, ISBN: 9780387848570.

[40] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, 2013. DOI: 10.1109/TPAMI.2012.213.

[41] G. E. Healey and R. Kondepudy, "Radiometric CCD camera calibration and noise estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 3, pp. 267–276, 1994. DOI: 10.1109/34.276126.

[42] F. Heide, M. Rouf, M. B. Hullin, B. Labitzke, W. Heidrich, and A. Kolb, "High-quality computational imaging through simple lenses," *ACM Transactions on Graphics*, vol. 32, no. 5, 2013. DOI: 10.1145/2516971.2516974.

[43] F. Heide, M. Steinberger, Y.-T. Tsai, M. Rouf, D. Pajak, D. Reddy, O. Gallo, J. Liu, W. Heidrich, K. Egiazarian, J. Kautz, and K. Pulli, "FlexISP: A flexible camera image processing framework," *ACM Transactions on Graphics*, vol. 33, no. 6, 2014. DOI: 10.1145/2661229.2661260.

[44] V. Heikkinen, "Spectral reflectance estimation using Gaussian processes and combination kernels," *IEEE Transactions on Image Processing*, vol. 27, no. 7, pp. 3358–3373, Jul. 2018. DOI: 10.1109/TIP.2018.2820839.

[45] M. E. Helou, F. Dümbgen, and S. Süsstrunk, "AAM: An assessment metric of axial chromatic aberration," in *IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 2486–2490. DOI: 10.1109/ICIP.2018.8451377.

[46] K. Hirakawa, "Cross-talk explained," in *International Conference on Image Processing (ICIP)*, 2008, pp. 677–680. DOI: 10.1109/ICIP.2008.4711845.

[47] J. Holloway, K. Mitra, S. J. Koppal, and A. N. Veeraraghavan, "Generalized assorted camera arrays: Robust cross-channel registration and applications," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 823–835, Mar. 2015, ISSN: 1057-7149. DOI: 10.1109/TIP.2014.2383315.

[48] B. Horn, *Robot vision*. Cambridge, Massachusetts: MIT Press, 1986, p. 509, ISBN: 0262081598.

[49] Q. Huang, W. Zhu, Y. Zhao, L. Chen, Y. Wang, T. Yue, and X. Cao, "Multispectral image intrinsic decomposition via subspace constraint," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018.

[50] J. Jang, J. D. Yun, and S. Yang, "Modeling non-stationary asymmetric lens blur by normal sinh-arcsinh model," *IEEE Transactions on Image Processing*, vol. 25, no. 5, pp. 2184–2195, 2016. DOI: 10.1109/TIP.2016.2539685.

[51] J. Jemec, F. Pernuš, B. Likar, and M. Bürmen, "2D sub-pixel point spread function measurement using a virtual point-like source," *International Journal of Computer Vision*, vol. 121, no. 3, pp. 391–402, 2017. DOI: 10.1007/s11263-016-0948-8.

[52] D. S. Jeon, S.-H. Baek, I. Choi, and M. H. Kim, "Enhancing the spatial resolution of stereo images using a parallax prior," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018.

[53] Y. Jia, Y. Zheng, L. Gu, A. Subpa-Asa, A. Lam, Y. Sato, and I. Sato, "From RGB to spectrum for natural scenes via manifold-based mapping," in *IEEE International Conference on Computer Vision (ICCV)*, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 4715–4723. DOI: 10.1109/ICCV.2017.504.

[54] J. Jiang, D. Liu, J. Gu, and S. Süsstrunk, "What is the space of spectral sensitivity functions for digital color cameras?" In *Proceedings of IEEE Workshop on Applications of Computer Vision*, 2013, pp. 168–179. DOI: 10.1109/WACV.2013.6475015.

[55] X. Jin, Y. Gu, and T. Liu, "Intrinsic image recovery from remote sensing hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 1, pp. 224–238, Jan. 2019, ISSN: 0196-2892. DOI: 10.1109/TGRS.2018.2853178.

[56] X. Kang, S. Li, L. Fang, and J. A. Benediktsson, "Intrinsic image decomposition for feature extraction of hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 2241–2253, Apr. 2015, ISSN: 0196-2892. DOI: 10.1109/TGRS.2014.2358615.

[57] H. C. Karaimer and M. S. Brown, "Improving color reproduction accuracy on cameras," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018.

[58] E. Kee, S. Paris, S. Chen, and J. Wang, "Modeling and removing spatially-varying optical blur," in *IEEE International Conference on Computational Photography (ICCP)*, Apr. 2011. DOI: 10.1109/ICCPHOT.2011.5753120.

[59] D. Kiku, Y. Monno, M. Tanaka, and M. Okutomi, "Beyond color difference: Residual interpolation for color image demosaicking," English, *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1288–1300, 2016. DOI: 10.1109/TIP.2016.2518082.

[60] J. T. Korneliussen and K. Hirakawa, "Camera processing with chromatic aberration," *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4539–4552, 2014. DOI: 10.1109/TIP.2014.2350911.

[61] D. Krishnan, T. Tay, and R. Fergus, "Blind deconvolution using a normalized sparsity measure," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 233–240. DOI: 10.1109/CVPR.2011.5995521.

[62] F. Kruse, A. Lefkoff, J. Boardman, K. Heidebrecht, A. Shapiro, P. Barloon, and A. Goetz, "The spectral image processing system (SIPS) — interactive visualization and analysis of imaging spectrometer data," *Remote Sensing of Environment*, vol. 44, no. 2, pp. 145–163, 1993, ISSN: 0034-4257. DOI: https://doi.org/10.1016/0034-4257(93)90013-N.

[63] A. B. Lamb and M. Khambete, "Perceived no reference image quality measurement for chromatic aberration," *Journal of Electronic Imaging*, vol. 25, no. 2, 2016. DOI: 10.1117/1.JEI.25.2.023004.

[64] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Jun. 2009, pp. 1964–1971. DOI: 10.1109/CVPRW.2009.5206815.

[65] J. Li, C. Bai, Z. Lin, and J. Yu, "Optimized color filter arrays for sparse representation-based demosaicking," *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2381–2393, 2017. DOI: 10.1109/TIP.2017.2679440.

[66] X. Lin, Y. Liu, J. Wu, and Q. Dai, "Spatial-spectral encoded compressive hyperspectral imaging," *ACM Transactions on Graphics*, vol. 33, no. 6, 2014. DOI: 10.1145/2661229.2661262.

[67] B. J. Lindbloom. (Apr. 7, 2017). Chromatic adaptation, [Online]. Available: http://www.brucelindbloom.com/index.html?Eqn_ChromAdapt.html (visited on 05/29/2019).

[68] ——, (Apr. 7, 2017). Spectral power distribution of a cie d-illuminant, [Online]. Available: http://www.brucelindbloom.com/Eqn_DIlluminant.html (visited on 05/29/2019).

[69] M. Lindstrand, "Sensor interpixel correlation analysis and reduction for color filter array high dynamic range image reconstruction," *Color Research and Application*, vol. 44, no. 3, pp. 335–347, 2019. DOI: 10.1002/col.22343.

[70] A. Lluis-Gomez and E. A. Edirisinghe, "Chromatic aberration correction in raw domain for image quality enhancement in image sensor processors," in *IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, Aug. 2012, pp. 241–244. DOI: 10.1109/ICCP.2012.6356192.

[71] J. Mallon and P. F. Whelan, "Calibration and removal of lateral chromatic aberration in images," *Pattern Recognition Letters*, vol. 28, no. 1, pp. 125–135, 2007. DOI: 10.1016/j.patrec.2006.06.013.

[72] A. Manakov, "Evaluation of computational radiometric and spectral sensor calibration techniques," in *SPIE - The International Society for Optical Engineering*, F. Truchetet, P. Schelkens, T. Ebrahimi, G. Cristobal, and P. Saarikko, Eds., vol. 9896, SPIE, 2016. DOI: 10.1117/12.2228071.

[73] F. Mannan and M. S. Langer, "Blur calibration for depth from defocus," in *Canadian Conference on Computer and Robot Vision (CRV)*, J. Guerrero, Ed., Institute of Electrical and Electronics Engineers Inc., Jun. 2016, pp. 281–288. DOI: 10.1109/CRV.2016.62.

[74] MathWorks. (Feb. 28, 2019). imregtform — estimate geometric transformation that aligns two 2-d or 3-d images. version 2018b, [Online]. Available: https://www.mathworks.com/help/images/ref/imregtform.html (visited on 04/17/2019).

