Visual Inference with Statistical Models for Color and Texture

A dissertation presented

by

Ayan Chakrabarti

to

The School of Engineering and Applied Sciences

in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy
in the subject of
Engineering Sciences

Harvard University
Cambridge, Massachusetts

August 2011

Dissertation Advisor: Prof. Todd Zickler **Ayan Chakrabarti**

VISUAL INFERENCE WITH STATISTICAL MODELS FOR COLOR AND TEXTURE

Abstract

Analyzing images, to estimate the underlying parameters that lead to their formation, is fundamentally an inverse problem. Since the observed image alone is usually not enough to uniquely determine these parameters, statistical models are frequently used to choose a *likely* solution from amongst those that are consistent with this observation. In this dissertation, we use such a statistical approach to develop image models and corresponding inference algorithms for two vision applications, and then explore image statistics in a new domain.

The first application seeks to segment moving objects from a static image using motion blur as a cue. We introduce an image model defined in terms of a convenient local Fourier decomposition, that captures the statistical properties of sharp edges while accounting for variations in contrast. We then use this model to derive the likelihood of a candidate kernel acting in a local region of the observed image, and combine this likelihood measure with color information for the final segmentation.

The second application addresses the problem of color constancy and involves estimating and correcting for the effect of an unknown illuminant on the colors in an observed image. We find a joint *spatio-spectral* approach to be useful here, and describe a statistical model for the colors of image sub-band coefficients. We derive an estimation algorithm to compute illuminant parameters using this model and show that it can be conveniently extended to include prior knowledge about illuminant statistics, when such knowledge is available.

Finally, we explore the statistical properties of hyperspectral images, *i.e.* those that include dense spectral measurements at each pixel, using a new database of natural scenes. We determine that the optimal basis to represent these images is separable along the spatial and spectral dimensions, and then explore statistical models for coefficients in this separable basis.

Table of Contents

Dedicated to

Ma & Baba

Acknowledgments

I would like to begin my thanking my advisor, Prof. Todd Zickler, for making graduate school a wonderful and enriching experience. Todd has been available to provide guidance, support and encouragement whenever I have needed it, for issues related to every aspect of being a researcher and more. It is difficult to imagine a better mentor. In my career, I hope to at least partly be able to emulate his approach to research and advising.

I have also had the opportunity to collaborate with a lot of wonderful people over the course of my PhD. I want to thank Prof. Keigo Hirakawa, Prof. Bill Freeman, Prof. Daniel Scharstein, Prof. Trevor Darrell, Dr. Kate Saenko, Trevor Owens and Ying Xiong. Working with them has shaped the way I approach research. Sincere thanks are also due to Mohamed-Ali Belabbas, Alan O' Connor, Phil Owrutsky, Sanjeev Koppal, Ioannis Gkioulekas, Ritwik Kumar, Moritz Baecher, and all my other friends and colleagues here at SEAS and the GVI group.

Ankur Agrawal and Anand Sampath were my flatmates for most of the time I was a graduate student, and I have come to think of and depend on them as family. The same goes for my friend Punyashloka Biswal and his parents, whom I have known for twenty years now. I also want to thank my friends Siddharth Garg, Sandeep Bhadra, Aditya Gopalan, Arun Koshy, Smita Gopinath, Ajit Narayanan, Siddharth Tata, Narayanan Ramachandran, Akhil Basha and Rachna Pande for lively and interesting conversations, technical and otherwise.

But above all, I want to thank my parents, Smt. Basabi Chakrabarti and Shri. Amarnath Chakrabarti. My graduation would perhaps have meant the most to them, and it is my greatest regret that they were not around to witness it. Everything I do has been, and will be, an attempt to do justice to their love, encouragement, and trust in me.

1

Introduction

"Facts are stubborn things, but statistics are more pliable."

— Mark Twain

The image of a scene acquired by a camera depends on a variety of factors— scene geometry, material properties, illumination, sensor noise, camera and object motion, *etc.* While knowing these properties is of value to vision systems, their contributions to the final observed image are usually confounding. Estimation is therefore often an ill-posed problem, meaning that there are multiple solutions that explain the observed image and satisfy the image formation model equally well. This chapter describes the utility of statistical modeling to solve such ill-posed estimation problems, where a unique solution is arrived at by requiring it to be likely as well as consistent with the observation. We give an overview of the approaches commonly used to define statistical models, and to exploit them during inference. We also discuss other estimation strategies used for visual tasks, and describe how they relate to these statistical approaches. Finally, we provide an outline for the rest of this dissertation and list its contributions.

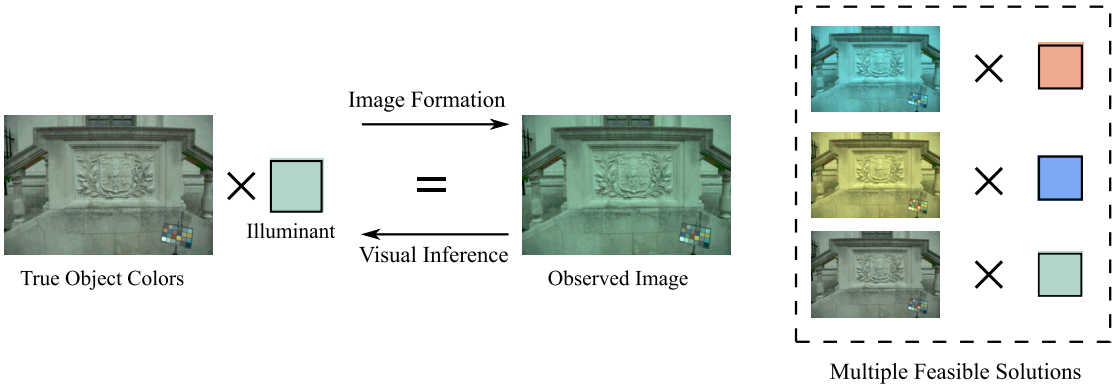


Figure 1.1: Visual inference usually involves solving ill-posed inverse problems. Shown here is the example for *color constancy* (explored in depth in Chapter 3), where the observed image is a function of true scene colors and the scene illuminant, both of which are unknown. Estimating these from the image is an under-determined problem, because there exist many plausible solutions consistent with the observation.

1.1 Ill-posed Problems in Vision

A variety of computer vision tasks involve solving inverse problems. Depending on the formulation, the observed image (or set of images) is modeled as a function of multiple unknown variables, and the task is to infer one or more of these variables from the observation. Examples include estimating the albedo and surface normals (for photometric stereo), the scene illuminant and object colors (for color constancy), the blur point spread function (PSF) and latent sharp image (for deblurring), *etc.* Even applications such as object recognition and detection can be interpreted as inverse problems, where the unknown variables are the object or category label along with factors such as pose and intra-class variability that contribute to the appearance of the object in the observed image.

These inverse problems are often under-determined, *i.e.* there exist multiple combinations of the unknown variables that all explain the observations equally well. At other times, while there is a unique solution under ideal circumstances, the inverse map is "ill-conditioned", meaning that estimation is very sensitive to observation noise. To address the ill-posed nature of these problems, researchers use knowledge of the statistical properties of one or more of the unknown variables to

identify solutions that are probable, from amongst the set that are physically plausible (*i.e.* consistent with the observation).

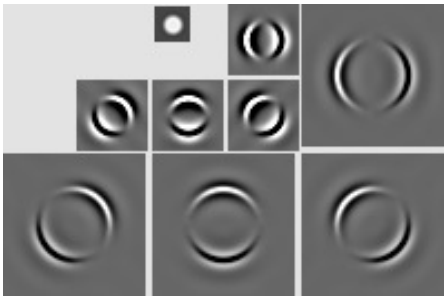
Typically, these problems are parametrized such that one of the unknown variables is a canonical image of the scene, *i.e.* one captured with some standard values for the other unknown variables (under a standard illuminant, with no noise or blur, *etc.*), and therefore a lot of research in computer vision has sought to define accurate and tractable statistical models for natural images. These models are then used, occasionally along with models for the other scene parameters, to derive the desired estimator. In the rest of this chapter, we discuss examples of image models and inference strategies commonly used to solve real-world vision tasks, and relate them to a few other common approaches that address ill-posed estimation problems. We then state the contributions of this dissertation, and outline the contents of the remaining chapters.

1.2 Image Models

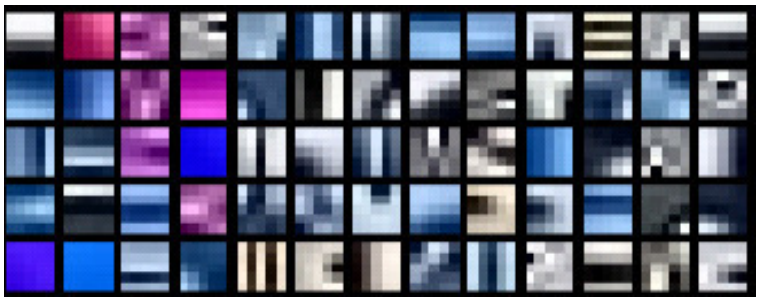
An accurate description of natural image statistics is crucial for many statistical inference methods in vision, and therefore, a lot of research has been aimed at developing image models that are accurate yet tractable. The first step in defining a model involves choosing an appropriate image representation. Rather than modeling pixels directly, it is common to define these models in an appropriately chosen transform domain. Formally, a linear map $\boldsymbol{\Phi}$ is applied to an image $\boldsymbol{I}$ to yield a set of coefficients $\boldsymbol{S} = \boldsymbol{\Phi}^T \boldsymbol{I}$. A probability distribution is then assigned to the elements of $\boldsymbol{S}$, and $\boldsymbol{\Phi}$ is usually chosen such that these elements can be treated as being independent, or having limited dependence. Inference is carried out on these coefficients, and if the task is to estimate the canonical image, an inverse map $\boldsymbol{\Phi}^{-1}$ is applied on the corresponding estimated canonical coefficients.

The transform $\boldsymbol{\Phi}$ is usually chosen to decompose an image based on spatial frequency, scale, orientation, *etc.* Examples include using the block discrete cosine transform (DCT) [1], or a basis derived from principal component analysis (PCA) [2] on image patches. These typically en-

DCT [1]



Steerable Pyramids [5]



Redundant dictionary learned from data [6]

Figure 1.2: Image representations. Statistical models for images are typically defined in terms of coefficients in some chosen transform domain. Shown here are the basis elements for three example transform domains.

sure optimal *energy compaction*, and ensure that the transform coefficients are uncorrelated. Other approaches, such as Gaussian and Laplacian pyramids [3] and wavelets [4], use a multi-scale approach to define the transform $\mathbf{\Phi}$. They decompose an image by applying sets of filters that are scaled versions of each other. The corresponding coefficients $\boldsymbol{S}$ can then be grouped according to these scales, and interpreted as forming an *image pyramid*.

The canonical versions of these transforms are defined such that the map $\mathbf{\Phi}$ corresponds to an orthonormal basis. However, researchers have often found over-complete representations to be more powerful during inference. Such representations can be formed simply by not including sub-sampling steps with pyramid-based transforms, or by considering overlapping patches with DCT and PCA-based transforms. Thus, they achieve over-completeness by considering various shifted versions of the corresponding complete transforms. In contrast, steerable pyramids [5] are

designed from scratch to be over-complete without the constraints of having an orthonormal basis, and include basis filters at various orientations.

The coefficients corresponding to an over-complete transform are clearly not independent, but they are often treated as such for convenience. For example, coefficients corresponding to overlapping patches may be modeled independently. Note that in applications where the goal is to estimate a corrected image, the overlapping corrected coefficients are also estimated independently, and therefore need not be consistent. The inverse map Φ^{-1} is then chosen accordingly to be a pseudo-inverse that harmonizes these inconsistencies, for example, by computing an average of the overlapping corrected patches.

Another approach, that is common with over-complete transforms, is to compute the coefficient vector through a minimization approach instead of a linear map [7, 8]. In fact, this minimization is often done simultaneously with the inference step. The set of basis elements here is thought of as a redundant *dictionary*, and the corresponding coefficient vector is chosen so as to be sparse, *i.e.* image regions are described as a linear combination of a small number of dictionary elements. In addition to using standard representations (as described above), optimal dictionaries that admit compact representations in this framework can also be learned from training data [6, 9].

In addition to the transforms described above to model generic images or patches, some methods choose image representations tailored to specific discriminative tasks. These transforms yield coefficient vectors (sometimes called feature vectors) that carry optimal information for some classification or recognition problem. They can correspond to subsets of standard transforms [10], statistical summaries of standard transform coefficients [11], or be learned from data [12].

Once an appropriate transform has been chosen, an image model is defined by assigning appropriate distributions to the coefficients under that transform. Coefficients which correspond to a local spatial average (also referred to as DC coefficients or scaling coefficients), are typically not modeled, or assigned a uniform distribution. The remaining coefficients are usually found to have zero-mean symmetric distributions. These are modeled using some convenient parametric form,

such as the Gaussian or Laplace distributions, generalized-Gaussians, discrete or continuous mixtures of Gaussians, *etc.* When available, the parameters of these distributions can be set from training data, either by fitting the empirical histograms or by validation to yield minimum error during inference. Coefficients may be modeled independently, though sometimes joint models are defined by encoding dependencies between the parameters of the per-coefficient distributions. For example, coefficients may be modeled as Gaussian mixtures, with a statistical model over the mixture parameters for different coefficients [13, 14].

While some of the above models have been used for color images (such as redundant dictionaries for color patches [6]), a vast majority of work in image modeling has focused on greyscale data. However, color statistics have been exploited in applications that explicitly depend on them. Demosaicking applications [15] use models that encode the correlation between different color channels to estimate a full color image from the sub-sampled data captured by a typical camera. They are loosely based on the fact that discontinuous changes in intensities for different channels co-occur at material boundaries. Image models for color constancy, on the other hand, have by and large focused on the distributions of color vectors for individual pixels (see Chapter 3 for an extended discussion).

The list of modeling strategies discussed in this section is far from being exhaustive. Many specialized models have been developed for specific applications, such as those involving Bayesian networks and Markov random fields to encode relationships between pixels and regions [16, 17]. Scene parsing and recognition applications frequently use image models based on semantic content (object categories [18], identities of people [19], *etc.*) rather than (or in addition to) low-level features. Overall, there is no one image model that is optimal for all applications and a choice needs to be made based on the task at hand. None of these models completely capture all the statistical properties of natural scenes— sampling from them will not yield a plausible image. Instead, one aims to choose a model that encodes all image properties that are informative for the given task, and allows inference at reasonable computational cost.

Table 1.1: Examples of observation models for different inference tasks

Task	**Model**	**Remarks**
Denoising	$\boldsymbol{J} = \boldsymbol{I} + \boldsymbol{Z}$	$\boldsymbol{Z}$ = Unknown observation noise.
Demosaicking	$\boldsymbol{J} = \boldsymbol{D}\boldsymbol{I} + \boldsymbol{Z}$	$\boldsymbol{D}$ = Known color sub-sampling pattern.
Deblurring	$\boldsymbol{J} = \boldsymbol{I} * \boldsymbol{k} + \boldsymbol{Z}$	$\boldsymbol{k}$ = Unknown blur point spread function.
Intrinsic Images	$\boldsymbol{J}[\boldsymbol{n}] = s[\boldsymbol{n}]\boldsymbol{I}[\boldsymbol{n}]$	$s[\boldsymbol{n}]$ = Unknown shading coefficient at pixel $\boldsymbol{n}$.
Color Constancy	$\boldsymbol{J}[\boldsymbol{n}] = \boldsymbol{M}\boldsymbol{I}[\boldsymbol{n}]$	$\boldsymbol{M}$ = Unknown scene illuminant color.