[75] R. Matsuoka, K. Asonuma, G. Takahashi, T. Danjo, and K. Hirana, "Evaluation of correction methods of chromatic aberration in digital camera images," *ISPRS Ann.Photogramm.Remote Sens.Spatial Inf.Sci.*, vol. I-3, pp. 49–55, 2012. DOI: 10.5194/isprsannals-I-3-49-2012.

[76] D. Mattes, D. R. Haynor, H. Vesselle, T. Lewellen, and W. Eubank, "Non-rigid multimodality image registration," in *Medical Imaging 2001: Image Processing*, SPIE Publications, Jul. 2001, pp. 1609–1620.

[77] D. Menon and G. Calvagno, "Regularization approaches to demosaicking," *IEEE Transactions on Image Processing*, vol. 18, no. 10, pp. 2209–2220, 2009. DOI: 10.1109/TIP.2009.2025092.

[78] S. Mihoubi, B. Mathon, J.-B. Thomas, O. Losson, and L. Macaire, "Illumination-robust multispectral demosaicing," in *7th International Conference on Image Processing Theory, Tools and Applications (IPTA)*, Institute of Electrical and Electronics Engineers Inc., Jan. 2018, pp. 1–6. DOI: 10.1109/IPTA.2017.8310135.

[79] Y. Monno, D. Kiku, M. Tanaka, and M. Okutomi, "Adaptive residual interpolation for color and multispectral image demosaicking," *Sensors (Switzerland)*, vol. 17, no. 12, 2017. DOI: 10.3390/s17122787.

[80] Y. Monno, S. Kikuchi, M. Tanaka, and M. Okutomi, "A practical one-shot multispectral imaging system using a single image sensor," *IEEE Transactions on Image Processing*, vol. 24, no. 10, pp. 3048–3059, Oct. 2015. DOI: 10.1109/TIP.2015.2436342.

[81] A. Mosleh, Y. Elmi Sola, F. Zargari, E. Onzon, and J. M. P. Langlois, "Explicit ringing removal in image deblurring," *IEEE Transactions on Image Processing*, vol. 27, no. 2, pp. 580–593, Feb. 2018, ISSN: 1057-7149. DOI: 10.1109/TIP.2017.2764625.

[82] A. Mosleh, P. Green, E. Onzon, I. Begin, and J. M. P. Langlois, "Camera intrinsic blur kernel estimation: A reliable framework," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE Computer Society, Jun. 2015, pp. 4961–4968. DOI: 10.1109/CVPR.2015.7299130.

[83] J. Mustaniemi, J. Kannala, and J. Heikkilä, "Parallax correction via disparity estimation in a multi-aperture camera," *Machine Vision and Applications*, vol. 27, no. 8, pp. 1313–1323, 2016. DOI: 10.1007/s00138-016-0773-7.

[84] R. M. H. Nguyen, D. K. Prasad, and M. S. Brown, "Training-based spectral reconstruction from a single RGB image," in *The European Conference on Computer Vision (ECCV)*, Sep. 2014. DOI: 10.1007/978-3-319-10584-0_13.

[85] S. Nie, L. Gu, Y. Zheng, A. Lam, N. Ono, and I. Sato, "Deeply learned filter response functions for hyperspectral reconstruction," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018.

[86] I. Nimeroff and J. A. Yurow, "Degree of metamerism," *Journal of the Optical Society of America*, vol. 55, no. 2, pp. 185–190, 1965. DOI: 10.1364/JOSA.55.000185.

[87] S. W. Oh, M. S. Brown, M. Pollefeys, and S. J. Kim, "Do it yourself hyperspectral imaging with everyday digital cameras," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE Computer Society, Jan. 2016, pp. 2461–2469.

[88] N. Otsu, "Threshold selection method from gray-level histograms.," *IEEE Trans Syst Man Cybern*, vol. SMC-9, no. 1, pp. 62–66, 1979.

[89] J. Pan, Z. Hu, Z. Su, and M. Yang, "L0-regularized intensity and gradient prior for deblurring text images and beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 342–355, Feb. 2017, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2016.2551244.

[90] J.-I. Park, M.-H. Lee, M. D. Grossberg, and S. K. Nayar, "Multispectral imaging using multiplexed illumination," in *IEEE International Conference on Computer Vision*, 2007. DOI: 10.1109/ICCV.2007.4409090.

[91] M. Parmar, S. Lansel, and B. A. Wandell, "Spatio-spectral reconstruction of the multispectral datacube using sparse recovery," in *International Conference on Image Processing (ICIP)*, 2008. DOI: 10.1109/ICIP.2008.4711794.

[92] M. Pharr and G. Humphreys, *Physically Based Rendering: From Theory to Implementation*, ser. Morgan Kaufmann series in interactive 3D technology. Burlington, Massachusetts: Elsevier Science, 2010, ISBN: 9780123750792.

[93] L. Pi, W. Wang, and M. Ng, "A spatially variant total variational model for chromatic aberration correction," *Journal of Visual Communication and Image Representation*, vol. 41, pp. 296–304, 2016. DOI: 10.1016/j.jvcir.2016.10.009.

[94] D. K. Prasad, R. Nguyen, and M. S. Brown, "Quick approximation of camera's spectral response from casual lighting," in *Proceedings of the IEEE International Conference on Computer Vision*, Institute of Electrical and Electronics Engineers Inc., 2013, pp. 844–851. DOI: 10.1109/ICCVW.2013.116.

[95] D. K. Prasad and L. Wenhe, "Metrics and statistics of frequency of occurrence of metamerism in consumer cameras for natural scenes," *Journal of the Optical Society of America A: Optics and Image Science, and Vision*, vol. 32, no. 7, pp. 1390–1402, 2015. DOI: 10.1364/JOSAA.32.001390.

[96] J. Qiu and H. Xu, "Camera response prediction for various capture settings using the spectral sensitivity and crosstalk model," *Applied Optics*, vol. 55, no. 25, pp. 6989–6999, 2016. DOI: 10.1364/AO.55.006989.

[97] S. Rahunathan, D. Stredney, P. Schmalbrock, and B. D. Clymer. (Jan. 26, 2005). Image registration using rigid registration and maximization of mutual information. Poster presented at: MMVR13. The 13th Annual Medicine Meets Virtual Reality Conference, Long Beach, CA, [Online]. Available: https://www.osc.edu/files/research/Biomed/vtbone/updates/image_registration.pdf (visited on 04/17/2019).

[98] E. Reinhard, G. Ward, S. Pattanaik, and P. Debevec, *High dynamic range imaging*. San Francisco, Calif.: Morgan Kaufmann, 2006, ISBN: 9780125852630.

[99] V. Rudakova and P. Monasse, "Precise correction of lateral chromatic aberration in images," in *Image and Video Technology*, 2014, pp. 12–22.

[100] L. I. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, ID: 271587, 1992. DOI: //doi.org/10.1016/0167-2789(92)90242-F.

[101] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Scholkopf, "Non-stationary correction of optical aberrations," in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 659–666. DOI: 10.1109/ICCV.2011.6126301.

[102] T. Seim and A. Valberg, "A neurophysiologically-based analysis of lightness and brightness perception," *Color Research and Application*, vol. 41, no. 4, pp. 339–351, Aug. 2016, ISSN: 1520-6378. DOI: 10.1002/col.21963.

[103] L. Shen, C. Yeo, and B.-S. Hua, "Intrinsic image decomposition using a sparse representation of reflectance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2904–2915, 2013. DOI: 10.1109/TPAMI.2013.136.

[104] Z. Shi, C. Chen, Z. Xiong, D. Liu, and F. Wu, "HSCNN+: Advanced CNN-based hyperspectral recovery from RGB images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Jun. 2018.

[105] Y. Shih, B. Guenter, and N. Joshi, "Image enhancement using calibrated lens simulations," ser. 12th European Conference on Computer Vision, ECCV 2012, vol. 7575 LNCS, 2012, ch. PART 4, p. 42. DOI: 10.1007/978-3-642-33765-9_4.

[106] E. P. Simoncelli, "Bayesian denoising of visual images in the wavelet domain," in *Bayesian Inference in Wavelet-Based Models*, P. Müller and B. Vidakovic, Eds. New York, NY: Springer New York, 1999, pp. 291–308. DOI: 10.1007/978-1-4612-0567-8_18.

[107] B. K. Sing, "Automatic removal of chromatic aberration from a single image," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2007. DOI: 10.1109/CVPR.2007.383214.

[108] Y. Song, D. Brie, E.-H. Djermoune, and S. Henrot, "Regularization parameter estimation for non-negative hyperspectral image deconvolution," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5316–5330, 2016. DOI: 10.1109/TIP.2016.2601489.

[109] A. L. Stubbs and C. W. Stubbs, "Spectral discrimination in color blind animals via chromatic aberration and pupil shape," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 113, no. 29, pp. 8206–8211, 2016. DOI: 10.1073/pnas.1524578113.

[110] M. Styner, C. Brechbuehler, G. Székely, and G. Gerig, "Parametric estimate of intensity inhomogeneities applied to MRI," *IEEE Transactions on Medical Imaging*, vol. 19, no. 3, pp. 153–165, 2000. DOI: 10.1109/42.845174.

[111] T. Sun, Y. Peng, and W. Heidrich, "Revisiting cross-channel information transfer for chromatic aberration correction," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 3268–3276. DOI: 10.1109/ICCV.2017.352. [Online]. Available: http://vccimaging.org/Publications/Sun2017RCI/.

[112] R. Szeliski, *Computer vision: algorithms and applications*. London: Springer, 2011, p. 824, ISBN: 9781848829350.

[113] ——, "Image processing," in *Computer Vision: Algorithms and Applications*. London: Springer London, 2011, pp. 87–180, ISBN: 9781848829350. DOI: 10.1007/978-1-84882-935-0_3.

[114] D. S. Tan, W.-Y. Chen, and K.-L. Hua, "DeepDemosaicking: Adaptive image demosaicking via multiple deep fully convolutional networks," *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2408–2419, 2018. DOI: 10.1109/TIP.2018.2803341.

[115] H. Tan, X. Zeng, S. Lai, Y. Liu, and M. Zhang, "Joint demosaicing and denoising of noisy bayer images with ADMM," in *International Conference on Image Processing (ICIP)*, Sep. 2018, pp. 2951–2955. DOI: 10.1109/ICIP.2017.8296823.

[116] R. Timofte, V. D. Smet, and L. V. Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Asian Conference on Computer Vision (ACCV)*, 2014, pp. 111–126. DOI: 10.1007/978-3-319-16817-3_8.

[117] P. Trouvé-Peloux, J. Sabater, A. Bernard-Brunel, F. Champagnat, G. L. Besnerais, and T. Avignon, "Turning a conventional camera into a 3D camera with an add-on," *Applied Optics*, vol. 57, no. 10, pp. 2553–2563, 2018. DOI: 10.1364/AO.57.002553.

[118] P. Trouvé, F. Champagnat, G. L. Besnerais, G. Druart, and J. Idier, "Design of a chromatic 3D camera with an end-to-end performance model approach," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Jun. 2013, pp. 953–960. DOI: 10.1109/CVPRW.2013.140.

[119] A. Wagadarikar, R. John, R. Willett, and D. Brady, "Single disperser design for coded aperture snapshot spectral imaging," *Applied Optics*, vol. 47, no. 10, B44–B51, 2008. DOI: 10.1364/AO.47.000B44.

[120] G. Wahba, *Spline models for observational data*. Philadelphia, Pa.: Society for Industrial and Applied Mathematics SIAM, 1990, ISBN: 9781611970128.

[121] L. Wang, T. Zhang, Y. Fu, and H. Huang, "HyperReconNet: Joint coded aperture optimization and image reconstruction for compressive hyperspectral imaging," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2257–2270, 2019. DOI: 10.1109/TIP.2018.2884076.

[122] X. Wang, M. Pedersen, and J.-B. Thomas, "The influence of chromatic aberration on demosaicking," in *European Workshop on Visual Information Processing (EUVIP)*, Institute of Electrical and Electronics Engineers Inc., Dec. 2014. DOI: 10.1109/EUVIP.2014.7018410.

[123] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004. DOI: 10.1109/TIP.2003.819861.

[124] J. Wu, J. Aeschbacher, and R. Timofte, "In defense of shallow learned spectral reconstruction from RGB images," in *IEEE International Conference on Computer Vision (ICCV) Workshops*, Institute of Electrical and Electronics Engineers, Jan. 2018, pp. 471–479. DOI: 10.1109/ICCVW.2017.63.

[125] J. Wu, R. Timofte, and L. V. Gool, "Demosaicing based on directional difference regression and efficient regression priors," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3862–3874, 2016. DOI: 10.1109/TIP.2016.2574984.

[126] X. Wu and X. Zhang, "Joint color decrosstalk and demosaicking for CFA cameras," *IEEE Transactions on Image Processing*, vol. 19, no. 12, pp. 3181–3189, 2010. DOI: 10.1109/TIP.2010.2052001.

[127] S.-n. Yang, Y.-c. Tai, H. Laukkanen, and J. E. Sheedy, "Effects of ocular transverse chromatic aberration on peripheral word identification," *Vision Research*, vol. 51, no. 21, pp. 2273–2281, Sep. 2011. DOI: 10.1016/j.visres.2011.08.022.

[128] F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar, "Generalized assorted pixel camera: Postcapture control of resolution, dynamic range, and spectrum," *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2241–2253, 2010. DOI: 10.1109/TIP.2010.2046811.

Appendix A

Additional Results on the KAIST Dataset

In Section 5.2, we presented spectral and colour error metrics for variants of our algorithm on four images from the KAIST spectral image dataset. In this appendix, we present the same error metrics for only the first image from the dataset (Figure 4.14a), but we include results for variants of our algorithm that incorporate a penalty on the image Laplacian. For brevity, these results were not included in Section 5.2. As mentioned in Section 5.2, we only tested the penalty on the image Laplacian on the first image from the KAIST dataset because the penalty greatly increased the running time of our optimization algorithm.

In Table A.1, we provide spectral image evaluation results for the first image from the KAIST dataset (analogous to Table 5.1).

Table A.1: Spectral error metrics evaluated on the first image from the KAIST dataset (Figure 4.14a). Rows are sorted by MRAE. The first column identifies the combination of penalty terms used in our algorithms, as listed in Table 3.1. The second column indicates whether or not we imposed a non-negativity constraint on the estimated image. The third column identifies the criterion used to select regularization weights (Section 3.6). Algorithms with empty rows failed to converge within 1000 ADMM iterations, with 100 conjugate gradient iterations per ADMM iteration.

			$MRAE$	$RMSE_{\text{pixel}}$	$RMSE_{\text{global}}$	GOF	$SSIM_{\text{mean}}$
Penalties	NNeg	Criterion					
L2	yes	MSEC	**0.249**	1.89×10^{-4}	2.59×10^{-4}	0.817	0.99941
L1L1	yes	MSEC	0.266	1.81×10^{-4}	2.37×10^{-4}	0.831	0.99951
L1	yes	DMC	0.269	2.04×10^{-4}	2.98×10^{-4}	0.821	0.99923

Table A.1: (continued)