1.3 Statistical Inference

A typical statistical inference task in vision is set up as follows: the observation $\boldsymbol{J}$ is modeled as $\boldsymbol{J} = f(\boldsymbol{I}; \boldsymbol{\theta})$, *i.e.* a function of some canonical image $\boldsymbol{I}$, which is assumed to be drawn from an appropriately chosen distribution $p(\boldsymbol{I})$, and other (potentially unknown) camera/environmental parameters $\boldsymbol{\theta}$. Examples of $f(\cdot)$ for some applications are listed in Table 1.1. Note that in all these examples, there is no unique solution for the unknown variables.

Given the observation $\boldsymbol{J}$, the task may be to infer either the canonical image $\boldsymbol{I}$, one of the parameters $\boldsymbol{\theta}$, or both. Since the function $f(\cdot)$ is not directly invertible, the solution (for $\boldsymbol{I}$ or $\boldsymbol{\theta}$, as the case may be) is chosen so as to maximize some statistically-motivated objective function. In some cases, this objective function is simply chosen to be $p(\boldsymbol{I}|\boldsymbol{J})$, $p(\boldsymbol{\theta}|\boldsymbol{J})$ or $p(\boldsymbol{I}, \boldsymbol{\theta}|\boldsymbol{J})$, *i.e.* the likelihoods of candidate images, scene parameters, or combinations of the two, conditioned on the observation $\boldsymbol{J}$. To derive these likelihoods, one first defines the joint conditional density over all unknown parameters as,

$$p(\boldsymbol{I}, \boldsymbol{\theta}|\boldsymbol{J}) \propto p(\boldsymbol{I}, \boldsymbol{\theta}, \boldsymbol{J}) = \delta\left(\boldsymbol{J} - f(\boldsymbol{I}; \boldsymbol{\theta})\right) p(\boldsymbol{I}) p(\boldsymbol{\theta}), \tag{1.1}$$

where the delta function $\delta(\cdot)$ enforces consistency between $\boldsymbol{J}, \boldsymbol{I}$ and $\boldsymbol{\theta}$ according to the observation model in $f(\cdot)$. When available, prior knowledge of the statistics of the parameters $\boldsymbol{\theta}$ is included in the $p(\boldsymbol{\theta})$ term, otherwise it is omitted and all values of $\boldsymbol{\theta}$ are assumed equally likely. The objective

function is then computed by marginalizing over all parameters other than those being estimated (for example, $p(\boldsymbol{I}|\boldsymbol{J})$ is computed by marginalizing over $\boldsymbol{\theta}$). Note that the solutions for $\boldsymbol{I}$ and $\boldsymbol{\theta}$ obtained by maximizing their joint conditional likelihood $p(\boldsymbol{I}, \boldsymbol{\theta}|\boldsymbol{J})$ need not be the same as those derived from their individual likelihoods $p(\boldsymbol{I}|\boldsymbol{J})$ and $p(\boldsymbol{\theta}|\boldsymbol{J})$.

Rather than set the objective to these conditional likelihoods, some methods define a notion of the cost $C(\cdot, \cdot)$ of choosing an estimate $\boldsymbol{I}$ or $\boldsymbol{\theta}$, when the true value is $\boldsymbol{I}'$ or $\boldsymbol{\theta}'$ respectively. The estimate is then chosen to minimize the expected value of this cost, under the conditional distribution for the true estimate computed as above. For example, the canonical image $\boldsymbol{I}$ may be estimated as

$$\boldsymbol{I} = \arg\min_{\boldsymbol{I}} \int C(\boldsymbol{I}, \boldsymbol{I}') p(\boldsymbol{I}'|\boldsymbol{J}) d\boldsymbol{I}'. \tag{1.2}$$

The cost C usually corresponds to some form of error (such as the sum of squared differences, 0-1 error, *etc.*) between the estimate and true value. Minimizing the expected error can lead to more robust estimates— as illustrated in Fig. 1.3, in some cases the conditional distribution itself is *noisy* and choosing the most likely value may correspond to an isolated peak. The integration in (1.2) can be thought of as applying a *smoothing* operation to this distribution prior to estimation.

To compute the estimate from these objective functions, one has to solve the corresponding optimization problem. While there is sometimes a closed form expression for this solution, iterative strategies are frequently required. These include gradient descent methods, iteratively solving for a subset of the unknowns keeping the others constant, successively approximating the objective function near the expected solution with a functional form that is easy to optimize, *etc.* Overall, models and inference strategies are chosen so as to yield accurate estimates with minimal computational expense. Sometimes, this involves choosing models with simplifying assumptions (for example, treating overlapping patches as independent) to keep estimation tractable. In other cases, the inference strategy is adapted for optimum performance on a specific application. For instance, in many deblurring algorithms [20, 21] that seek to estimate the latent sharp image $\boldsymbol{I}$ from a blurry

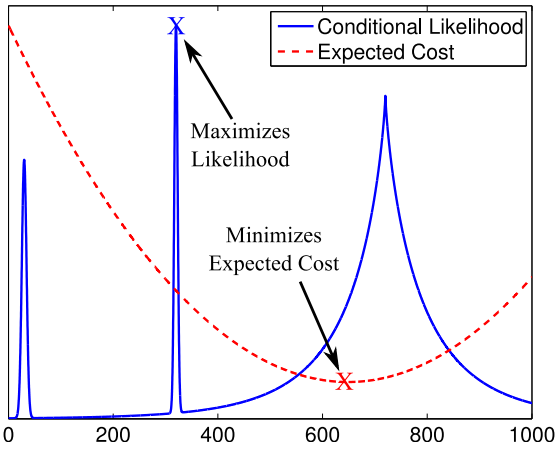


Figure 1.3: Inference strategies. Shown here is the conditional likelihood for the variable being estimated, along with the expected cost under that likelihood. In this example, the likelihood function itself has a few peaks that are isolated from the remaining region of high probability, one of which corresponds to the single most likely value. Minimizing the expected cost (the squared-difference cost is used here) effectively smooths out these peaks, and provides a more stable estimate.

observation $\boldsymbol{J}$, a two-step estimation strategy is used— the blur parameter $\boldsymbol{k}$ is first estimated by maximizing $p(\boldsymbol{k}|\boldsymbol{J})$ (*i.e.* marginalizing over $\boldsymbol{I}$), and then the sharp image $\boldsymbol{I}$ is computed by maximizing $p(\boldsymbol{I}|\boldsymbol{J},\boldsymbol{k})$.

1.4 Related Approaches

In this section, we given an overview of some other commonly used approaches for solving ill-posed estimation problems. While they do not explicitly use statistical models, we shall see that they do leverage statistical information *implicitly*.

1.4.1 Regularization

For a variety of inference problems in vision, estimation is done through a technique called *regularization*. The observation $\boldsymbol{J}$ is related to the parameter of interest, say the canonical image $\boldsymbol{I}$ as $\boldsymbol{J} = f(\boldsymbol{I})$. In contrast to the examples in Table 1.1, in this setup $f(\cdot)$ itself does not typically account for observation noise. $\boldsymbol{I}$ is then estimated by minimizing a cost function of the

form:

$$\boldsymbol{I} = \arg\min_{\boldsymbol{I}} \|\boldsymbol{J} - f(\boldsymbol{I})\|_e + \alpha\|\boldsymbol{I}\|_r, \tag{1.3}$$

where the first term penalizes the error in terms of the model (under some norm e), and the second term *regularizes* the estimation problem to compensate for the fact that $f(\cdot)$ can not be directly inverted. The parameter α is a scalar that weighs the relative contributions of these two terms. Although this approach is often motivated without an explicit statistical characterization, choosing the parameter α and the forms of the error and regularization terms amounts to making assumptions about the statistics of the image and observation noise.

In fact, we can show that there is an equivalent statistical approach that leads to the exact same expression as in (1.3). We update the observation model to include a noise term Z as

$$\boldsymbol{J} = \tilde{f}(\boldsymbol{I}, \boldsymbol{Z}) = f(\boldsymbol{I}) + \boldsymbol{Z}, \tag{1.4}$$

and choose the distributions for $\boldsymbol{Z}$ and $\boldsymbol{I}$ to match (1.3) as

$$p(\boldsymbol{Z}) \propto \exp\left(-\frac{\|\boldsymbol{Z}\|_e}{\beta}\right), \quad p(\boldsymbol{I}) \propto \exp\left(-\frac{\|\boldsymbol{I}\|_r}{\gamma}\right). \tag{1.5}$$

We can now compute $\boldsymbol{I}$ in this formulation by maximizing $p(\boldsymbol{I}|\boldsymbol{J})$ as

$$\begin{aligned}\boldsymbol{I} &= \arg\max_{\boldsymbol{I}} p(\boldsymbol{I}|\boldsymbol{J}) = \arg\max_{\boldsymbol{I}} \int \delta\left(\boldsymbol{J} - \tilde{f}(\boldsymbol{I}, \boldsymbol{Z})\right) \exp\left(-\frac{\|\boldsymbol{Z}\|_e}{\beta}\right) \exp\left(-\frac{\|\boldsymbol{I}\|_r}{\gamma}\right) d\boldsymbol{Z} \\ &= \arg\max_{\boldsymbol{I}} \exp\left(-\frac{\|\boldsymbol{J} - f(\boldsymbol{I})\|_e}{\beta} - \frac{\|\boldsymbol{I}\|_r}{\gamma}\right) = \arg\min_{\boldsymbol{I}} \|\boldsymbol{J} - f(\boldsymbol{I})\|_e + \frac{\beta}{\gamma}\|\boldsymbol{I}\|_r,\end{aligned} \tag{1.6}$$

which is equivalent to the regularized estimate from (1.3), when $\alpha = \beta/\gamma$. Therefore, the error penalty in (1.3) can be interpreted as the log-likelihood of observation noise and the regularization term as the log-likelihood of a candidate image. However, this connection is not always made when setting up a regularization framework. This is appropriate in many cases when the models in (1.5) are poor fits to observed empirical distributions, but the overall estimation strategy leads to desirable results in practice.

1.4.2 Machine Learning

Learning-based methods like boosting, support vector classification and regression, *etc.* take a different approach to estimation. They *learn* a mapping $g(\cdot)$ from the observation $\boldsymbol{J}$ to the variable of interest, say $\boldsymbol{\theta}$. This mapping is chosen automatically from a space of functions $\mathcal{H}$ (sometimes called the *hypothesis space*) so as to minimize estimation error on a training set of input-output pairs $\{\boldsymbol{J}_t, \boldsymbol{\theta}_t\}_{t=1}^{T}$. To prevent "over-fitting", *i.e.* choosing a complex mapping that fits exactly to the data, including small perturbations possibly caused by noise, these approaches also seek to simultaneously minimize some notion of complexity (often measured in terms of smoothness) of the chosen map. Formally, the map $g(\cdot)$ is chosen as

$$g = \arg\min_{g \in \mathcal{H}} \sum_{t=1}^{T} \frac{1}{T} \operatorname{err}\left(g(\boldsymbol{J}_t); \boldsymbol{\theta}_t\right) + \alpha \left\|\boldsymbol{g}\right\|_{\mathcal{H}}, \tag{1.7}$$

where the first term corresponds to error on the training set, and the second measures complexity as described above. Note that the expression in (1.7) can itself be interpreted as a regularized estimation framework, used here to choose the map $g(\cdot)$ based on training data. Once this inverse map $g(\cdot)$ has been computed, it is applied directly to novel observations to compute the corresponding estimates of $\boldsymbol{\theta}$.

There is a clear parallel between picking $g(\cdot)$ as per (1.7) and minimizing the expected cost under a conditional distribution as described in Sec. 1.3. The summation over the training set in (1.7) approximates the joint distribution $p(\boldsymbol{J}, \boldsymbol{\theta})$, and corresponds to choosing a map that minimizes the expected error value under this distribution. This approach has various advantages—one avoids making the assumptions that are implicit in defining an observation model, assigning statistical distributions to the image and other parameters, *etc.*. All these decisions are assumed to be made automatically by the training algorithm in (1.7) for optimal performance. This can be useful when the actual observation model and statistics are unknown or hard to model, and the learning method can be trusted to *discover* the attributes of the observation that are most useful for computing an accurate estimate.

However, learning-based methods make a design choice in picking the hypothesis space $\mathcal{H}$. The training step in (1.7) can at best learn the most accurate map in $\mathcal{H}$, but $\mathcal{H}$ itself has to be *expressive* enough and contain the class of functions appropriate for the estimation task. When this task is complex, the hypothesis space $\mathcal{H}$ must be chosen to include adequately complex functions, and a lot more training data is required to reliably learn the mapping $g(\cdot)$. In contrast, if the observation model is known and a parametric form for image and parameter distributions has been chosen, the parameters of these distributions can be learned with far fewer training samples and using statistical inference methods may prove more convenient. Furthermore as noted in Sec. 1.3, the optimal estimates derived from statistical models often do not have closed form solutions, but it is impractical to define a hypothesis space $\mathcal{H}$ of maps that involve iterative computation.

In practice, statistical inference and machine learning serve as complementary approaches for visual inference, with each being optimal for different types of applications, based on the complexity of the problem and availability of training data. While it may be tempting to always use a purely data-driven machine learning approach, the use of explicit statistical models and inference approaches, that incorporate useful domain knowledge, can yield dividends in terms of improved accuracy and speed for many applications .

1.5 Dissertation Outline

This dissertation seeks to demonstrate the efficacy of statistical modeling for visual inference, and the design choices that need to be made to ensure accurate yet tractable inference for different tasks. Accordingly, we present novel statistical estimation algorithms for two visual inference tasks. Each involves the use of a different image model, individually adapted for each application to encode different aspects of the statistical structure present in natural images.

Chapter 2 describes an algorithm to estimate the parameters of spatially-varying motion blur acting on an image, allowing the segmentation of moving objects from a still image using

this blur as a cue. We use an image model to describe texture content in greyscale images to estimate blur parameters, and augment it with per-pixel color appearance models to carry out the segmentation. Chapter 3 tackles the problem of color constancy, *i.e.* correcting for the color cast in an observed image caused by the spectrum of the scene illumination. This lets us automatically estimate a canonical white-balanced image, in which object colors are independent of illumination and can serve as, say, stable cues for recognition. For this task, we introduce a joint *spatio-spectral* model that encodes the color statistics of spatial sub-band coefficients.

In Chapter 4, we investigate the statistical structure of hyperspectral images, and discuss modeling strategies in that domain. Hyperspectral images provide higher-resolution spectral measurements of the incident light at each pixel, and are expected to be useful for various vision applications. Using a new database of hyperspectral images of real-world scenes, we explore the statistical properties that are likely to be useful when building hyperspectral image models for statistical inference.

With the goal of ensuring that the research presented here is reproducible, we have made prototype implementations of the presented algorithms as well as all newly collected image data available for download [22–24]. Also, note that early versions of this research have appeared in the following publications:

- Ayan Chakrabarti, Keigo Hirakawa, and Todd Zickler, "Color Constancy Beyond Bags of Pixels," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

- Ayan Chakrabarti, Todd Zickler, and William T. Freeman, "Analyzing Spatially-varying Blur," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

- Ayan Chakrabarti and Todd Zickler, "Statistics of Real-world Hyperspectral Images," in *Pro-*

ceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

We conclude the dissertation in Chapter 5 with an overview of the contributions, and describe opportunities to expand on the insights developed here and apply them for use in other applications.