Penalties	NNeg	Criterion	$MRAE$	$RMSE_{\text{pixel}}$	$RMSE_{\text{global}}$	GOF	$SSIM_{\text{mean}}$
L2L2	yes	MSEC	0.270	1.82×10^{-4}	2.34×10^{-4}	0.817	0.99951
L1	yes	MSEC	0.279	$\mathbf{1.77}\times10^{-4}$	2.30×10^{-4}	**0.835**	**0.99953**
L2	no	MSEC	0.279	1.84×10^{-4}	2.33×10^{-4}	0.814	0.99950
L1	no	DMC	0.285	1.82×10^{-4}	$\mathbf{2.29}\times10^{-4}$	0.823	0.99951
L2L2	no	MSEC	0.291	1.84×10^{-4}	2.30×10^{-4}	0.812	0.99951
L1	no	MSEC	0.309	1.86×10^{-4}	2.40×10^{-4}	0.821	0.99948
Choi *et al.* [19]			0.341	2.02×10^{-4}	2.89×10^{-4}	0.800	0.99927
L2	yes	DMC	0.344	2.12×10^{-4}	2.93×10^{-4}	0.805	0.99923
L2L2	yes	DMC	0.344	2.12×10^{-4}	2.93×10^{-4}	0.805	0.99923
L1L1	yes	MDC	0.353	1.93×10^{-4}	2.41×10^{-4}	0.812	0.99947
L1L1	yes	DMC	0.362	2.32×10^{-4}	3.43×10^{-4}	0.799	0.99894
L1L1	no	MSEC	0.364	2.03×10^{-4}	2.70×10^{-4}	0.781	0.99936
L1	yes	MDC	0.365	2.69×10^{-4}	4.73×10^{-4}	0.785	0.99820
L2L2	yes	MDC	0.383	2.06×10^{-4}	2.61×10^{-4}	0.808	0.99939
L2	yes	MDC	0.404	2.33×10^{-4}	3.23×10^{-4}	0.803	0.99908
L1L1	no	MDC	0.433	2.02×10^{-4}	2.45×10^{-4}	0.791	0.99944
Lap1	yes	MSEC	0.440	2.68×10^{-4}	3.86×10^{-4}	0.644	0.99865
Lap1	yes	DMC	0.440	2.68×10^{-4}	3.86×10^{-4}	0.643	0.99865
Lap2	no	MSEC	0.441	2.69×10^{-4}	3.86×10^{-4}	0.640	0.99864
Lap2	yes	MDC	0.441	2.68×10^{-4}	3.85×10^{-4}	0.641	0.99866
Lap2	yes	DMC	0.441	2.68×10^{-4}	3.85×10^{-4}	0.641	0.99866
L1Spectral-Lap2	yes	DMC	0.442	2.68×10^{-4}	3.85×10^{-4}	0.641	0.99866
Lap2	yes	MSEC	0.442	2.68×10^{-4}	3.85×10^{-4}	0.641	0.99865
L1Spatial-Lap2	no	MSEC	0.444	2.69×10^{-4}	3.86×10^{-4}	0.641	0.99864
Lap2	no	DMC	0.444	2.69×10^{-4}	3.86×10^{-4}	0.641	0.99864
Lap1	no	DMC	0.444	2.69×10^{-4}	3.86×10^{-4}	0.641	0.99864
L1Spectral-Lap2	no	DMC	0.444	2.69×10^{-4}	3.86×10^{-4}	0.641	0.99864
Lap2	no	MDC	0.444	2.69×10^{-4}	3.86×10^{-4}	0.641	0.99864
Lap1	no	MSEC	0.444	2.69×10^{-4}	3.86×10^{-4}	0.641	0.99864
L1Spatial-Lap2	yes	DMC	0.445	2.73×10^{-4}	4.14×10^{-4}	0.670	0.99845
L2L2	no	MDC	0.446	2.11×10^{-4}	2.60×10^{-4}	0.796	0.99938
L1Spatial-Lap2	no	MDC	0.447	2.70×10^{-4}	3.87×10^{-4}	0.640	0.99864
L2L2	no	DMC	0.500	2.22×10^{-4}	2.83×10^{-4}	0.790	0.99927
L2	no	DMC	0.500	2.22×10^{-4}	2.83×10^{-4}	0.790	0.99927
L1Spatial-Lap2	no	DMC	0.511	2.75×10^{-4}	3.96×10^{-4}	0.635	0.99858

Table A.1: (continued)

Penalties	NNeg	Criterion	$MRAE$	$RMSE_{\text{pixel}}$	$RMSE_{\text{global}}$	GOF	$SSIM_{\text{mean}}$
L2	no	MDC	0.535	2.38×10^{-4}	3.08×10^{-4}	0.793	0.99915
L1Spectral-Lap2	no	MDC	0.611	2.75×10^{-4}	3.48×10^{-4}	0.554	0.99887
L1L1	no	DMC	0.680	1.66×10^{-4}	3.72×10^{-4}	0.778	0.99874
L1	no	MDC	0.715	2.85×10^{-4}	4.45×10^{-4}	0.780	0.99840
L1Spectral-Lap2	no	MSEC	6.997	1.76×10^{-3}	6.05×10^{-3}	0.535	0.96451
L1Spatial-Lap2	yes	MDC	—	—	—	—	—
L1Spectral-Lap2	yes	MSEC	—	—	—	—	—
L1Spatial-Lap2	yes	MSEC	—	—	—	—	—
Lap1	yes	MDC	—	—	—	—	—
L1Spectral-Lap2	yes	MDC	—	—	—	—	—
Lap1	no	MDC	—	—	—	—	—

Similarly, in Table A.2, we provide colour image reconstruction error metrics, in the same format as Table 5.2.

Table A.2: RGB error metrics evaluated on the first image from the KAIST dataset (Figure 4.14a). Rows are sorted by MRAE in the Green channel ($MRAE_G$). The first column distinguishes spectral image estimation algorithms from colour image estimation algorithms. The second column identifies the combination of penalty terms used in our algorithms, as listed in Table 3.1. The third column indicates whether or not we imposed a non-negativity constraint on the estimated image. The fourth column identifies the criterion used to select regularization weights (Section 3.6). Algorithms with empty rows failed to converge within 1000 ADMM iterations with 100 conjugate gradient iterations per ADMM iteration.

				$MRAE$ Red	$MRAE$ Green	$MRAE$ Blue	$RMSE$ Red	$RMSE$ Green	$RMSE$ Blue	$SSIM$ Red	$SSIM$ Green	$SSIM$ Blue
Spectral	Penalties	NNeg	Criterion									
yes	Lap1	no	MSEC	0.00855	**0.00354**	0.00785	0.00031	0.00021	0.00040	0.99994	0.99997	0.99991
no	Lap2	no	MSEC	0.00873	0.00354	0.00829	0.00033	**0.00020**	0.00042	0.99994	**0.99998**	0.99989
no	Lap1	no	MSEC	0.00880	0.00370	0.00834	0.00034	0.00021	0.00043	0.99994	0.99998	0.99989
yes	L1Spatial-Lap2	no	MSEC	**0.00844**	0.00371	**0.00785**	0.00031	0.00021	0.00039	**0.99994**	0.99997	0.99991
yes	Lap2	yes	MSEC	0.00846	0.00372	0.00786	**0.00031**	0.00021	**0.00039**	0.99994	0.99997	**0.99991**
yes	Lap1	no	DMC	0.00991	0.00396	0.00866	0.00039	0.00027	0.00046	0.99991	0.99996	0.99987
no	L1Spectral-Lap2	yes	MSEC	0.01114	0.00407	0.00822	0.00069	0.00026	0.00088	0.99983	0.99996	0.99972
no	L1Spectral-Lap2	no	MSEC	0.00884	0.00415	0.00833	0.00036	0.00024	0.00045	0.99993	0.99997	0.99989
no	L1Spatial-Lap2	yes	MSEC	0.00983	0.00425	0.00928	0.00042	0.00023	0.00052	0.99991	0.99997	0.99986
no	ARI [79]			0.01043	0.00437	0.00874	0.00037	0.00025	0.00046	0.99991	0.99996	0.99986
yes	Lap1	yes	DMC	0.01116	0.00438	0.00939	0.00084	0.00037	0.00114	0.99971	0.99994	0.99954
yes	Lap1	yes	MSEC	0.01123	0.00446	0.00941	0.00059	0.00036	0.00070	0.99983	0.99994	0.99979
no	Lap2	yes	MSEC	0.00953	0.00447	0.00892	0.00039	0.00024	0.00050	0.99992	0.99997	0.99987
no	L2L2	no	MSEC	0.01246	0.00449	0.00988	0.00064	0.00041	0.00075	0.99977	0.99993	0.99975

Table A.2: (continued)

				$MRAE$ Red	$MRAE$ Green	$MRAE$ Blue	$RMSE$ Red	$RMSE$ Green	$RMSE$ Blue	$SSIM$ Red	$SSIM$ Green	$SSIM$ Blue
Spectral	Penalties	NNeg	Criterion									
yes	Lap2	no	DMC	0.00955	0.00460	0.00853	0.00036	0.00024	0.00042	0.99993	0.99997	0.99990
no	Bilinear interpolation			0.01042	0.00464	0.00941	0.00055	0.00044	0.00067	0.99924	0.99996	0.99946
no	L1Spatial-Lap2	no	MSEC	0.00991	0.00465	0.00914	0.00040	0.00025	0.00050	0.99992	0.99997	0.99987
no	L2L2	no	DMC	0.01397	0.00466	0.01181	0.00085	0.00044	0.00103	0.99967	0.99993	0.99959
no	L2	no	DMC	0.01343	0.00467	0.01143	0.00074	0.00044	0.00089	0.99973	0.99992	0.99965
no	L1Spatial-Lap2	no	DMC	0.01394	0.00469	0.01175	0.00084	0.00043	0.00101	0.99968	0.99993	0.99960
no	L2	no	MSEC	0.01425	0.00475	0.01195	0.00087	0.00045	0.00104	0.99966	0.99992	0.99958
no	L1	no	DMC	0.01412	0.00480	0.01187	0.00085	0.00044	0.00102	0.99967	0.99993	0.99959
no	L2	yes	MSEC	0.02965	0.00493	0.02858	0.00093	0.00046	0.00117	0.99951	0.99992	0.99915
no	L2L2	yes	DMC	0.04816	0.00520	0.04753	0.00099	0.00044	0.00138	0.99921	0.99992	0.99795
no	Lap2	yes	DMC	0.01032	0.00535	0.00951	0.00040	0.00029	0.00050	0.99992	0.99996	0.99986
no	L1Spectral-Lap2	yes	DMC	0.01025	0.00535	0.00945	0.00040	0.00029	0.00050	0.99992	0.99996	0.99986
no	Lap2	no	DMC	0.01026	0.00535	0.00945	0.00040	0.00029	0.00049	0.99992	0.99996	0.99987
no	L2	yes	DMC	0.04747	0.00540	0.04666	0.00102	0.00045	0.00139	0.99921	0.99992	0.99798
no	L1Spectral-Lap2	no	DMC	0.00996	0.00542	0.00929	0.00038	0.00030	0.00048	0.99992	0.99996	0.99987
no	L1Spatial-Lap2	yes	DMC	0.01069	0.00546	0.00958	0.00041	0.00029	0.00051	0.99991	0.99996	0.99986
yes	L1Spectral-Lap2	yes	DMC	0.01107	0.00572	0.00945	0.00044	0.00032	0.00048	0.99990	0.99995	0.99987
yes	Lap2	yes	DMC	0.01108	0.00573	0.00946	0.00044	0.00032	0.00048	0.99990	0.99995	0.99987