2

Estimating Blur

"The English can't draw a line without blurring it."

— Winston Churchill

The image of an object or region is blurred when a pixel receives light from multiple scene locations, due to motion or defocus. This induced blur can vary spatially across the image plane as a function of scene content, such as depth and object motion parameters. In this chapter, we describe an algorithm that uses spatially-varying motion blur as a cue to segment out a moving object from a still image. At the core of this method is an image model that encodes the statistical properties of sharp edges, independent of their *contrast*. A local Fourier-based representation allows us to use this model for tractable inference, to compute the likelihood of a local region in the observed image being blurred by a kernel from a candidate set. This likelihood measure is then combined with a color model in a Markov random field (MRF) framework to carry out the actual segmentation. The method is evaluated on a diverse set of real images captured using a variety of cameras, and found to perform well.

Figure 2.1: What's moving ? We seek to segment out a moving object (right) from a static single image (left), using motion blur as a cue. We develop an image model and inference method that allows us to determine the parameters of this motion blur, as well as the regions it affects.

2.1 Introduction

Blur refers to image degradation caused by a pixel recording light from multiple scene points, and its common causes include camera shake, defocus, and object motion. Blur is characterized by a *kernel* or point-spread function (PSF), which corresponds to the recorded image of a single scene point of unit intensity. In a typical application this kernel as well as the corresponding *latent* sharp image are unknown, and hence estimation is ill-posed. Moreover, when the cause of blur is defocus or object motion, the kernel is not constant and changes across the image plane. Such spatially-varying blur is even harder to analyze than its uniform counterpart because of the larger number of unknown parameters. As a result, most methods for estimating spatially-varying blur require multiple images, an initial segmentation, or something else to augment the seemingly meager information available in a single photograph.

The limited success with spatially-varying blur lies in stark contrast to recent advances with uniform blur (as caused by camera shake). In the latter case, one can now estimate and reverse the effects of this kind of blur through a variety of methods with reasonable success, even when the (spatially-uniform) kernel is of a complex, non-parametric form [20, 21, 25–30]. The difference in the spatially-varying case is that blur must be inferred locally, using many fewer observations than are available in a large, uniformly-blurred image. And this means that one must use stronger image models to be able to extract the required information from a reduced set of measurements, while

also ensuring that inference is tractable.

This chapter describes a novel cue for analyzing blur locally and reasoning about non-uniform cases. Starting with a standard sub-band decomposition (a "local" Fourier transform), we introduce a probability model for unblurred natural images that is simple enough to "play well" with the decomposition but powerful enough for accurate inference. This gives us a robust and efficient likelihood measure for a small image window being blurred by a given kernel. This cue is then applied to the problem of segmenting motion-blurred regions from otherwise sharp images and simultaneously selecting the blur kernel acting on the affected region. We evaluate this approach using a diverse collection of images that each contain a single moving object. It yields satisfactory estimates of the blur kernel in most cases, and is able to obtain useful segmentations when combined with color information.

2.2 Related work

Inferring the blur kernel from a blurred image requires simultaneously reasoning about the kernel and the latent sharp image on which it operates. To address the problem, one must define the family of blur kernels to be considered, as well as an image model which encodes the statistical properties that distinguish sharp images from their blurred counterparts. Existing methods differ in terms of the types of blur they consider, the observations they use as input, and whether or not they allow the kernel to vary spatially.

When the blur is caused by camera shake and is spatially-uniform, one has the good fortune of being able to accumulate evidence across the entire image plane, and as a result, one can afford to consider a very general class of blur kernels. In this context, a variety of recent techniques have shown that it is possible to recover non-parametric and fairly arbitrary blur kernels, such as those induced by camera shake, from as little as one image [20, 21, 25–28]. Even in instances of camera shake when the actual kernel varies spatially, because the camera motion involves rotation

or out of plane movement, these varying kernels can be related by using a *homography* to relate images corresponding to different camera positions and orientations. Since the parameters of this homography are still global, they can be estimated with reasonable accuracy by pooling cues from across the image [29, 30]. One general insight that is drawn from these works is that instead of simultaneously estimating the blur kernel and a single sharp image that best explain a given input, it is often preferable to first estimate the blur kernel (say, from its conditional distribution) by marginalizing over all consistent sharp images. Levin et al. [21] refer to this process as "MAP$_k$ estimation", and we will use it here.

Blur caused by motion or defocus usually varies spatially across an image, and in these cases, one must infer the blur kernels largely using local evidence. To succeed at this task, most methods consider a more constrained family of blur kernels, and they incorporate more input than a single image. When two or more images are available, one can exploit the differences between blur and sensor noise [31] or the required consistency between blur and apparent motion [32–35] or blur and depth [36, 37]. As an alternative to using two or more images, one can use a single image but assume that a solution to the foreground/background matting problem is given as input [38–40]. Finally, one may be able to successfully use a single image, but with special capture conditions to exaggerate the effect of the blur. This is the approach taken in [41], where images captured with a coded aperture are used (along with additional user input) to estimate defocus blur that varies spatially as a function of scene depth.

More related to the method presented here is the pioneering effort of Levin [42], who also considers the case with just a single image, acquired from a regular camera, as input. The idea is to segment an image into blurred and non-blurred regions and to estimate the PSFs by exploit differences in the distribution of intensity gradients within the two types of regions. These relatively simple image statistics allow compelling results in some cases, but they fail dramatically in others. One of our primary motivations in this chapter will be to understand these failures and develop stronger computational tools to eliminate them.

2.3 Observation Model

Let $y[\boldsymbol{n}]$ denote the observed image, with $\boldsymbol{n} \in \mathbb{R}^2$ indicating the location of a pixel. We shall consider the case where $y[\boldsymbol{n}]$ is a single-channel linear (*i.e.* not gamma-corrected) greyscale image. Let $x[\boldsymbol{n}]$ be the corresponding latent sharp image, *i.e.* the image that would have been captured in the absence of any blur or noise. The images x and y can then be related as

$$y[\boldsymbol{n}] = \left[\sum_{\boldsymbol{n}'} x[\boldsymbol{n}']\, k_{\boldsymbol{n}'}[\boldsymbol{n} - \boldsymbol{n}'] \right] + z[\boldsymbol{n}], \tag{2.1}$$

where z is sensor noise and $k_{\boldsymbol{n}}$ is the blur kernel acting at pixel location $\boldsymbol{n}$. Therefore, each pixel $\boldsymbol{n}$ accumulates light from multiple neighboring pixels $\boldsymbol{n}'$, weighted based on the corresponding kernels $k_{\boldsymbol{n}'}$ acting at those locations. The observation noise $z[\boldsymbol{n}]$ is assumed to be white Gaussian with variance σ_z^2:

$$z[\boldsymbol{n}] \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_z^2). \tag{2.2}$$

Our goal is to estimate the blur kernel $k_{\boldsymbol{n}}$ at every location $\boldsymbol{n}$. This problem is underdetermined even when the kernel does not change from point to point, and allowing spatial variation adds considerable complexity. Allowing an arbitrary kernel at every pixel renders estimation impractical, and we will need to make some simplifying assumptions to proceed. One assumption we make is that the kernel is constant in local neighborhoods. This turns out to be a reasonable assumption in most cases, but as we shall see, it limits our ability to obtain localized boundaries based on blur information alone.

Note that when the blur is uniform and $k_{\boldsymbol{n}} = k, \forall \boldsymbol{n}$, the model in (2.1) reduces to a regular convolution, *i.e.*

$$y[\boldsymbol{n}] = (x * k)[\boldsymbol{n}] + z[\boldsymbol{n}]. \tag{2.3}$$

This case is naturally analyzed in the frequency domain. The convolution theorem lets us simplify inference with the Fourier transform by *diagonalizing* the action of the constant kernel k as

$$Y(\boldsymbol{\omega}) = X(\boldsymbol{\omega})K(\boldsymbol{\omega}) + Z(\boldsymbol{\omega}), \tag{2.4}$$

where $\boldsymbol{\omega} \in [-\pi, \pi]^2$ corresponds to spatial frequency, and $Y(\boldsymbol{\omega})$, $X(\boldsymbol{\omega})$, $K(\boldsymbol{\omega})$ and $Z(\boldsymbol{\omega})$ are Fourier transforms of $x[\boldsymbol{n}]$, $y[\boldsymbol{n}]$, $k[\boldsymbol{n}]$ and $z[\boldsymbol{n}]$ respectively. However, when the blur kernel varies spatially, the signals $x[\boldsymbol{n}]$ and $y[\boldsymbol{n}]$ are no longer related by convolution, and a global Fourier transform is of limited utility.

Instead, we use a localized frequency representation. Let $w[\boldsymbol{n}] \in \{0, 1\}$ be a symmetric window function with limited spatial support, and $\{f_i\}$ be a set of complex-valued filters defined as

$$f_i[\boldsymbol{n}] = w[\boldsymbol{n}] \times e^{-j\langle \boldsymbol{\omega}_i, \boldsymbol{n} \rangle}, \tag{2.5}$$

with frequencies $\boldsymbol{\omega}_i \in \mathbb{R}^2$. The choice of $\{\boldsymbol{\omega}_i\}_i$ will depend on the window size, and it is made to ensure that the filters $\{f_i\}_i$ are orthogonal, *i.e.* $\langle f_i, f_j^* \rangle = 0$ for $i \neq j$, where f_j^* refers to the complex conjugate of f_j. Applying these filters to an image x yields the corresponding responses $x_i[\boldsymbol{n}] = (x * f_i)[\boldsymbol{n}]$. For every location $\boldsymbol{n}$, the set of coefficients $\{x_i[\boldsymbol{n}]\}_i$ can be interpreted as the Fourier decomposition of a local window centered at that location.

Now, we consider the corresponding local Fourier decomposition of the observed image $y[\boldsymbol{n}]$. As stated earlier, we make the assumption that the kernel is constant in local neighborhoods. As illustrated in Fig. 2.2, we assume that for every location $\boldsymbol{n}$, the same blur kernel $k_{\boldsymbol{n}}$ acts at every scene location in $x[\boldsymbol{n}]$ that contributes to any pixel in a window centered at $\boldsymbol{n}$ in the observed image $y[\boldsymbol{n}]$. This allows us to combine (2.1) and (2.5) and model the filter responses for the observed image as

$$y_i[\boldsymbol{n}] = (x * (k_n * f_i))[\boldsymbol{n}] + (z * f_i)[\boldsymbol{n}]. \tag{2.6}$$

This is similar to the spatially-uniform case (2.4), but different in a critical way. In this spatially-varying case, there is no means of expressing the transform coefficients $\{y_i[\boldsymbol{n}]\}_i$ in terms of the corresponding $\{x_i[\boldsymbol{n}]\}_i$ alone, because pixels in any window of y have values determined by pixels outside of that window in x.

However, we can derive a useful expression for the *statistics* of the observed coefficients, under specific modeling assumptions. Consider the case where there is no observation noise. In

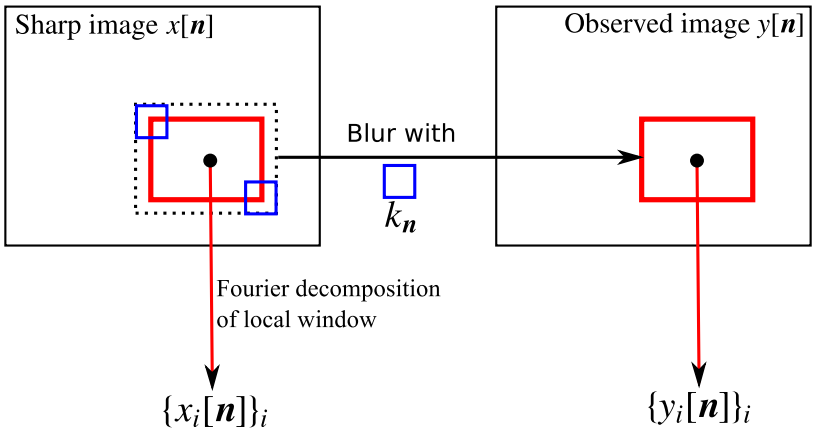


Figure 2.2: Modeling images with spatially-varying blur. We use a localized frequency representation corresponding to Fourier transforms of local image windows (shown with solid red border). For every window, a constant kernel k_n is assumed to act at all locations in the neighborhood (shown with a dotted border) that affect observed pixels inside that window. There is no deterministic relationship between the corresponding observed and sharp coefficients, since observed coefficients depend on the larger neighborhood and not just the Fourier window in the sharp image.

the local neighborhood around $\boldsymbol{n}$ where we assume the kernel to be constant, *i.e.* one equal to the support of w plus that of $k_{\boldsymbol{n}}$, let all elements of x be independent and identically distributed as $\mathcal{N}(0, \sigma^2)$. Since each $y_i[\boldsymbol{n}]$ is a linear combination of $x[\boldsymbol{n}]$, it is also distributed as a zero-mean Gaussian, with variance given by

$$\begin{aligned} \mathbb{E}\,|y_i[\boldsymbol{n}]|^2 &= \mathbb{E}\left|\sum_{\boldsymbol{l}} x[\boldsymbol{n}-\boldsymbol{l}](f_i * k_{\boldsymbol{n}})[\boldsymbol{l}]\right|^2 \\ &= \sigma^2 \sum_{\boldsymbol{l}} |(f_i * k_{\boldsymbol{n}})[\boldsymbol{l}]|^2, \end{aligned} \tag{2.7}$$

since the different $x[\boldsymbol{n}]$ are un-correlated. This can be thought of as an analogue to expression in (2.4), and accordingly we refer to

$$\left\{\sigma_{ki}^2\right\}_i = \left\{\sum_{\boldsymbol{l}} |(k * f_i)[\boldsymbol{l}]|^2\right\}_i, \tag{2.8}$$

as the *blur spectrum* in the remainder of this chapter.