Table A.2: (continued)

Spectral	Penalties	NNeg	Criterion	$MRAE$ Red	$MRAE$ Green	$MRAE$ Blue	$RMSE$ Red	$RMSE$ Green	$RMSE$ Blue	$SSIM$ Red	$SSIM$ Green	$SSIM$ Blue
yes	L1Spectral-Lap2	no	DMC	0.01108	0.00573	0.00946	0.00044	0.00032	0.00048	0.99990	0.99995	0.99987
yes	Lap2	no	MDC	0.01179	0.00623	0.00989	0.00048	0.00036	0.00051	0.99988	0.99994	0.99986
no	L2L2	yes	MSEC	0.01428	0.00623	0.01600	0.00069	0.00046	0.00081	0.99973	0.99991	0.99962
yes	L1	yes	DMC	0.01486	0.00641	0.01060	0.00098	0.00054	0.00122	0.99964	0.99990	0.99958
yes	L1L1	yes	DMC	0.01685	0.00651	0.01166	0.00114	0.00050	0.00125	0.99948	0.99987	0.99944
no	Lap2	no	MDC	0.01159	0.00665	0.01043	0.00046	0.00039	0.00057	0.99989	0.99993	0.99983
no	L1L1	no	MSEC	0.01523	0.00666	0.01153	0.00058	0.00036	0.00070	0.99981	0.99994	0.99977
no	Lap2	yes	MDC	0.01159	0.00666	0.01044	0.00047	0.00039	0.00058	0.99989	0.99993	0.99983
no	L1	no	MSEC	0.01802	0.00680	0.01337	0.00078	0.00037	0.00096	0.99970	0.99994	0.99962
yes	Lap2	no	MSEC	0.02310	0.00684	0.01923	0.00110	0.00079	0.00197	0.99931	0.99943	0.99841
yes	Lap2	yes	MDC	0.01277	0.00691	0.01048	0.00054	0.00041	0.00056	0.99985	0.99993	0.99983
no	Lap1	yes	MSEC	0.01666	0.00726	0.01313	0.00341	0.00078	0.00477	0.99736	0.99971	0.99630
no	Lap1	yes	DMC	0.01692	0.00772	0.01316	0.00321	0.00077	0.00445	0.99770	0.99972	0.99692
yes	L1Spatial-Lap2	no	DMC	0.01650	0.00838	0.01159	0.00041	0.00029	0.00047	0.99989	0.99995	0.99986
no	Lap1	no	DMC	0.01663	0.00857	0.01233	0.00043	0.00026	0.00053	0.99988	0.99995	0.99982
yes	L2L2	no	DMC	0.02173	0.00866	0.01499	0.00134	0.00074	0.00120	0.99923	0.99982	0.99947
yes	L2	no	DMC	0.02173	0.00866	0.01499	0.00134	0.00074	0.00120	0.99923	0.99982	0.99947
yes	L2	yes	DMC	0.02191	0.00867	0.01500	0.00134	0.00074	0.00120	0.99923	0.99982	0.99947
yes	L2L2	yes	DMC	0.02191	0.00867	0.01500	0.00134	0.00074	0.00120	0.99923	0.99982	0.99947
yes	L1Spatial-Lap2	yes	DMC	0.01753	0.00914	0.01213	0.00042	0.00030	0.00048	0.99988	0.99995	0.99986
no	L1L1	yes	DMC	0.02326	0.00920	0.01726	0.00176	0.00073	0.00212	0.99896	0.99982	0.99876

Table A.2: (continued)

Spectral	Penalties	NNeg	Criterion	*MRAE* Red	*MRAE* Green	*MRAE* Blue	*RMSE* Red	*RMSE* Green	*RMSE* Blue	*SSIM* Red	*SSIM* Green	*SSIM* Blue
yes	L2L2	no	MSEC	0.02010	0.00940	0.01610	0.00106	0.00102	0.00152	0.99927	0.99952	0.99899
no	L1L1	yes	MSEC	0.01829	0.00953	0.01391	0.00107	0.00061	0.00128	0.99951	0.99985	0.99945
yes	L2L2	yes	MSEC	0.02009	0.00978	0.01555	0.00100	0.00092	0.00135	0.99939	0.99955	0.99922
yes	L1L1	no	MSEC	0.02160	0.00991	0.01455	0.00146	0.00134	0.00178	0.99905	0.99931	0.99896
no	L1L1	no	DMC	0.02196	0.00995	0.01536	0.00063	0.00034	0.00077	0.99974	0.99993	0.99967
yes	L1	no	MSEC	0.02160	0.01035	0.01890	0.00125	0.00354	0.00538	0.99922	0.99483	0.99449
yes	L2	yes	MSEC	0.02055	0.01139	0.01667	0.00124	0.00111	0.00169	0.99920	0.99931	0.99886
yes	L2	no	MSEC	0.01814	0.01162	0.01434	0.00096	0.00107	0.00117	0.99950	0.99935	0.99935
no	L1	yes	MSEC	0.02841	0.01386	0.02007	0.00172	0.00100	0.00200	0.99897	0.99973	0.99886
yes	L1L1	yes	MSEC	0.03237	0.01550	0.03430	0.00201	0.00492	0.00774	0.99818	0.99055	0.98970
yes	L1Spectral-Lap2	no	MDC	0.03408	0.01689	0.02194	0.00100	0.00101	0.00136	0.99908	0.99923	0.99885
no	L2	yes	MDC	0.03492	0.01896	0.02628	0.00219	0.00186	0.00255	0.99812	0.99909	0.99814
no	L2	no	MDC	0.03498	0.01896	0.02630	0.00218	0.00186	0.00255	0.99812	0.99909	0.99814
no	L2L2	yes	MDC	0.03492	0.01899	0.02629	0.00219	0.00187	0.00255	0.99812	0.99908	0.99814
no	L2L2	no	MDC	0.03504	0.01900	0.02634	0.00219	0.00187	0.00255	0.99811	0.99908	0.99813
no	L1	yes	DMC	0.03692	0.01980	0.02468	0.00177	0.00098	0.00204	0.99876	0.99964	0.99869
yes	L1L1	no	DMC	0.03956	0.02079	0.02497	0.00102	0.00056	0.00114	0.99931	0.99965	0.99919
yes	L1Spatial-Lap2	no	MDC	0.03347	0.02122	0.02257	0.00201	0.00192	0.00193	0.99821	0.99880	0.99863
yes	L1	no	DMC	0.04787	0.02163	0.02191	0.00105	0.00080	0.00145	0.99921	0.99962	0.99930
yes	L1	yes	MSEC	0.04722	0.02349	0.04297	0.00209	0.00468	0.00692	0.99796	0.99124	0.99072
yes	L1Spectral-Lap2	no	MSEC	0.10908	0.03523	0.05193	0.00296	0.00155	0.00286	0.98980	0.99811	0.99539

Table A.2: (continued)