It is worth noting here that while the transform coefficients $\{x_i[\boldsymbol{n}]\}_i$ are un-correlated owing to the filters $\{f_i\}_i$ being orthogonal, strictly speaking the corresponding observed coefficients

$\{y_i[\boldsymbol{n}]\}_i$ are not. Specifically for $i \neq j$, we have

$$\begin{aligned}
&\mathbb{E}\, y_i[\boldsymbol{n}] y_j^*[\boldsymbol{n}] = \sigma^2 \sum_{\boldsymbol{l}} (f_i * k_{\boldsymbol{n}})[\boldsymbol{l}] (f_j * k_{\boldsymbol{n}})^*[\boldsymbol{l}] \\
&= \sigma^2 \sum_{\boldsymbol{l}} \sum_{\boldsymbol{m}_1} \sum_{\boldsymbol{m}_2} k_{\boldsymbol{n}}[\boldsymbol{m}_1] k_{\boldsymbol{n}}[\boldsymbol{m}_2] f_i[\boldsymbol{l} - \boldsymbol{m}_1] f_j^*[\boldsymbol{l} - \boldsymbol{m}_2] \\
&= \sigma^2 \sum_{\boldsymbol{m}_1} \sum_{\boldsymbol{m}_2} k_{\boldsymbol{n}}[\boldsymbol{m}_1] k_{\boldsymbol{n}}[\boldsymbol{m}_2] \left(\sum_{\boldsymbol{l}} f_i[\boldsymbol{l}] f_j^*[\boldsymbol{l} - \Delta\boldsymbol{m}] \right), \qquad (2.9)
\end{aligned}$$

where $\Delta\boldsymbol{m} = \boldsymbol{m}_2 - \boldsymbol{m}_1$. For all terms in the summation above where $\Delta\boldsymbol{m} = 0$, the expression in parantheses is just the inner product between the filters f_i and f_j which evaluates to zero. For the case of $\Delta\boldsymbol{m} \neq 0$, we have the following expression:

$$\begin{aligned}
\sum_{\boldsymbol{l}} f_i[\boldsymbol{l}] f_j^*[\boldsymbol{l} - \Delta\boldsymbol{m}] &= \sum_{\boldsymbol{l}} w[\boldsymbol{l}] e^{-j\langle \boldsymbol{\omega}_i, \boldsymbol{l} \rangle}\, w[\boldsymbol{l} - \Delta\boldsymbol{m}] e^{j\langle \boldsymbol{\omega}_j, \boldsymbol{l} - \Delta\boldsymbol{m} \rangle} \\
&= e^{-j\langle \boldsymbol{\omega}_j, \Delta\boldsymbol{m} \rangle} \sum_{\boldsymbol{l}} w[\boldsymbol{l} - \Delta\boldsymbol{m}]\; f_i[\boldsymbol{l}] f_j^*[\boldsymbol{l}], \qquad (2.10)
\end{aligned}$$

using the definition of f_i from (2.5). Therefore, the expressions for terms with $\Delta\boldsymbol{m} \neq 0$ differ in two aspects— there is a *phase-shift* in the complex exponential which simply shows up as a unit-magnitude constant outside the summation; and the summation itself is truncated because of the shift in the window function $w[\boldsymbol{l} - \Delta\boldsymbol{m}]$. Whereas the full summation corresponds to the inner product $\langle f_i, f_j^* \rangle$ and would have evaluated to zero, the truncation causes the above expression to yield a small residual value. In practice, we find the cross-correlation values between the coefficients due to this residual to be negligible, and therefore in the following sections, we treat the observed coefficients as uncorrelated to simplify inference.

2.4 Inference with Image Prior

We now describe a statistical inference approach for estimating the blur kernels $k_{\boldsymbol{n}}$ from the observed image $y[\boldsymbol{n}]$, using an appropriate prior model for the latent sharp image $x[\boldsymbol{n}]$. We define this prior by modeling the distributions of image gradients. Formally, let $x^{\nabla}[\boldsymbol{n}] = (\nabla * x)[\boldsymbol{n}]$

correspond to the gradient map of x for a gradient filter ∇. We will consider different models for $x^{\nabla}[\boldsymbol{n}]$, and use them for inference on the gradient map $y^{\nabla}[\boldsymbol{n}]$ of the observed image $y[\boldsymbol{n}]$.

2.4.1 Gaussian Distribution

We first look at using a Gaussian distribution to model the gradient map $x^{\nabla}[\boldsymbol{n}]$. Specifically, all gradient values in the image are assumed to be independent and identically distributed with zero mean and a fixed variance $s > 0$, *i.e.*

$$x^{\nabla}[\boldsymbol{n}] \overset{\text{iid}}{\sim} \mathcal{N}(0, s). \tag{2.11}$$

Now consider the local Fourier coefficients $\{y_i^{\nabla}[\boldsymbol{n}]\}$ of the observed gradient map $y^{\nabla}[\boldsymbol{n}]$. In the absence of observation noise, these are given by

$$y_i^{\nabla}[\boldsymbol{n}] = (y^{\nabla} * f_i)[\boldsymbol{n}] = (\nabla * y * f_i)[\boldsymbol{n}] = (\nabla * x * k_{\boldsymbol{n}} * f_i)[\boldsymbol{n}] = (x^{\nabla} * (k_{\boldsymbol{n}} * f_i))[\boldsymbol{n}]. \tag{2.12}$$

We can now use the identity in (2.7) to derive an expression for the likelihood $p(k_{\boldsymbol{n}} = k|y^{\nabla}[\boldsymbol{n}])$, of a candidate kernel k acting on a window centered at location $\boldsymbol{n}$, as

$$\begin{aligned} p(k_{\boldsymbol{n}} = k|y^{\nabla}[\boldsymbol{n}]) \quad &\propto \quad p(\{y_i^{\nabla}[\boldsymbol{n}]\}_i|k_{\boldsymbol{n}} = k) \propto \prod_i \mathcal{N}\left(y_i^{\nabla}[\boldsymbol{n}]|0, s\sigma_{ki}^2\right) \\ &\propto \quad \left(\prod_i \sigma_{ki}^2\right)^{-1/2} \exp\left(-\frac{1}{2s}\sum_i \frac{|y_i^{\nabla}[\boldsymbol{n}]|^2}{\sigma_{ki}^2}\right), \end{aligned} \tag{2.13}$$

where σ_{ki}^2 is the blur spectrum of the candidate kernel k, as per (2.8).

A Gaussian prior has the virtue of simplicity, and the likelihood in (2.13) can be computed easily in closed form. Moreover, despite the fact that empirical distributions of gradients in natural images have significantly heavier tails than a Gaussian distribution, the model has been used with some amount of success for blur estimation in the absence of spatial variation [21]. Unfortunately, it is far less useful when dealing with spatially-varying blur.

This is illustrated in Fig. 2.3 using a toy one-dimensional "image". We consider two local windows, one that contains a low-contrast edge that is not blurred, and another with a high-contrast

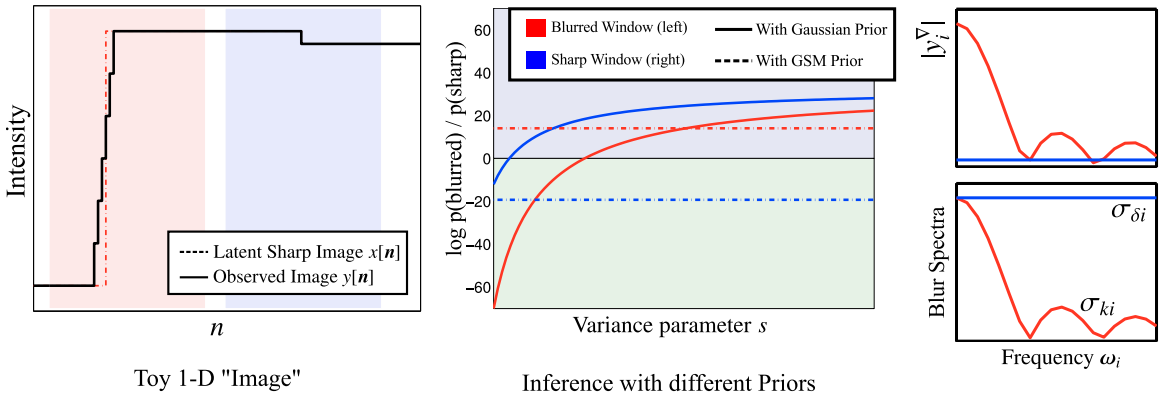


Figure 2.3: Inference with different priors on a toy example. We consider two windows in a 1-D image (Left), containing a high contrast blurred edge (the dotted line corresponds to the latent sharp edge) and a low contrast blurred edge. (Center) For each window, we show the ratio between the likelihoods of it being blurred and sharp, under the Gaussian and GSM priors. For the Gaussian prior, this ratio is shown as a function of the variance parameter s. A window is classified as blurred if the log of the ratio is above 0. While the GSM-based likelihood yields the correct classification for both edges, there is no single value of s for which both windows are classified correctly. (Right) A comparison of the blur spectra (bottom) corresponding to the blur kernel σ_{ki} and to no blur $\sigma_{\delta i}$, with the actual observed coefficient magnitudes (top) for the two windows containing the blurred and sharp edges. We see that while the observed magnitudes match the "shapes" of their corresponding expected spectra, they are scaled differently based on the contrast of the edges.

edge that is blurred by a box filter k. Our task is to decide, by looking at the observed coefficients $\{y_i^\nabla[\boldsymbol{n}]\}_i$ in each local window, whether that window was blurred by k (which is assumed to be known), or was not blurred at all (*i.e.* it was blurred by an impulse kernel δ). We shall make this decision based on the which kernel's likelihood, as defined in (2.13), is greater.

Figure 2.3 (center) shows the ratio of these likelihoods for different values of the sharp image model variance parameter s, and we see that the Gaussian-based likelihood measure in (2.13) is never able to classify *both* windows correctly. That is, we can choose the model parameter to correctly classify one or the other, but not both simultaneously. Intuitively, this results from the fact that even if one can reasonably expect the mean square gradient values of an entire sharp image to be close to s, the same is not true within different small windows of that sharp image, whose statistics can change quite dramatically.

To gain further insight, we can look at the local spectra of the input image around the

two edges, as depicted in Fig. 2.3 (right). We see that while the magnitudes of the spectra match the *shapes* of the two blur kernel spectra, their relative scales are very different. The classification fails, then, because the simple Gaussian-based likelihood model (2.13) involves *absolute* variance values, at a scale that is fixed and determined by the choice of our image prior model parameter s. Viewed another way, the likelihood term is "distracted" by the difference in contrasts between the two edges, and this prevents it from being able to make a decision based purely on how sharp the edges are.

2.4.2 Gaussian Scale Mixture

In this section, we introduce an image prior model that specifically seeks to capture edge sharpness distinct from edge contrast. Instead of fixing the variance parameter s for the entire image, we let each local neighborhood have its own variance, and treat that variance as a random variable instead of a model parameter. For every local neighborhood η, we model the gradient values in that neighborhood as

$$p\left(\{x^{\nabla}[\boldsymbol{n}]\}_{\boldsymbol{n}\in\eta}\right) = \int p_s(s) \prod_{\boldsymbol{n}\in\eta} \mathcal{N}\left(x^{\nabla}[\boldsymbol{n}]|0, s\right) ds, \tag{2.14}$$

where $p_s(s)$ is a probability distribution on s. This corresponds to a Gaussian Scale Mixture (GSM) model [43], where s corresponds to a common *scale* parameter for the neighborhood η. Conditioned on this common scale s, the image gradients in any neighborhood η are independent and identically distributed. Note that the marginal distribution of individual gradients in $x^{\nabla}[\boldsymbol{n}]$ is more kurtotic than a Gaussian distribution. For this reason, GSM-based priors have been used previously to model natural images, for applications in denoising [13] and deconvolution [44].

We want the distribution $p_s(s)$ to be such that the overall prior model in (2.14) is agnostic of edge contrast. Therefore, we use the Jeffreys non-informative prior [45] for Gaussian variance parameters to model s as

$$p_s(s) \propto s^{-1}, \quad \text{for } s > 0. \tag{2.15}$$

Note that this is an *improper* prior distribution, because it is not integrable. Nevertheless, as we show next, it yields a well-defined likelihood measure, and one that is not biased by the scale of the observed gradients.

Setting η to be the neighborhoods in $x^{\nabla}[\boldsymbol{n}]$ on which each set of the observed local Fourier transform coefficients $\{y_i^{\nabla}[\boldsymbol{n}]\}_i$ depend (*i.e.* equal to the sum of the supports of the window function w and candidate blur kernel k), we can derive the likelihood $p(k_{\boldsymbol{n}} = k|y^{\nabla}[\boldsymbol{n}])$ under the GSM-based prior model as

$$\begin{aligned} p\left(k_{\boldsymbol{n}} = k \left| y^{\nabla}[\boldsymbol{n}] \right.\right) &\propto \int \frac{1}{s} \prod_i \mathcal{N}\left(y_i^{\nabla}[\boldsymbol{n}]|0, s\sigma_{ki}^2\right) ds \\ &\propto \int s^{-(F/2+1)} \left(\prod_i \sigma_{ki}^2\right)^{-1/2} \exp\left(-\frac{1}{2s}\sum_i \frac{|y_i^{\nabla}[\boldsymbol{n}]|^2}{\sigma_{ki}^2}\right) ds, \end{aligned} \tag{2.16}$$

where F is the total number of frequency bands. Fortunately, this expression also has a closed form solution, that can be derived by matching the integrand above to the form of the inverse-Gamma distribution:

$$\gamma^{-1}(s|\alpha,\beta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} s^{-(\alpha+1)} e^{-\beta/s}. \tag{2.17}$$

Using the fact that integrating the above distribution over s yields one, we have

$$p\left(k_{\boldsymbol{n}} = k \left| y^{\nabla}[\boldsymbol{n}] \right.\right) \propto \left(\prod_i \sigma_{ki}^2\right)^{-1/2} \left[\sum_i \frac{|y_i^{\nabla}[\boldsymbol{n}]|^2}{\sigma_{ki}^2}\right]^{-F/2}. \tag{2.18}$$

We can show that the likelihood measure in (2.18) has several desirable properties. Note that it can be re-written as:

$$p\left(k_{\boldsymbol{n}} = k \left| y^{\nabla}[\boldsymbol{n}] \right.\right) \propto \left(\prod_i |y_i^{\nabla}[\boldsymbol{n}]|\right)^{-1} \left[\frac{\mathrm{GM}(\{r_{ki}[\boldsymbol{n}]\}_i)}{\mathrm{AM}(\{r_{ki}[\boldsymbol{n}]\}_i)}\right]^{F/2}, \tag{2.19}$$

where,

$$r_{ki}[\boldsymbol{n}] = \frac{|y_i^{\nabla}[\boldsymbol{n}]|^2}{\sigma_{ki}^2}, \tag{2.20}$$

are the ratios between the observed and blur spectra, and the GM$(\cdot)$ and AM$(\cdot)$ terms refer to their geometric and arithmetic means, respectively. Scaling all the r_{ki} terms for a particular window by a

constant will not affect the ratio between the geometric and arithmetic means. Therefore, we have the following identity:

$$\frac{p\left(k_{\boldsymbol{n}} = \beta_1 k_1 \left| \alpha y^{\nabla}[\boldsymbol{n}]\right.\right)}{p\left(k_{\boldsymbol{n}} = \beta_2 k_2 \left| \alpha y^{\nabla}[\boldsymbol{n}]\right.\right)} = \frac{p\left(k_{\boldsymbol{n}} = k_1 \left| y^{\nabla}[\boldsymbol{n}]\right.\right)}{p\left(k_{\boldsymbol{n}} = k_2 \left| y^{\nabla}[\boldsymbol{n}]\right.\right)}, \tag{2.21}$$

for any scalars α, β_1 and β_2. This implies that for a candidate window, the likelihood ratio between two candidate kernels is independent of the *scale* or absolute magnitudes of the kernels and observed gradients.

Furthermore, we note that the geometric mean of a series is always less than or equal to the arithmetic mean, with the two being equal only when all elements of the series, in this case the ratios $\{r_{ki}\}_i$, are the same. Therefore, for a given set of observed coefficients $\{y_i^{\nabla}[\boldsymbol{n}]\}_i$, the likelihood is maximal when $|y_i^{\nabla}[\boldsymbol{n}]|^2 = \alpha\ \sigma_{ki}^2,\ \forall i$, *i.e.* the observed spectrum is exactly a scaled version of the candidate blur spectrum. Because of these properties, the likelihood measure in (2.18) can be interpreted as matching the *shapes* of the observed spectrum to those of candidate blur kernels, *independent* of their magnitudes. As a result, we see that the GSM-based likelihood is able to ignore the contrasts of the two edges in our toy 1-D example and, as shown in Fig. 2.3 (center), is able to classify both correctly.

2.4.3 Inference with Noisy Observations

So far, we have looked at computing blur likelihoods ignoring the effects of observation noise, *i.e.* $z[\boldsymbol{n}]$ in (2.1). We now consider the effect of noise on the statistics of the observed gradient coefficients $\{y_i^{\nabla}[\boldsymbol{n}]\}$. In the presence of noise, these coefficients are given by

$$y_i^{\nabla}[\boldsymbol{n}] = \tilde{y}_i^{\nabla}[\boldsymbol{n}] + z_i^{\nabla}[\boldsymbol{n}], \tag{2.22}$$

where $\{\tilde{y}_i^{\nabla}[\boldsymbol{n}]\}_i$ are the corresponding *clean* coefficients that would have been observed in the absence of noise, and $\{z_i^{\nabla}[\boldsymbol{n}]\}_i$ are the local Fourier coefficients of the gradient map of the noise image $z[\boldsymbol{n}]$, given by

$$z_i^{\nabla}[\boldsymbol{n}] = (\nabla * z * f_i)[\boldsymbol{n}]. \tag{2.23}$$

Given the noise model in (2.2), where $z[\boldsymbol{n}]$ is assumed to be independent and identically distributed as a Gaussian with variance σ_z^2 *in the pixel domain*, it follows from the identity in (2.7) that the expected variance of the noise gradient coefficients $\{z_i^{\nabla}[\boldsymbol{n}]\}$ will be given by the spectrum $\sigma_{\nabla i}^2$ of the gradient filter ∇ computed as per (2.8), *i.e.*

$$\mathbb{E}\ \left|z_i^{\nabla}[\boldsymbol{n}]\right|^2 = \sigma_z^2\sigma_{\nabla i}^2 = \sigma_z^2 \sum_{\boldsymbol{n}} |(\nabla * f_i)[\boldsymbol{n}]|^2 \,. \tag{2.24}$$

Unfortunately, updating the GSM-based prior likelihood to exactly model the effect of these noise coefficients as

$$p\left(k_{\boldsymbol{n}} = k \left| y^{\nabla}[\boldsymbol{n}]\right.\right) \propto \int p_s(s) \prod_i \mathcal{N}\left(y_i^{\nabla}[\boldsymbol{n}]|0, s\sigma_{ki}^2 + \sigma_z^2\sigma_{\nabla i}^2\right)\, ds, \tag{2.25}$$

is problematic. Using the Jeffreys prior from (2.15) no longer yields a finite likelihood measure, because the integrand above goes to infinity as s approaches zero. Even if $p_s(s)$ were to be suitably modified, there would be no closed form solution and the integration above would have to be done numerically.

Instead, we take the following approach to approximately account for the effect of observation noise in our original likelihood measure in (2.18): for every window, we only consider those observed coefficients whose magnitudes are significantly above the corresponding noise variances from (2.24), and then compute the likelihood from only these coefficients, treating them as noise-free. This updated likelihood measure is given by

$$p\left(k_{\boldsymbol{n}} = k \left| y^{\nabla}[\boldsymbol{n}]\right.\right) \propto \left(\prod_{i\in\mathcal{I}[\boldsymbol{n}]} \sigma_{ki}^2\right)^{-1/2} \left[\sum_{i\in\mathcal{I}[\boldsymbol{n}]} \frac{|y_i^{\nabla}[\boldsymbol{n}]|^2}{\sigma_{ki}^2}\right]^{-|\mathcal{I}[\boldsymbol{n}]|/2}, \tag{2.26}$$

where $\mathcal{I}[\boldsymbol{n}]$ corresponds to the *active* set of coefficients for each window, and is given by

$$\mathcal{I}[\boldsymbol{n}] = \{\, i \ :\ |y_i^{\nabla}[\boldsymbol{n}]|^2 > T\ \sigma_z^2\sigma_{\nabla i}^2\}, \tag{2.27}$$

for some scalar T. This choice of $\mathcal{I}[\boldsymbol{n}]$ is independent of the candidate kernel k, and the likelihood measures for different kernels is computed over the same set of coefficients for each window.

Finally, note that gradient filters are typically *high-pass* and therefore the values of $\sigma^2_{\nabla i}$ corresponding to the higher frequencies $\boldsymbol{\omega}_i$ will be large. As a result, the observed coefficients for these bands will often fail to cross the noise threshold as defined in (2.27). Therefore, to reduce computational cost, we avoid computing coefficients for bands with high values of $\sigma^2_{\nabla i}$ and apply only a subset of the filters $\{f_i\}_i$ to the observed image.

2.5 Detecting Motion

We now address the task of detecting and segmenting out a moving object from a single still image. We describe an algorithm that exploits motion blur as a cue for this task, using the likelihood measure described in the previous section. We assume that there is only one moving object in the image, and instead of detecting a separate blur kernel for each region in this object, we assume that the object is moving uniformly and that all regions inside the object are therefore affected by the same kernel k_m. This assumption makes the segmentation more robust, and our estimate of the kernel k_m provides direct information about the orientation and speed (relative to exposure time) of the moving object [32–35].

Formally, for a given observed image $y[\boldsymbol{n}]$, every pixel is modeled as either part of the stationary sharp "background" or the motion-blurred "foreground". Therefore, we assume that every kernel $k_{\boldsymbol{n}}$ belongs to the set $\{k_0, k_m\}$, where k_0 corresponds to the blur kernel acting on the stationary regions of the image and is chosen to either be an impulse (*i.e.* no blur at all) or a mild defocus blur; and k_m is the motion blur kernel acting on the moving object. Our task is to estimate the motion kernel k_m, and to assign a label $M[\boldsymbol{n}]$ to every location, where $M[\boldsymbol{n}] = 0$ for pixels in stationary regions, and 1 for pixels in the moving object. We shall interpret this to imply that $k_{\boldsymbol{n}} = k_0$ for $M[\boldsymbol{n}] = 0$ and $k_{\boldsymbol{n}} = k_m$ otherwise, but our assumptions about the blur kernel $k_{\boldsymbol{n}}$ being constant in a neighborhood clearly do not hold at the borders of this segmentation. Therefore, we will augment the blur likelihood with a per-pixel color model that will help with inference near these boundaries.

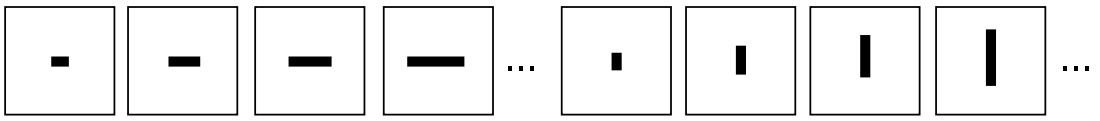


Figure 2.4: Candidate set of motion kernels. The kernel k_m acting on the moving object is assumed to belong to a set of horizontal and vertical box filters of different length. These kernels correspond to the object moving with different uniform velocities in the corresponding direction during exposure.

2.5.1 Selecting the Kernel

We first address the task of choosing the blur kernel k_m. It is assumed to be one of a discrete set of possible candidates, corresponding to horizontal or vertical box filters of different lengths (see Fig. 2.4). These correspond roughly to horizontal or vertical object motion with fixed velocity. Formally, for a chosen set of lengths $\{l_1, l_2, \ldots l_L\}$, we assume that

$$k_m \in \{b_{hl_1}, \ldots b_{hl_L}, b_{vl_1}, \ldots b_{vl_L}\}, \tag{2.28}$$

where b_{hl} is a horizontal "box" filter of length l (corresponding to number of pixels the object moved during exposure), *i.e.*

$$b_{hl}[\boldsymbol{n}] = \begin{cases} 1 & \text{if } n_y = 0, 0 \leq n_x < l, \\ 0 & \text{otherwise,} \end{cases} \tag{2.29}$$

and b_{vl} is a similarly defined vertical box filter.

To handle both horizontal and vertical candidate blurs $\{b_{hl}\}$ and $\{b_{vl}\}$, we need to use two sets of coefficients $\{y_i^h[\boldsymbol{n}]\}_i$ and $\{y_i^v[\boldsymbol{n}]\}_i$ defined as,

$$y_i^h[\boldsymbol{n}] = (f_{ih} * \nabla_h * y)[\boldsymbol{n}], \quad y_i^v[\boldsymbol{n}] = (f_{iv} * \nabla_v * y)[\boldsymbol{n}], \tag{2.30}$$

where ∇_h and ∇_v are horizontal and vertical gradient filters, and $\{f_{ih}\}_i$ and $\{f_{iv}\}_i$ correspond to local Fourier filters defined as per (2.5), using horizontal and vertical one-dimensional windows.

To select the motion blur kernel k_m, we need to derive an expression for the global likelihood $p(k_m = b)$. If b is horizontal, then the coefficients $y_i^h[\boldsymbol{n}]$ at all $\boldsymbol{n}$ should be explained either by

b or k_0, and the coefficients in the orthogonal direction $y_i^v[\boldsymbol{n}]$ should all be explained by k_0 (*i.e.* they are not affected by the motion blur). The converse is true if b is vertical, and this reasoning leads us to define

$$\begin{aligned} p(k_m = b_{lh}) &\propto \left[\prod_{\boldsymbol{n}} \max_{k\in\{k_0,b_{lh}\}} p\left(k_{\boldsymbol{n}} = k \left| \left\{y_i^h[\boldsymbol{n}]\right\}_i\right.\right)\right] \times \left[\prod_{\boldsymbol{n}} p\left(k_{\boldsymbol{n}} = k_0 \left| \left\{y_i^v[\boldsymbol{n}]\right\}_i\right.\right)\right], \\ p(k_m = b_{lv}) &\propto \left[\prod_{\boldsymbol{n}} p\left(k_{\boldsymbol{n}} = k_0 \left| \left\{y_i^h[\boldsymbol{n}]\right\}_i\right.\right)\right] \times \left[\prod_{\boldsymbol{n}} \max_{k\in\{k_0,b_{lv}\}} p\left(k_{\boldsymbol{n}} = k \left| \left\{y_i^v[\boldsymbol{n}]\right\}_i\right.\right)\right], \end{aligned} \quad (2.31)$$

where the individual per-window likelihood measures are computed as per (2.26). The blur kernel k_m can then be chosen amongst the candidate set in (2.28) to maximize this likelihood $p(k_m)$. Note that the most expensive part of computing the likelihoods, for each of the candidate kernels, is the local Fourier decomposition, but this only needs to be done once for each orientation. Once the coefficients $\{y_i^h[\boldsymbol{n}]\}_i$ and $\{y_i^v[\boldsymbol{n}]\}_i$ have been computed, the kernel k_m can be selected very efficiently using the above strategy.

2.5.2 Segmentation

Having chosen the motion kernel k_m, we now address the problem of assigning the segmentation labels $M[\boldsymbol{n}]$ at every location $\boldsymbol{n}$. As mentioned above, the blur likelihoods alone are not sufficient for this task since they are not well-defined at segmentation boundaries, and also in smooth regions where the most of the observed gradient coefficients are below the noise threshold in (2.27). Therefore, these likelihoods are combined with a complimentary cue derived from statistical distributions for object and background pixel colors into a Markov random field (MRF) model, as used in traditional segmentation [46].

This MRF model will be defined in terms of an energy function that incorporates the blur likelihoods, color distributions, and a spatial smoothness constraint on $M[\boldsymbol{n}]$ that favors adjacent pixels having the same label assignments. The blur component $B_{\boldsymbol{n}}[m]$ of this energy, as a function of the label $m \in \{0, 1\}$ at that location, is defined simply as the negative log-likelihood of the

corresponding kernel (*i.e.* k_m or k_0), *i.e.*

$$B_{\boldsymbol{n}}(m) = -m \ \log p\left(k_{\boldsymbol{n}} = k_m \left| y^{\nabla}[\boldsymbol{n}]\right.\right) - (1-m) \log p\left(k_{\boldsymbol{n}} = k_0 \left| y^{\nabla}[\boldsymbol{n}]\right.\right). \tag{2.32}$$

Next, we define color models for pixels corresponding to the moving object and stationary background. While the blur model was based on the linear greyscale version of the observed image, these color models are defined on the gamma-corrected RGB vectors $\boldsymbol{y}^c[\boldsymbol{n}] \in \mathbb{R}^3$. We use a finite mixture of multivariate Gaussians to model both background and object pixels, where the object color model is defined as

$$p\left(\boldsymbol{y}^c[\boldsymbol{n}] \left| \boldsymbol{\theta}_m = \left\{\gamma_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right\}_{j=1}^{J_m}\right.\right) = \sum_{j=1}^{J_m} \gamma_j \mathcal{N}\left(\boldsymbol{y}^c[\boldsymbol{n}] \left| \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right.\right). \tag{2.33}$$

Here, $\boldsymbol{\theta}_m$ correspond to the parameters of this model that also need to be estimated. The background color model is defined in a similar way in terms of the corresponding parameters $\boldsymbol{\theta}_0 = \left\{\gamma_j, \boldsymbol{\mu_j}, \boldsymbol{\Sigma}_j\right\}_{j=1}^{J_0}$. The color component $C_{\boldsymbol{n}}(m)$ of the MRF energy is then defined as

$$C_{\boldsymbol{n}}(m) = -m \log p\left(\boldsymbol{y}^c[\boldsymbol{n}] \left| \boldsymbol{\theta}_m\right.\right) - (1-m) \log p\left(\boldsymbol{y}^c[\boldsymbol{n}] \left| \boldsymbol{\theta}_0\right.\right). \tag{2.34}$$

The labels $M[\boldsymbol{n}] \in \{0, 1\}$ are chosen as the solution to the following energy minimization problem:

$$M[\boldsymbol{n}] = \arg\min_{M} \left(\min_{\boldsymbol{\theta}_m, \boldsymbol{\theta}_0} E\left(M[\boldsymbol{n}], \boldsymbol{\theta}_m, \boldsymbol{\theta}_0\right)\right), \tag{2.35}$$

where the energy function $E(\cdot)$ is defined as

$$E\left(M[\boldsymbol{n}], \boldsymbol{\theta}_m, \boldsymbol{\theta}_0\right) = \sum_{\boldsymbol{n}} B_{\boldsymbol{n}}\left(M[\boldsymbol{n}]\right) + \lambda \sum_{\boldsymbol{n}} C_{\boldsymbol{n}}\left(M[\boldsymbol{n}]\right) + \frac{\rho}{2} \sum_{(\boldsymbol{n}, \boldsymbol{n}') \in \mathcal{P}} \left|M[\boldsymbol{n}] - M[\boldsymbol{n}']\right|. \tag{2.36}$$

The first two terms in this energy function favor label assignments that match the likelihood measures under the blur and color models, with the scalar parameter λ controlling the relative contribution of the two. The third term enforces a smoothness constraint amongst all pairs $\mathcal{P}$ of neighboring locations, by adding a penalty of ρ every time a pair is assigned different labels.