Spectral	Penalties	NNeg	Criterion	*MRAE* Red	*MRAE* Green	*MRAE* Blue	*RMSE* Red	*RMSE* Green	*RMSE* Blue	*SSIM* Red	*SSIM* Green	*SSIM* Blue
yes	L1L1	yes	MDC	0.11489	0.04417	0.05359	0.00463	0.00285	0.00367	0.98820	0.99614	0.99444
yes	L1	yes	MDC	0.12124	0.04598	0.05690	0.00588	0.00390	0.00469	0.98456	0.99397	0.99218
yes	L1L1	no	MDC	0.11787	0.04623	0.05543	0.00463	0.00287	0.00367	0.98805	0.99600	0.99439
yes	Choi *et al.* [19]			0.06522	0.04683	0.04000	0.00135	0.00195	0.00224	0.99833	0.99686	0.99736
yes	L1	no	MDC	0.13349	0.05459	0.06087	0.00530	0.00352	0.00428	0.98709	0.99480	0.99321
no	Lap1	no	MDC	0.11959	0.06937	0.07600	0.00457	0.00395	0.00488	0.98798	0.99246	0.99002
no	Lap1	yes	MDC	0.11959	0.06938	0.07596	0.00456	0.00395	0.00488	0.98800	0.99247	0.99003
yes	L2L2	yes	MDC	0.16799	0.07204	0.07825	0.00820	0.00730	0.00683	0.97914	0.99008	0.98952
yes	L2L2	no	MDC	0.16780	0.07227	0.07833	0.00820	0.00730	0.00683	0.97914	0.99005	0.98950
yes	L2	yes	MDC	0.17209	0.07499	0.08125	0.00874	0.00776	0.00744	0.97797	0.98912	0.98857
yes	L2	no	MDC	0.17200	0.07500	0.08125	0.00874	0.00776	0.00744	0.97797	0.98912	0.98857
no	L1Spectral-Lap2	yes	MDC	0.28369	0.10015	0.13209	0.00996	0.00630	0.00912	0.95112	0.97444	0.95864
no	L1	no	MDC	0.47642	0.21258	0.24840	0.01031	0.00744	0.01037	0.93855	0.97202	0.95999
no	L1	yes	MDC	0.47686	0.21285	0.24851	0.01031	0.00744	0.01037	0.93850	0.97200	0.95998
no	L1L1	no	MDC	0.45159	0.23732	0.23445	0.00985	0.00871	0.00958	0.94297	0.96668	0.96351
no	L1L1	yes	MDC	0.45204	0.23757	0.23462	0.00985	0.00872	0.00959	0.94293	0.96667	0.96348
no	L1Spectral-Lap2	no	MDC	0.61718	0.61397	0.61015	0.05610	0.08113	0.06504	0.68096	0.59548	0.55584
no	L1Spatial-Lap2	no	MDC	1.12919	0.66826	0.56533	0.02924	0.02732	0.02919	0.84900	0.87880	0.88884
no	L1Spatial-Lap2	yes	MDC	1.17809	0.72118	0.58092	0.03148	0.03010	0.03121	0.84517	0.87070	0.88632

Table A.2: (continued)

Spectral	Penalties	NNeg	Criterion	$MRAE$ Red	$MRAE$ Green	$MRAE$ Blue	$RMSE$ Red	$RMSE$ Green	$RMSE$ Blue	$SSIM$ Red	$SSIM$ Green	$SSIM$ Blue
yes	L1Spatial-Lap2	yes	MDC	—	—	—	—	—	—	—	—	—
yes	L1Spectral-Lap2	yes	MSEC	—	—	—	—	—	—	—	—	—
yes	L1Spatial-Lap2	yes	MSEC	—	—	—	—	—	—	—	—	—
yes	Lap1	yes	MDC	—	—	—	—	—	—	—	—	—
yes	L1Spectral-Lap2	yes	MDC	—	—	—	—	—	—	—	—	—
yes	Lap1	no	MDC	—	—	—	—	—	—	—	—	—

Appendix B

Supplemental Experiments on Dispersion Model Calibration

In this appendix, we describe supplemental experiments on spectral dispersion calibration. These experiments follow our initial calibration experiments presented in Section 5.3. We conducted these supplemental experiments for two reasons: First, we sought to validate our calibration method by testing whether different variants of our method produced similar models of dispersion. Second, we needed to determine whether the non-monotonic change in the magnitude of dispersion with respect to distance from the image center, in Figures 5.3 and 5.4, is an artifact of our calibration method.

B.1 Vignetting Correction

In our first supplemental experiment, we tested if disk keypoint locations were sensitive to non-uniform shading, by applying a vignetting correction to the images of the disk pattern. We calibrated the vignetting correction model, as described in Section 3.2.2, from the white background of the disk pattern. The left side of Figure B.1 shows the dispersion pattern obtained without first correcting the disk pattern images for non-uniform shading. This dispersion pattern is almost identical to the corresponding pattern obtained with correction for non-uniform shading, shown on the right side of Figure B.1. The only observable differences are a slight improvement in the overall symmetry of the dispersion pattern, especially towards the image edges, with shading correc-

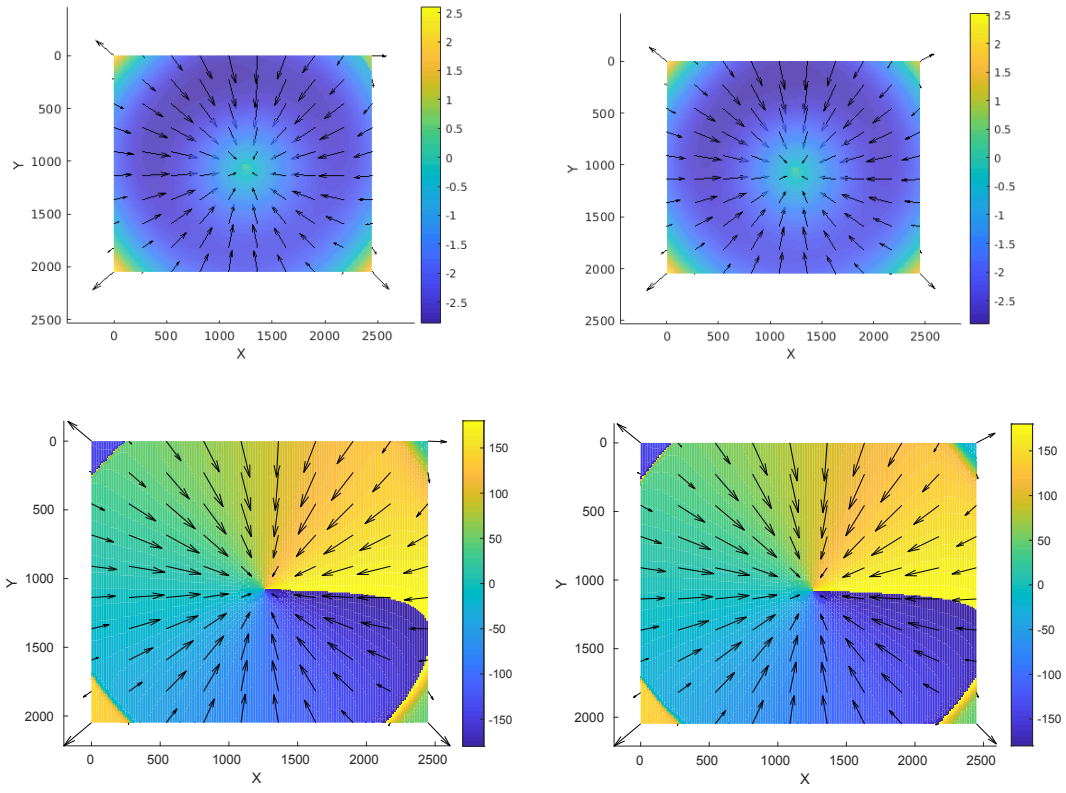
tion. While our dispersion calibration is insensitive to non-uniform shading, we chose to correct for non-uniform shading by default (as in Figures 5.3 and 5.4).

B.2 Image Registration vs. Disk Keypoints

While correcting the calibration image for non-uniform shading indicated that disk keypoint detection is robust to non-uniform shading effects, it is possible that there are other sources of bias in disk keypoint locations that are not related to shading. To conclusively validate our dispersion keypoints, we compared disk keypoints to keypoints obtained by patch-wise image registration (described in Section 3.2.3). For patch-wise image registration, we used a patch size of 64×64 pixels. We sampled patches in a grid of partially-overlapping patches, with horizontal and vertical offsets of 32 pixels between neighbouring patches. The models of dispersion we obtained using the two types of keypoints are compared in Figure B.2. Since the two algorithms for generating keypoints are very different, and since they lead to similar models of dispersion, then it is likely that both are unbiased estimators of dispersion vectors.