Note that the minimization in (2.35) has to be done over both the labels $M[\boldsymbol{n}]$ as well as the color model parameters $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_0$. For a fixed value of the color parameters, the optimal assignments for $M[\boldsymbol{n}]$ can be computed exactly using graph cuts [47] to best satisfy the smoothness constraints. Therefore, we propose the following iterative algorithm to carry out the overall minimization: we initialize the iterations by setting the label assignments $M[\boldsymbol{n}]$ using graph cuts, based only on the blur and smoothness terms in (2.36) which do not involve the color parameters. At each subsequent iteration, we set the color parameters $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_0$ to minimize the energy function keeping the current estimates of the label assignments $M[\boldsymbol{n}]$ fixed. This corresponds to fitting the color parameters so as to maximize the log-likelihoods of the sets of object and foreground labeled pixels under their respective distributions (as defined in (2.33)), and we do this using the Expectation Maximization (EM) algorithm [48]. The labels $M[\boldsymbol{n}]$ are then recomputed using these values of the color model parameters. Note that both these steps reduce the value of the energy function in (2.36), and we iterate till the label assignments $M[\boldsymbol{n}]$ converge.

2.6 Experimental Results

We now evaluate the algorithm on a database of images of real-world scenes, captured using three consumer cameras: a digital SLR, a point-and-shoot and a cell-phone. Each image contains a single motion-blurred object, and we evaluating the proposed method's ability to latch on to the motion blur present in the image, which includes choosing the correct approximate orientation and length for the blur kernel, to yield a useful segmentation. This image database and a MATLAB implementation of the algorithm are available for download [22].

2.6.1 Implementation Details

Both the horizontal and vertical sub-band transforms were defined in terms of windows of length $W = 61$. The filters $\{f_{ih}\}$ and $\{f_{iv}\}$ were then generated with $\boldsymbol{\omega}_i$ equal to the corresponding

set of frequencies $2\pi u/W$, $u \in \{1, \ldots 15\}$, *i.e.* we only used the lower half set of frequencies which had acceptable levels of noise variance in the gradient domain. The gradient filters ∇ were defined simply as $[-1, \quad 1]$ (in the appropriate direction), *i.e.* the gradient values corresponded to the finite differences between neighboring pixels.

For the blur length, we searched over the discrete set of integers from 4 to 15 pixels. Larger lengths can be handled by downsampling the image to an appropriate resolution. For segmentation, the object and foreground RGB color vectors were modeled as mixtures of 4 and 6 Gaussians respectively. The choices for all parameters (including σ_z^2, γ and ρ) were kept fixed for all images.

The JPEG images generated by the cameras were gamma-corrected, and therefore the input image had to be linearized for use with the blur model. Since the exact tone maps applied by the cameras were unknown, we assumed standard sRGB gamma-correction and computed the linear images by raising all intensities to the power $(1/2.2)$. Note that this is only an approximation, and better results are likely when using a calibrated camera.

2.6.2 Results

Figure 2.5 shows the full segmentation results for all images in the database. This includes a visualization of the blur and color components of the energy $B_{\boldsymbol{n}}$ and $C_{\boldsymbol{n}}$, along with the corresponding detected motion blur kernel k_m, and estimated color model parameters $\boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_0$. We then display the combination of these two energy components, and the final labels $M[\boldsymbol{n}]$ computed using graph cuts.

We find that the algorithm does well on this diverse set of real world images, despite approximate assumptions about noise variance, gamma correction, uniform object motion, *etc.* It is also able to detect the blurred regions even in cases of relatively minor motion (such as for images 6 and 11), and is fairly robust even in cluttered backgrounds (such as image 3). The detected kernel k_m are also found to approximately correlate to the direction and speed of apparent motion of these

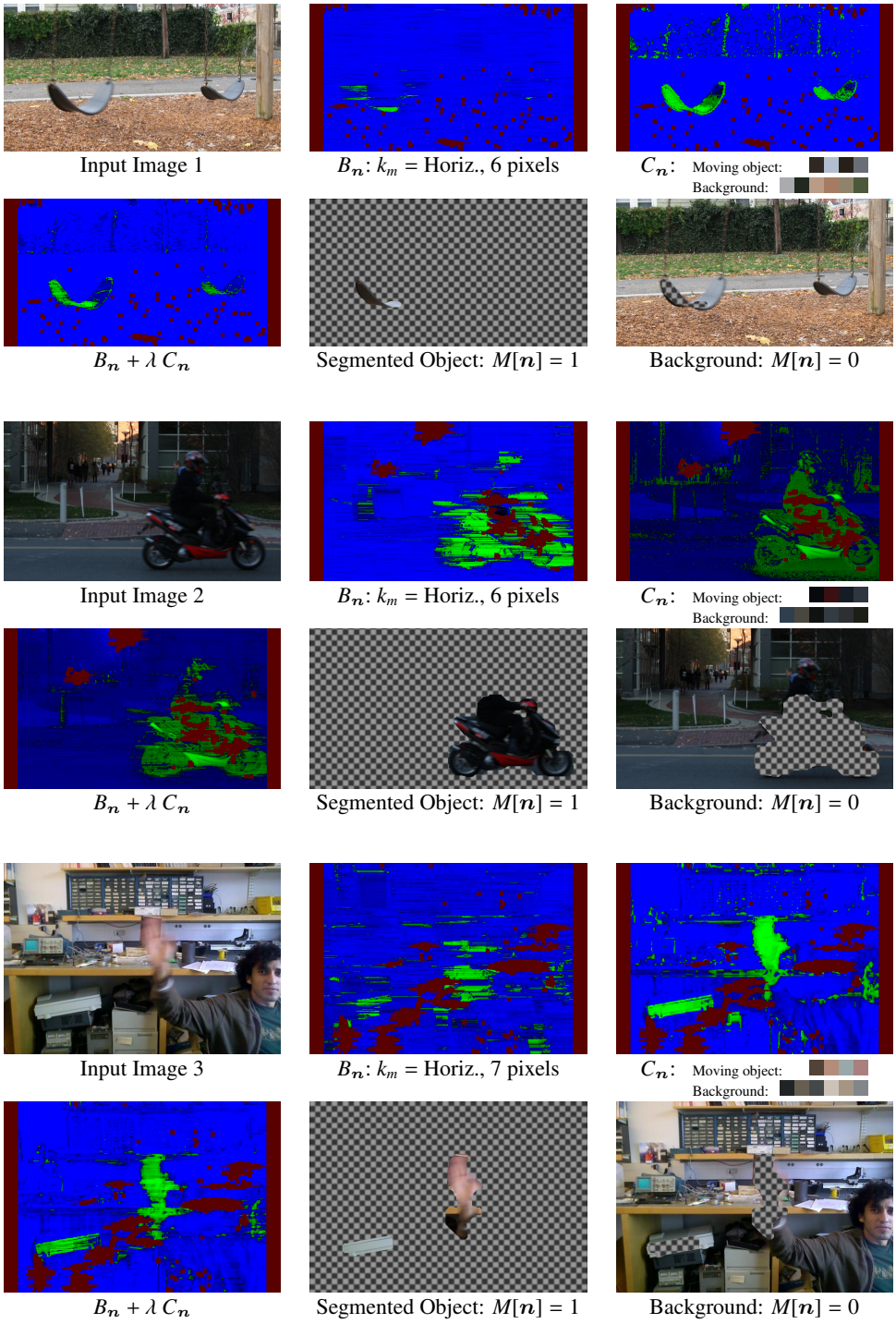
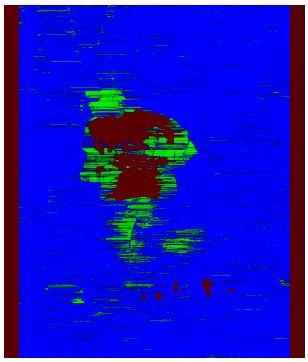
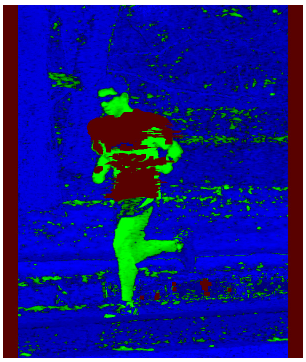


Figure 2.5: Segmentation results on various image. For each input image, we show the blur and color energy components B_n and C_n, where the intensity corresponds to the absolute difference between the energies for the two labels, with blue regions indicating that the $m = 0$ label (*i.e.* for the stationary background) is preferred, and green $m = 1$. Red corresponds to invalid or unavailable data. Then, we show the combined energy, and the final segmentation $M[n]$. The chosen motion kernel k_m, and estimated mean color parameters $\{\mu_j\}_j$ for the color models are also indicated.

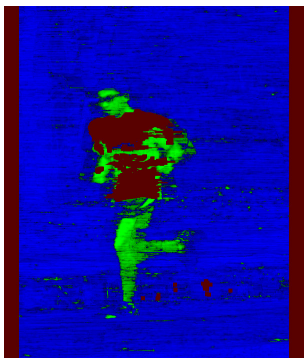
Input Image 4



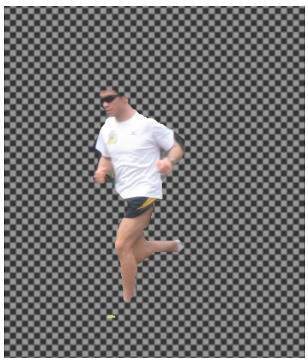
B_n: k_m = Horiz., 4 pixels



C_n: Moving object: Background:



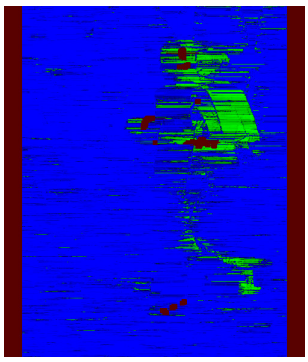
$B_n + \lambda\, C_n$



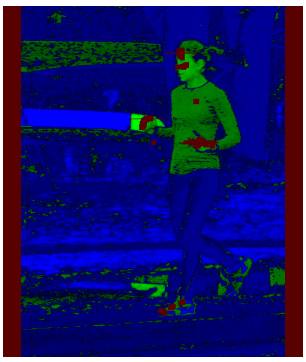
Segmented Object: $M[n] = 1$

Background: $M[n] = 0$

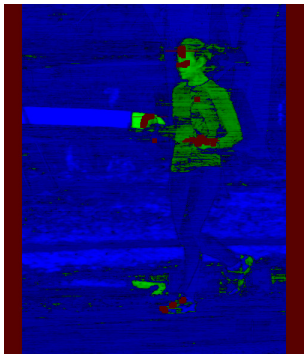
Input Image 5



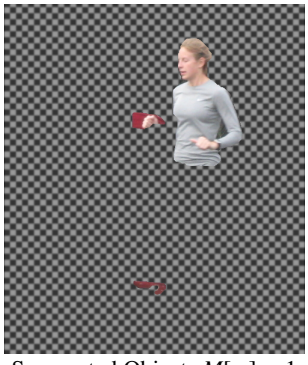
B_n: k_m = Horiz., 4 pixels



C_n: Moving object: Background:



$B_n + \lambda\, C_n$



Segmented Object: $M[n] = 1$

Background: $M[n] = 0$

Figure 2.5: *(continued)*

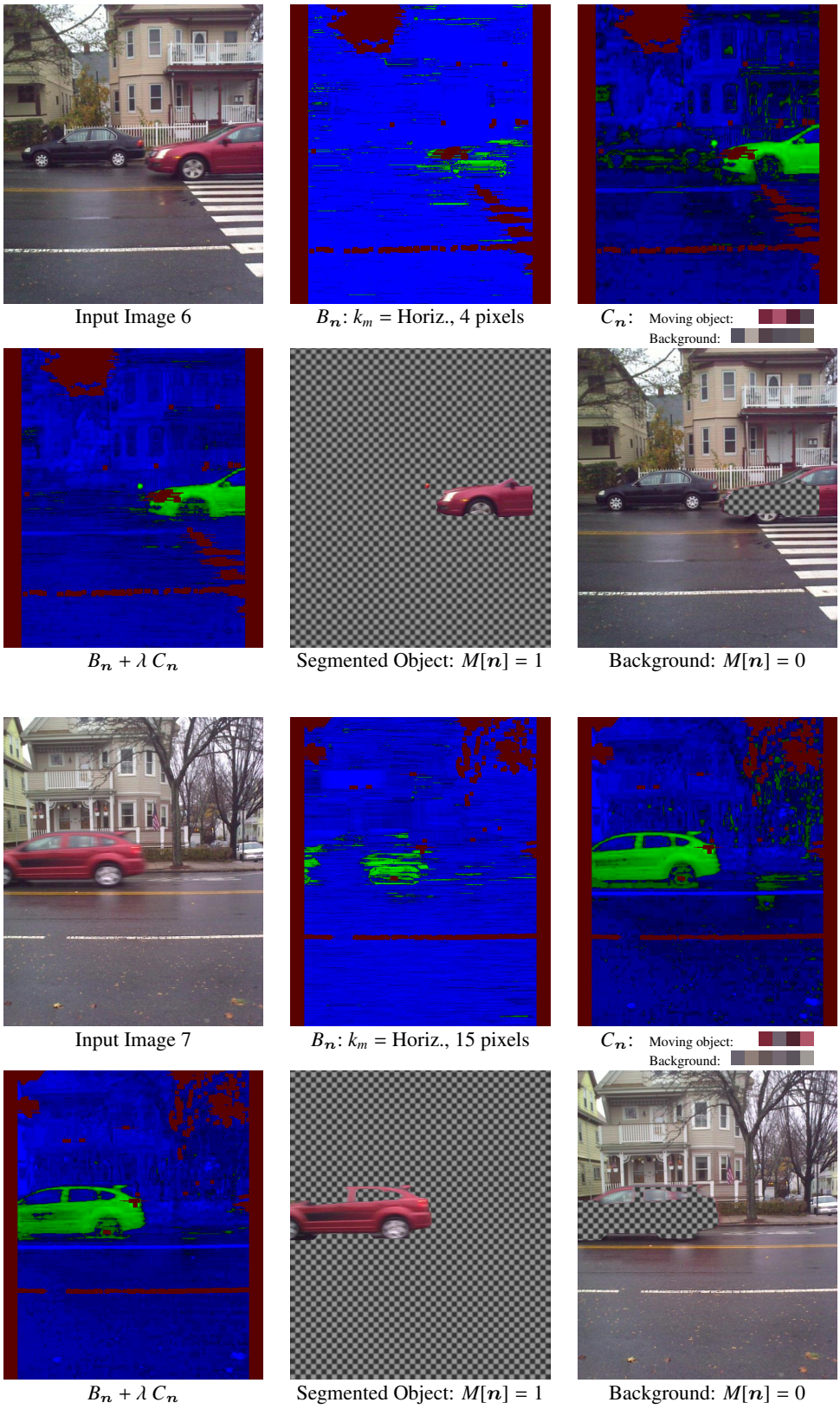


Figure 2.5: *(continued)*

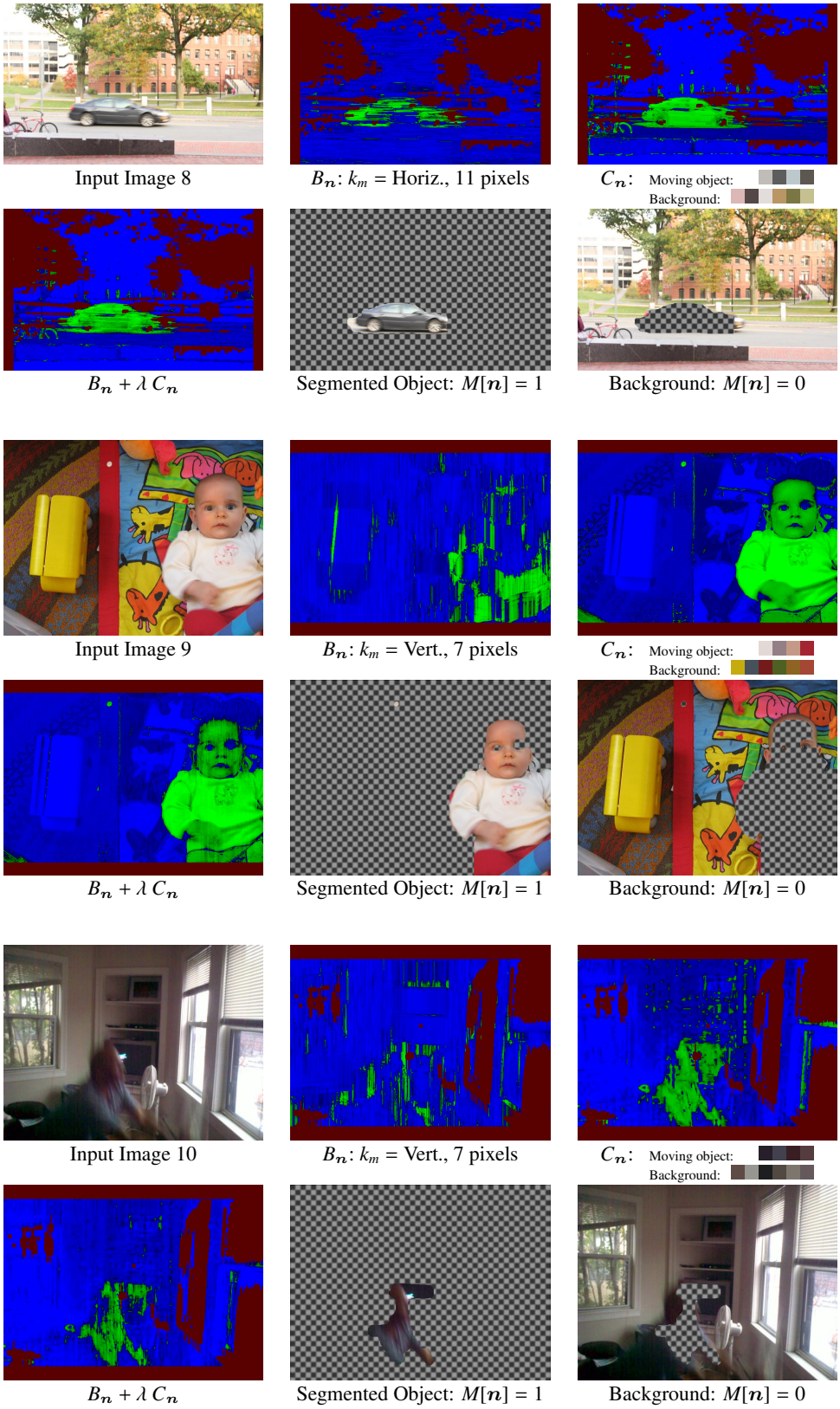


Figure 2.5: *(continued)*

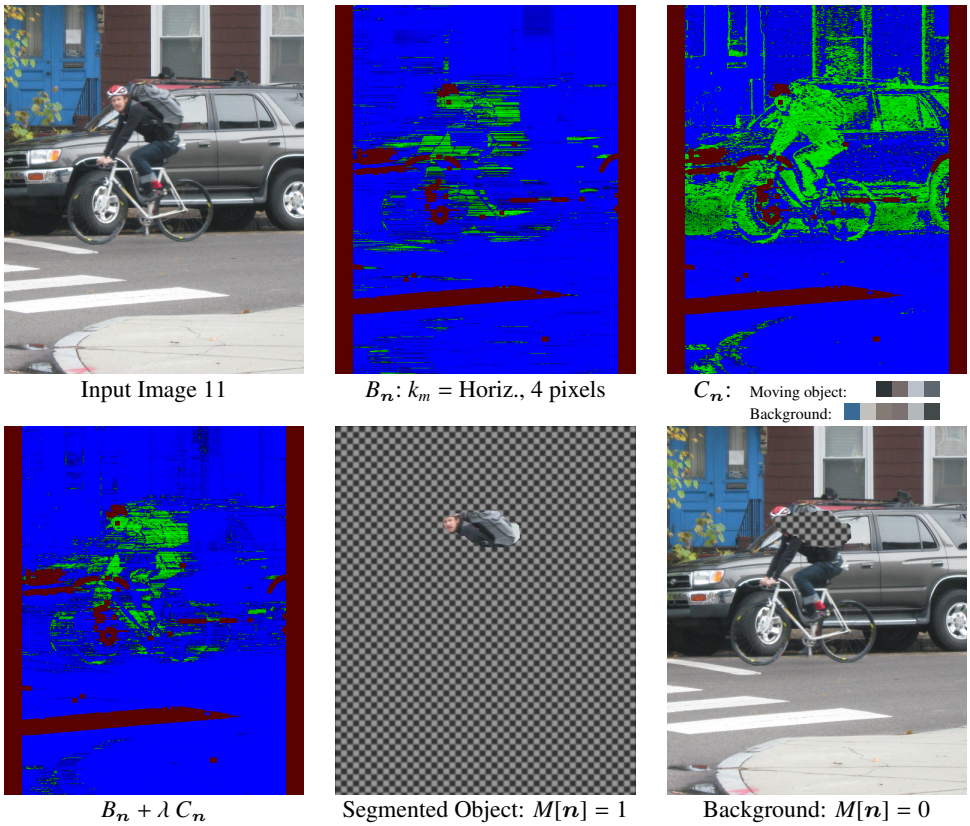


Figure 2.5: *(continued)*

objects.

These results also demonstrate the complimentary nature of the blur and color cues. We show that the blur map B_n typically has poor localization, with a region of confusion around the boundaries equal to the window length W. While the color map C_n has much better localization, since it is based on individual pixel colors, it is unable to differentiate between static and moving regions with similar appearances— for instance, the two swings in image 1. But considered jointly, these cues are able to overcome their individual drawbacks to yield an acceptable segmentation of the moving object.

It is important to note the limitations of this approach. The color cue's localization near the boundaries, though better than that of the blur cue, is not perfect. This is because the pixels in that region are weighted sums of the moving object and background, and will not be perfect fits

to either of the object or background color models. The uniform motion assumption can also be limiting in cases of complex non-rigid motion, such as in images 3, 4, 5 and 9, where both the true motion speed and orientation change across the object. Finally, the blue cue is prone to misclassification in cases when there are stationary image regions that in fact appear blurred based on the local gradient information, due to soft shadows for example, as is the case for the background object in image 3. But note that these classification errors occur only when the region in question matches the gradient statistics that *specifically* correspond to the selected kernel k_m, rather than for every region that is a poor fit to our model for sharp edges.

2.7 Discussion

In this chapter, we described a statistical approach to analyze the spatially-varying parameters of motion blur that acts on an image. This method involved the use of a local Fourier decomposition which gave us direct access to the effect of blur on image gradient statistics. This was combined with an image model that was able to differentiate between sharp and blurred edges, without being adversely affected by the natural variation in edge contrast that occurs in a typical image. Equipped with this model and representation, we derived a closed form expression for the likelihood of a local image window being blurred by a candidate kernel.

We used this likelihood measure to segment out moving objects from an image based on motion blur, and as a part of this segmentation, also estimated the direction and "speed" of motion. Since the blur model was based on image windows that were assumed to be blurred with a uniform kernel, we augmented it with a color-based appearance model to yield better localizations at object boundaries. We described a segmentation algorithm that combined cues from both the blur and color models, and evaluated it on a database of diverse real images captured using consumer-level cameras.

Future research should look at improving the quality of segmentations further, especially

at the boundaries, by reasoning about both the blur and color models in these regions. This reasoning may have to take the relative depth of the moving object into account, for example, to consider different types of occlusion scenarios. The blur model could also be made more robust by a pre-processing step that distinguishes between material boundaries, that are expected to be sharp, and smooth gradients from shading effects that are poor fits to our current model, as we saw with soft shadows in the experimental results. This processing step may require color information to be directly used in the blur model itself.

Local Fourier decompositions with changing window sizes should also be considered for the blur model, to optimally trade off localization and discriminability between blur kernels. For instance, if the motion kernel k_m is found to be small, a smaller window can be used for the segmentation part of the algorithm than the one used to select the kernel. Approaches to combine different window sizes during segmentation should also be considered, with smaller windows near boundaries and larger ones in the interior of objects likely to yield better results than a uniform window size.

More broadly, the underlying blur model can be deployed for other vision tasks that involve spatially-varying blur. For example, for defocus blur where the kernel changes with scene depth, one can imagine using the framework in [41] but doing away with the requirement for special capture conditions by using the proposed likelihood measure in (2.18) instead of cues from a coded aperture. Other applications with motion blur are also worth considering. For instance, instead of assuming uniform rigid motion, one could make local estimates of the blur kernel (possibly using a weak smoothness constraint) to yield a single image counterpart to *optical flow* [49]. These estimates, possibly combined with other cues, could allow the analysis of both object and camera motion that is more complex than the case considered here.

The blur model in this chapter was based purely on the texture content of sharp images, *i.e.* greyscale gradient information, since the action of blur (aside for cases of chromatic aberration) is independent of color. In the next chapter, we address the problem of estimating the color of the

illuminant in a scene. The nature of this task makes color a core cue during inference, and therefore we will introduce an appropriate model that jointly encodes the color and texture statistics of natural images.

3

Color Constancy

"The red brick wall was the color of a brick-red Crayola."

— Douglas Adams

In this chapter, we tackle the problem of color constancy— correcting the colors of an observed image to remove the cast caused by the spectral distribution of the scene illuminant. This is an ill-posed problem since both the illuminant spectrum and scene reflectances are unknown. Therefore, we develop a statistical model for colors in canonical white-balanced images (*i.e.* those that have no color cast), and use this model to infer the most likely illumination parameters in an observed image. We find that by applying spatial band-pass filters to color images, one unveils color distributions that are well-represented by a simple parametric form. Once these distributions are fit to training data, they enable efficient maximum likelihood estimation of the dominant illuminant in a new image. We show that this estimation method can be easily extended to leverage prior information about illuminant statistics. Finally, experimental results on a standard image database are used to verify the accuracy of these estimates.

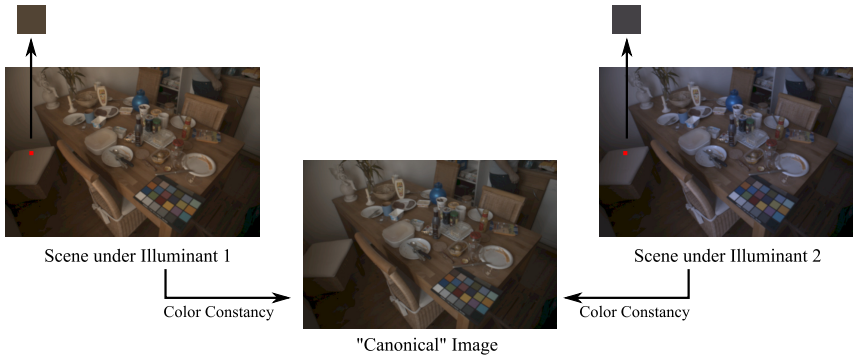


Figure 3.1: The observed colors in a scene depend on the spectrum of the illuminant. To estimate a stable color descriptor that is independent of illumination conditions, one seeks to correct for the "color cast" corresponding to the unknown illuminant. This process is known as *color constancy*.

3.1 Introduction

In addition to its intrinsic reflectance properties, the observed color of a material depends on the spectral and spatial distributions of its surrounding illumination. The same object can therefore have different recorded color values in different environments (see Fig. 3.1). In order to use color as a reliable cue in applications such as recognition, we must somehow compensate for these extrinsic factors and infer a color descriptor that is stable despite changes in lighting. The ability to make this inference, termed *color constancy*, is exhibited by the human visual system to a certain degree, and there are clear benefits to building it into machines.

One important part of computational color constancy, and the part we consider here, is compensating for the "color cast" that affects the image as a whole. For this, one ignores spatial variations in lighting spectra (as might be caused by multiple light sources), and makes the assumption that the spectrum of the illuminant is approximately uniform throughout the scene. The problem is then one of inferring the inverse map $\boldsymbol{M}^{-1}: \mathbb{R}^3 \rightarrow \mathbb{R}^3$ from the space of observed tristimulus vectors (RGB, human cone, *etc.*) to the space of *canonical colors*, *i.e.* those that would have been obtained for the same scene by a standard observer under a standard illuminant. This version of the color constancy problem is ill-posed since there exist multiple pairs of illuminant spectrum and scene reflectances that serve as equally valid solutions for any observed image. To resolve this

ambiguity, we require a prior model for the set of canonical scene colors, and optionally one for illuminants as well. The color cast (*i.e.* the map $\boldsymbol{M}$) can then be uniquely determined by choosing the solution that is the most likely under the prior.

Traditional approaches to this problem use models based on statistics of per-pixel colors. The principal challenge they face is in defining statistics that are informative and reliable, while allowing tractable inference. Methods such as grey-world [50] and white-patch [51] use simple first-order statistics— the mean and brightest scene color respectively. While these algorithms have the benefit of speed, they are susceptible to outliers and dominant colors in a scene and can fail quite dramatically. On the other hand, using sophisticated models for per-pixel scene colors, such as the shape of their convex hull in color space [52] or a non-parametric representation of their empirical probability distribution [53–55], lead to improvements in performance but involve significant computational cost during estimation.

We introduce a model that goes beyond statistics of per-pixel colors and leverages the joint spatio-spectral structure that is found to exist in natural images. Motivated by the success of filter-based methods [56, 57], we begin by decomposing an input color image into distinct spatial sub-bands and then define an image prior in terms of separate models for the color statistics of each sub-band. Unlike per-pixel colors, we find that the empirical probability distributions of the colors in each sub-band can be accurately represented using simple parametric forms. Furthermore, they allow robust and efficient inference of illumination color cast for an observed image. The resulting method outperforms existing approaches on a standard database, and allows incorporating a prior model for illuminants in a statistical framework.