B.3 Polynomial vs. Thin-Plate Spline Interpolation

After validating the keypoints used to fit our models of dispersion, we assessed whether the non-monotonic change in the magnitude of dispersion with distance from the image center was an artifact of polynomial fitting. As polynomials are typically unstable outside the domain of the data to which they are fitted, we reasoned that the reversal in dispersion direction towards the edges of the image was exacerbated by underconstrained polynomials. We experimented with thin-plate spline models of dispersion as an alternative to polynomial models, fitting trivariate thin-plate splines in x, y, and λ as described in Section 3.2.4. As shown on the right side of Figure B.3, spline models are comparable to polynomial models, but can be more sensitive to local noise, which worsens near the image edges.



(c) (d)

Figure B.1: Trivariate polynomial models of the dispersion of the image for 450 nm light with respect to the image for 600 nm light, for a Computar 07I lens mounted on a Point Grey BlackFly Flea3 camera. In the left column, the heat map colours show the magnitude (a) and angle (c) of the dispersion vectors in the absence of vignetting correction prior to disk keypoint localization. The right column shows the magnitude (b) and angle (d) of the dispersion vectors computed with vignetting correction prior to disk keypoint localization. The magnitudes of dispersion vectors shown in the top row are their signed magnitudes in the direction of the image centre, with positive magnitudes chosen for vectors pointing away from the image centre. The arrows in all plots shown the direction of dispersion.

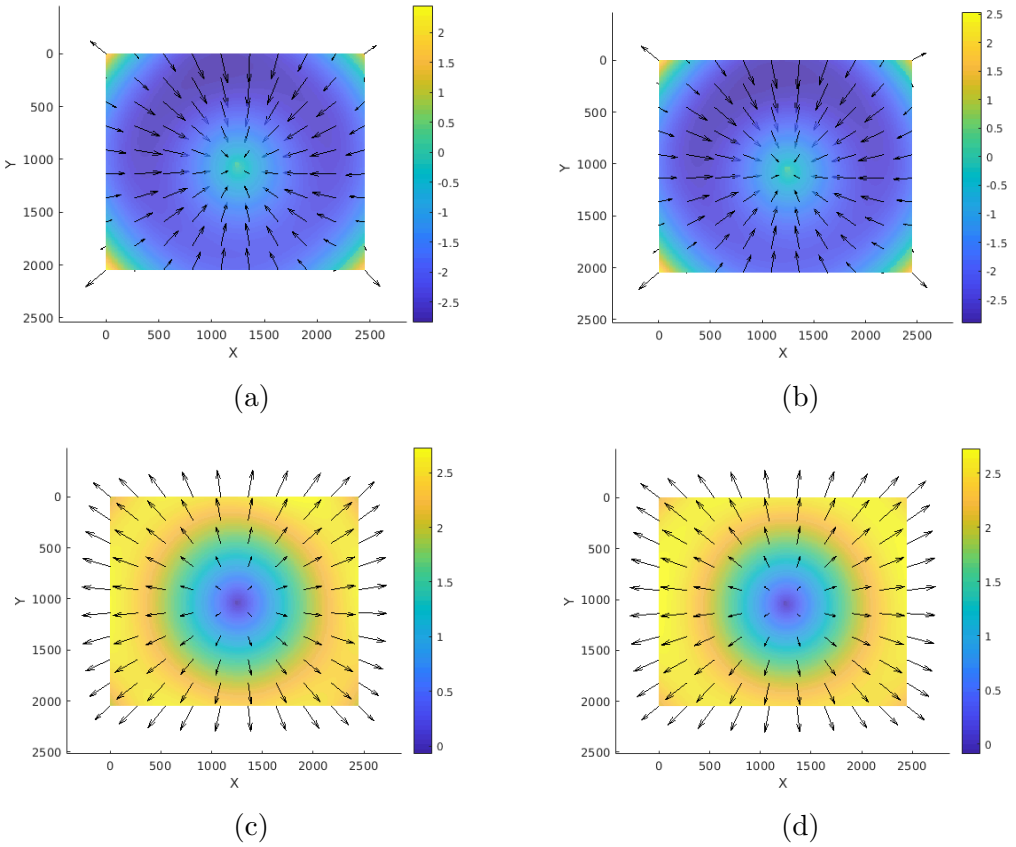
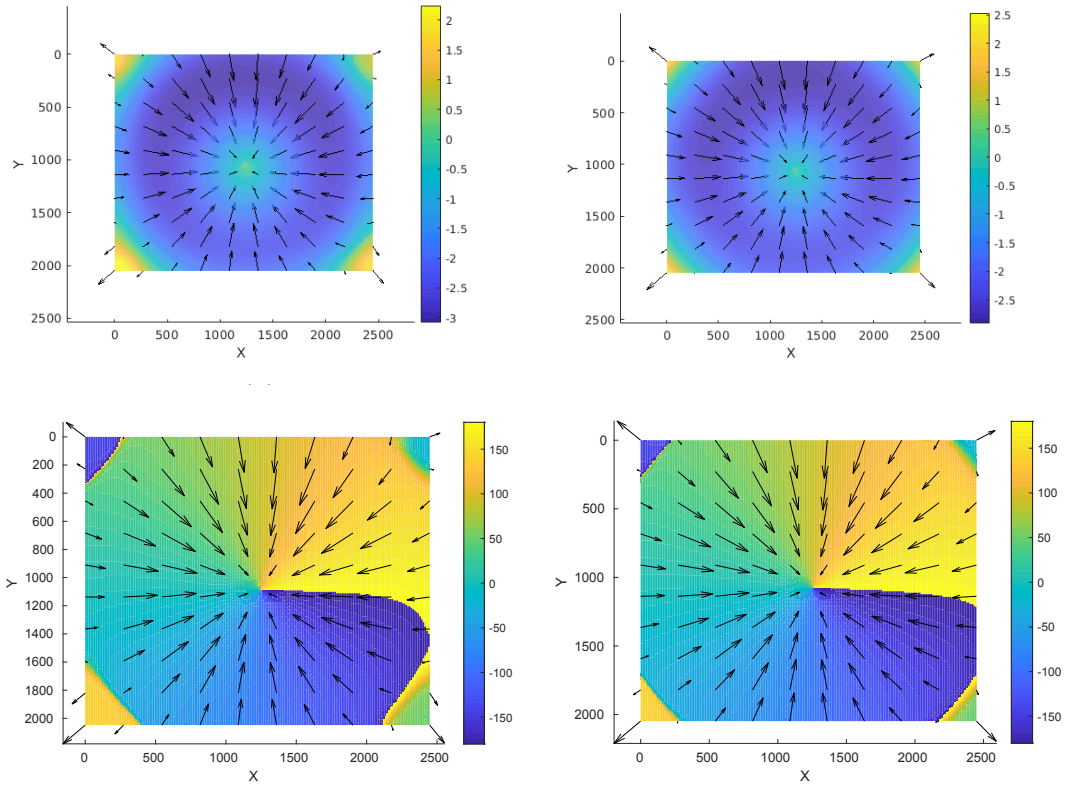


Figure B.2: Trivariate polynomial models of the dispersion of the image for 450 nm light (top row) or 700 nm light (bottom row) with respect to the image for 600 nm light, for a Computar 07I lens mounted on a Point Grey BlackFly Flea3 camera. The left column shows the model fit to keypoints obtained by patch-wise image registration between spectral bands. The right column shows the model fit to keypoints obtained by disk centre detection in individual spectral bands. Positive dispersion magnitudes indicate that the dispersion vectors point away from the image center. Arrows in the figures show the dispersion direction.



(c) (d)

Figure B.3: Alternative models of the dispersion of 450 nm light with respect to 600 nm light, for a Computar 07I lens mounted on a Point Grey BlackFly Flea3 camera. On the left, the model is a thin-plate spline model in x, y, and λ, showing that thin-plate spline models are more flexible, but may be unstable at the image borders. On the right, the model is a trivariate polynomial in x, y, and λ. The top row of plots shows the magnitudes of the radial components of dispersion vectors, with positive magnitudes chosen for components pointing away from the image centre. Note the different scales of the colour axis between the two plots. The bottom row of plots shows the angles of the dispersion vectors. The arrows in both rows of plots follow the direction of dispersion.

The same conclusions can be drawn by looking at the behaviour of spline vs. polynomial models across the visible spectrum, as shown in Figure B.4. The two different dispersion keypoint interpolations agree well except at the edges of the image, where the spline model fits more closely to the keypoints. The polynomial model will exhibit poorer extrapolation behaviour at wavelengths at the extremes of the visible spectrum than the spline model. The difference arises because the polynomial model relies on calibration data to constrain its behaviour everywhere in its spectral-spatial domain, whereas the spline model approaches a more stable linear model in regions far from calibration data. Fortunately, the camera spectral response falls off at the extremes of the visible spectrum, so the inaccuracy of the polynomial model in these regions is tolerable. As the two methods for interpolating dispersion keypoints give similar results, but since thin-plate splines are computationally expensive, both to fit, and to evaluate, for large numbers of keypoints, we normally prefer polynomial models of dispersion.

B.4 Comparison of Dispersion between Lenses

Lastly, we investigated whether dispersion was also non-monotonic with respect to distance from the image center in a Fujifilm Fujinon 1:1.4/12.5 mm lens, exchanged for the Computar 07I lens on the same Point Grey BlackFly Flea3 camera. Representative results from our experiment are shown in Figure B.5. As we did not collect as many images to calibrate dispersion for this lens, there are fewer keypoints available for fitting models of dispersion. As such, our polynomial fitting algorithm selected low degree polynomials, hiding non-monotonicity near the image borders (Figures B.5a and B.5c). In contrast, our thin-plate spline models show this monotonicity, at the expense of retaining more of the noise in the keypoints extracted from the images (Figures B.5b and B.5d). The patterns of dispersion for the two different lenses (Figures B.5 vs. 5.3) are qualitatively similar. Therefore, we suggest that, at high viewing angles, real lens and camera combinations may not follow the commonly-accepted notion that the image for longer wavelengths should be

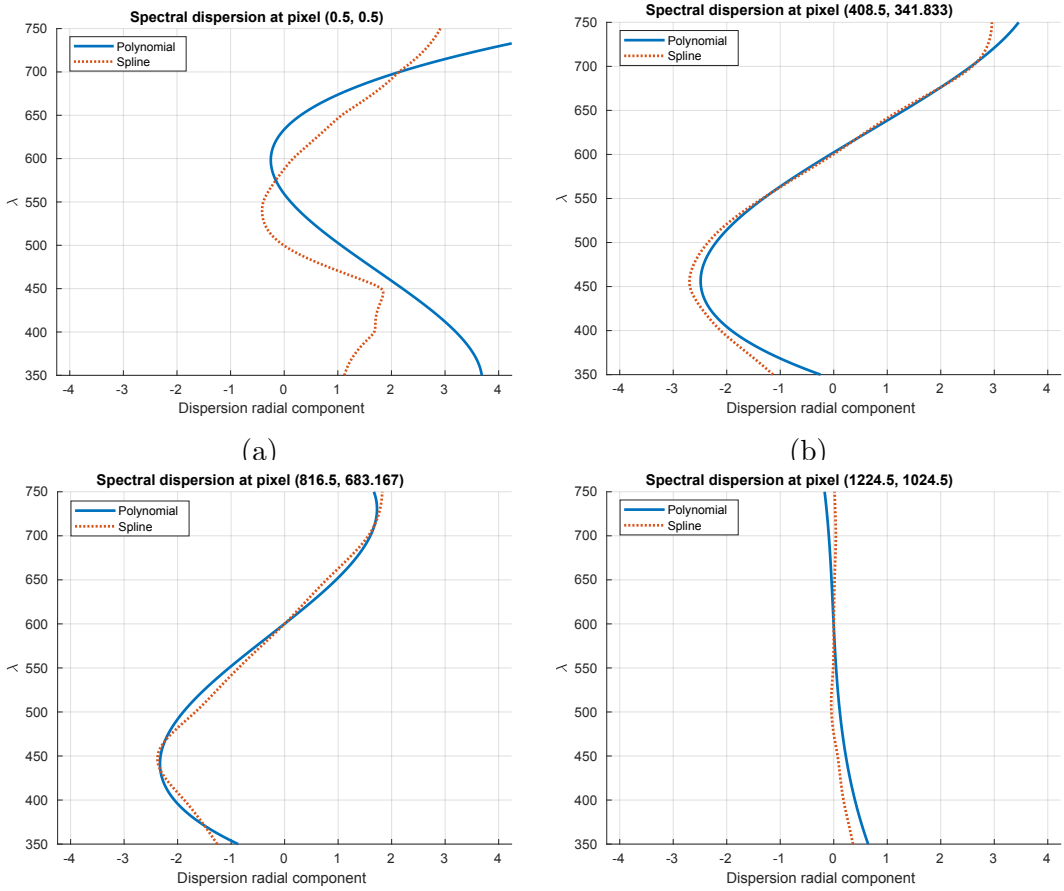


Figure B.4: Magnitudes of the radial components of dispersion vectors, with positive magnitudes chosen for components pointing away from the image centre. The magnitudes of dispersion vectors are plotted with respect to the wavelength of the light. Over the figures, from left-to-right, top-to-bottom, the image location at which the dispersion models are being evaluated moves from the image plane's top-left corner to the image center. The solid blue lines in the figures represent polynomial models of dispersion, whereas the dotted red lines represent thin-plate spline models of dispersion, both for a Computar 07I lens mounted on a Point Grey BlackFly Flea3 camera.

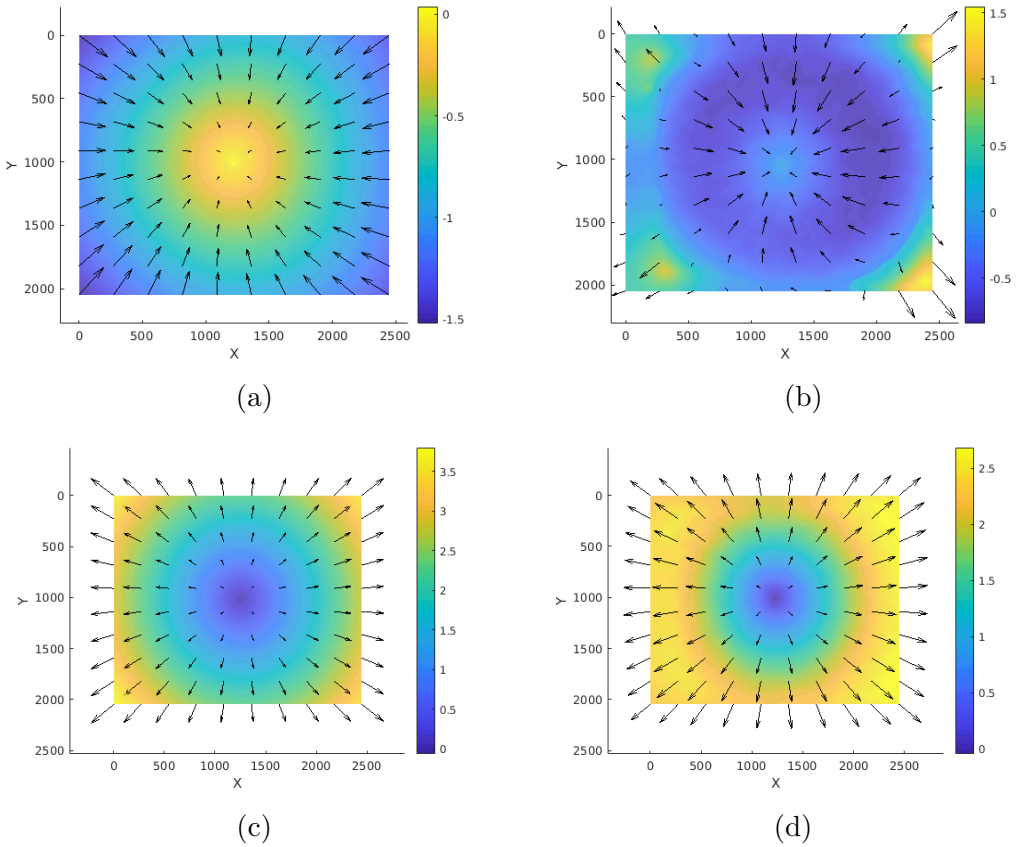


Figure B.5: Trivariate polynomials (left column) or thin-plate splines (right column) in x, y, and λ, modeling dispersion for a Fujifilm Fujinon 1:1.4/12.5 mm lens mounted on a Point Grey BlackFly Flea3 camera. The first and second rows of plots show aberration relative to 600 nm for spectral bands centered at 450 nm, and 700 nm, respectively. Positive dispersion magnitudes indicate that the dispersion vectors point away from the image center. Arrows in the figures show the dispersion direction.

magnified relative to the image for shorter wavelengths. Exploring possible physical causes for the observed dispersion patterns would be an interesting future project.

B.5 Summary

During our experimentation with different variants of our dispersion calibration method, we obtained consistent models of dispersion, suggesting that our calibration methods are reliable. We also note that the strong radial pattern in our models of dispersion conforms to our expectations. The radial pattern

follows from basic principles of optics (Section 1.1.2, Figure 1.1), and is often assumed in literature on chromatic aberration correction [75].