3.2 Problem Formulation

Assuming a Lambertian model, we denote the effective spectral reflectance of a surface patch observed at pixel $\boldsymbol{n} \in \mathbb{Z}^2$ by $\kappa(\lambda, \boldsymbol{n}) \in \mathbb{R}$, where $\lambda \in \mathbb{R}$ denotes wavelength. Here, κ accounts

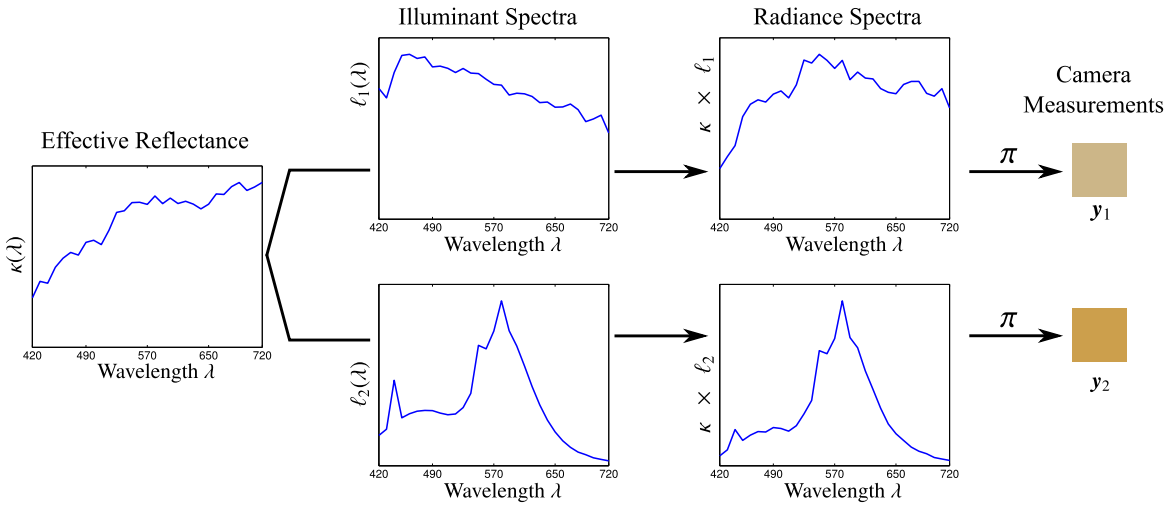


Figure 3.2: Camera measurements under different illuminants. The effective surface reflectance (κ) gets shaped by the spectrum (ℓ) of the scene illuminant. Different illuminants can lead to significantly different radiance spectra ($\kappa \times \ell$) being incident on the camera, in turn leading to different trichromatic measurements by the camera's sensors (π).

for both the material reflectance and surface orientation with respect to the illuminant. We assume a single dominant scene illuminant and represent its spectral distribution by $\ell(\lambda)$. As depicted in Fig. 3.2, the spectral power distribution (SPD) of the radiance that is emitted toward the observer is then the product of ℓ and κ.

The three spectral measurements recorded by a camera at each pixel $\boldsymbol{n}$ (assuming ideal *demosaicking* [15, 58]) are then given by

$$\boldsymbol{y}[\boldsymbol{n}] = \left[y^{\{1\}}[\boldsymbol{n}],\ y^{\{2\}}[\boldsymbol{n}],\ y^{\{3\}}[\boldsymbol{n}]\right]^T = \int \boldsymbol{\pi}(\lambda)\ \kappa(\lambda, \boldsymbol{n})\ \ell(\lambda)\ d\lambda, \tag{3.1}$$

where $\boldsymbol{\pi}(\lambda) = [\pi^{\{1\}}(\lambda), \pi^{\{2\}}(\lambda), \pi^{\{3\}}(\lambda)]^T$ are the spectral transmittance distributions of the camera's color filters. From image $\boldsymbol{y}[\boldsymbol{n}]$ our task is to estimate an illuminant-invariant representation, or "canonical color" image, $\boldsymbol{x}[\boldsymbol{n}]$ given by

$$\boldsymbol{x}[\boldsymbol{n}] = \int \boldsymbol{\pi}(\lambda)\ \ell_0(\lambda)\ \kappa(\lambda, \boldsymbol{n})\ d\lambda. \tag{3.2}$$

We refer to ℓ_0 as the "canonical illuminant", and for the specific choice $\ell_0(\lambda) = 1$, each canonical color $\boldsymbol{x}[\boldsymbol{n}]$ can be interpreted as the trichromatic projection (via $\boldsymbol{\pi}$) of the effective reflectance $\kappa(\lambda, \boldsymbol{n})$ at the corresponding surface point.

Note that the colors $\boldsymbol{y}$ and $\boldsymbol{x}$ in (3.1) and (3.2) are trichromatic reductions of the full SPD that cannot be exactly reversed, even when the illuminant ℓ and ℓ_0 are known. However, prior work [59–61] suggests that a linear mapping of the form

$$\boldsymbol{x}[\boldsymbol{n}] = \boldsymbol{M}^{-1}\, \boldsymbol{y}[\boldsymbol{n}], \qquad \boldsymbol{M} \in \mathbb{R}^{3\times 3} \tag{3.3}$$

can achieve accurate chromatic adaptation in many cases, and that it can often be approximated further as a diagonal transformation (*i.e.*, $M_{ij} = 0$, if $i \neq j$). We apply such diagonal mappings directly in the color space defined by $\boldsymbol{\pi}$, even though applying them in an optimized (often called "sharpened") color space can improve performance [59–61]. For this reason, our results may provide a conservative estimate of what can be achieved by our algorithm. With some abuse of notation, we let $\boldsymbol{M}$ represent a diagonal matrix in the subsequent sections and refer to its recovery as "estimating the illuminant".

3.3 Related Work

Having settled on a linear diagonal form for the mapping from input color $\boldsymbol{y}$ to canonical color $\boldsymbol{x}$, the color constancy problem reduces to the task of estimating the three diagonal entries of $\boldsymbol{M}$. Since both the illuminant and scene reflectances are unknown, this is an under-determined problem. Prior work in this area typically addresses this by introducing a statistical prior model for surface reflectance κ or canonical color $\boldsymbol{x}$. White-patch [51] and grey-world [50] are two well-known methods with intuitive interpretations. Based on the observation that color-neutral or achromatic surfaces (*i.e.* those with constant spectral reflectance) are the most efficient reflectors, the white-patch [51] algorithm posits that pixels observed to have the greatest intensity correspond to a color-neutral surface patch. Similarly, based on the assertion that the mean surface reflectance in a scene is likely to be color-neutral, the grey-world [50] method assumes that the sample mean of canonical pixel colors is achromatic. The diagonal elements of $\boldsymbol{M}$ are then estimated as those that map the corresponding vector $\boldsymbol{y}[\boldsymbol{n}_{\text{brightest}}]$ or $\text{avg}(\boldsymbol{y}[\boldsymbol{n}])$ to white.

Other approaches perform more sophisticated analysis of pixel color distributions. The gamut-mapping [52] method defines constraints on the set of canonical scene colors $\{\boldsymbol{x}[\boldsymbol{n}]\}_{\boldsymbol{n}}$ in terms of their convex hull, or "gamut". Given an input image $\boldsymbol{y}$, this approach estimates the illuminant $\boldsymbol{M}$ as that which maps the convex hull of $\{\boldsymbol{y}[\boldsymbol{n}]\}_{\boldsymbol{n}}$ to this expected gamut. Finlayson *et al.* [62] use a "di-chromatic" model to reason that, due to varying degrees of specular reflection, the observed colors of each homogeneous surface in a scene will lie along a line in chromaticity space (defined as the R and G components of a normalized RGB vector). While colors for different surfaces correspond to different lines, the extended lines all intersect at a common point corresponding to perfect specular reflection for each surface. This intersection point corresponds to the chromaticity of the illuminant and can be used to set $\boldsymbol{M}$, along with additional constraints based on knowledge of natural illuminant chromaticities.

Another alternative is to develop a complete prior probability distribution over canonical color images that permits, along with a prior distribution over illuminant spectra, the Bayesian estimation of $\boldsymbol{M}$. To make this Bayesian approach tractable, pixels are assumed to be independent and identically distributed with a probability density that is either Gaussian [53], or more generally, non-parametric [54, 55]. Related to these are a number of leaning-based approaches that use databases of canonical color images to learn linear filters [63] or train neural networks [64] that can then be used to infer illuminant $\boldsymbol{M}$ for a novel input image.

All of the above methods can be considered "pixel-based" because they model the set of individual pixel colors without considering each pixel's spatial context. One can arbitrarily re-order the pixels of any input image, for example, without affecting the resulting estimate of illuminant $\boldsymbol{M}$. Because they ignore spatial structure, these methods can be negatively influenced by the presence of large colorful objects or regions that skew the color histogram. A variety of work suggests that improvements might be gained by employing spatial image features, such as segmentations or linear filter responses, that incorporate spatial information in a tractable manner. For example, Gershon *et al.* [65] assume that the average of mean colors of segmented regions of an image, rather than of

individual pixels, is color neutral. Alternatively, van de Weijer *et al.* [56] and Gijsenij *et al.* [57] respectively apply the grey-world and gamut-mapping procedures described above to the outputs of linear filters instead of the individual pixels they were originally designed for. The grey-edge [56] method posits that image gradients are on average color neutral, while the generalized-gamut [57] algorithm proposes strategies for combining cues from the expected gamuts of various linear filter coefficients. Finally, Singh *et al.* [66] reason about the illuminant using a linear spatio-spectral basis for small spatial patches of color images.

We propose an algorithm that seeks to leverage joint spatial and spectral structure in a more efficient and reliable manner. We achieve this by representing the prior probability distribution over canonical color images in terms of the color coefficients in distinct spatial sub-bands. We empirically show that, unlike per-pixel colors, these sub-band coefficients have kurtotic distributions that can be well-represented using convenient parametric forms, and that these parametric forms can be used in a maximum-likelihood framework to estimate the illuminant $\boldsymbol{M}$ efficiently.

3.4 Spatio-Spectral Modeling

As stated above, we draw inspiration from previous studies [56, 57, 66] and move beyond statistics of individual pixels colors to exploit information about a pixel's spatial context. We take an approach similar to the grey-edge [56] and generalized-gamut [57] methods, by looking at the properties of filter coefficients instead of individual pixels. These methods represent important first steps in looking beyond individual pixels and demonstrate that filter coefficients provide more robust cues for color constancy. However, they are based on the direct application of pixel-based color constancy techniques, grey-world [50] and gamut-mapping [52] respectively, to filter coefficients. Therefore, there is reason to believe that a new estimation procedure specifically tailored to the statistical behavior of these coefficients can improve performance.

We begin with the observation that statistics of filter coefficients show far more structure

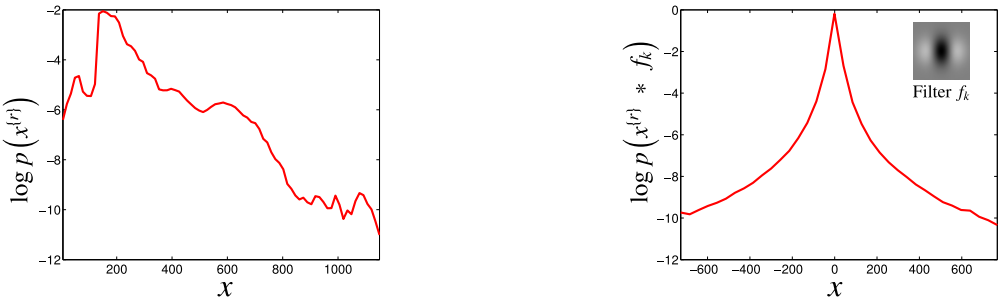


Figure 3.3: Distributions of individual pixels vs. filter coefficients. Shown are empirical log-histograms of the red values of individual pixels (Left) and coefficients (Right) of a sub-band filter (Inset). The coefficient histogram shows more structure and can be modeled by parametric distributions.

than those of individual pixels. Figure 3.3 compares the empirical histograms of red values of individual pixels to those of coefficients for a particular band-pass filter, in images under a canonical illuminant. We note that the histogram for the filter coefficients is uni-modal and symmetric, while individual pixels show less discernible statistical structure. This lack of structure provides insight in to the limitations that pixel-based color constancy methods face in terms of building reliable and tractable models. In contrast, sub-band filter coefficients have distributions that can be described accurately by parametric models. In this section, we propose such a model and describe a training algorithm to learn its parameters from data. Subsequent sections present methods that use this model to estimate the unknown illuminant from an observed image.

3.4.1 Image Model

We begin by defining a statistical model for an image $\boldsymbol{x}[\boldsymbol{n}]$ observed under canonical illumination. Since pixels in a spatial neighborhood are strongly correlated, we first apply a spatially-decorrelating transform by filtering the image with a series of spatial sub-band filters $\{f_k\}_k$. While these filters can be chosen to correspond to any standard image decomposition (such as wavelets or steerable pyramids), Gaussian derivative filters have proved successful in edge-based color constancy methods [56, 57]. Therefore, we use a decomposition based on horizontal and vertical second-derivative Gaussian filters at multiple scales as illustrated in Fig. 3.4. Each filter is ap-

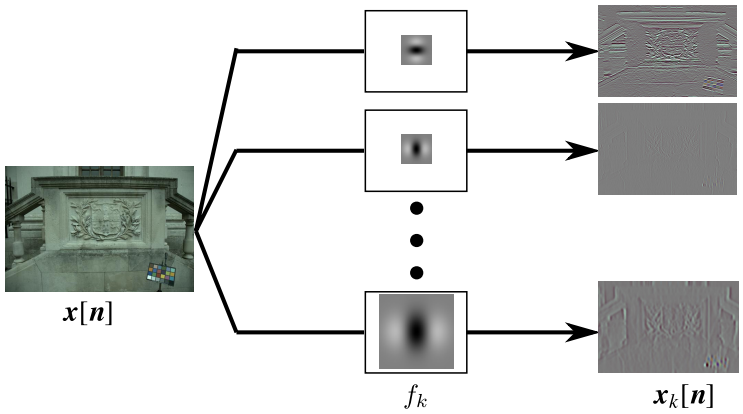


Figure 3.4: Sub-band Decomposition. We use horizontal and vertical second-derivative Gaussian filters at multiple scales to spatially decorrelate the image, and model corresponding coefficients independently.

plied separately to all channels of the image, and we define the trichromatic color coefficient vector $\boldsymbol{x}_k[\boldsymbol{n}]$ as

$$\boldsymbol{x}_k[\boldsymbol{n}] = \left[(x^{\{1\}} * f_k)[\boldsymbol{n}], (x^{\{2\}} * f_k)[\boldsymbol{n}], (x^{\{3\}} * f_k)[\boldsymbol{n}]\right]^T, \tag{3.4}$$

where $*$ denotes convolution. We assume that following decomposition by the filter bank $\{f_k\}$, each sub-band canonical color image $\boldsymbol{x}_k[\boldsymbol{n}]$ can be modeled as being independent from the rest.

We represent the prior probability of each sub-band canonical color image as the product of independent and identical distributions per trichromatic coefficient, $\prod_{\boldsymbol{n}} p\left(\boldsymbol{x}_k[\boldsymbol{n}]\right)$, and for the coefficient distributions we use the "radial exponential" [67]:

$$p(\boldsymbol{x}_k) = \frac{1}{\pi\sqrt{\det(\boldsymbol{\Sigma}_k)}} \exp\left(-2\sqrt{\boldsymbol{x}_k^T \boldsymbol{\Sigma}_k^{-1} \boldsymbol{x}_k}\right), \tag{3.5}$$

where $\boldsymbol{\Sigma}_k$ is a 3×3 positive-definite matrix corresponding to the covariance of $\boldsymbol{x}_k$. Note that $p(\boldsymbol{x}_k)$, as defined in (3.5), is more kurtotic, or heavy-tailed, than a multi-variate Gaussian distribution, and can be thought of as a generalization of the Laplace distribution to the multi-variate case.

It differs, however, from the standard multi-variate Laplace distribution [68] in that the multi-variate Laplace has equiprobable contours that are L_1-spherical or "diamond" shaped (see Fig. 3.5 (left)) while those of the radial exponential distribution are ellipsoidal (see Fig. 3.5 (center)). This implies that the components of $\boldsymbol{x}_k$ along the eigen-vectors of $\boldsymbol{\Sigma}_k$ are un-correlated but *not*

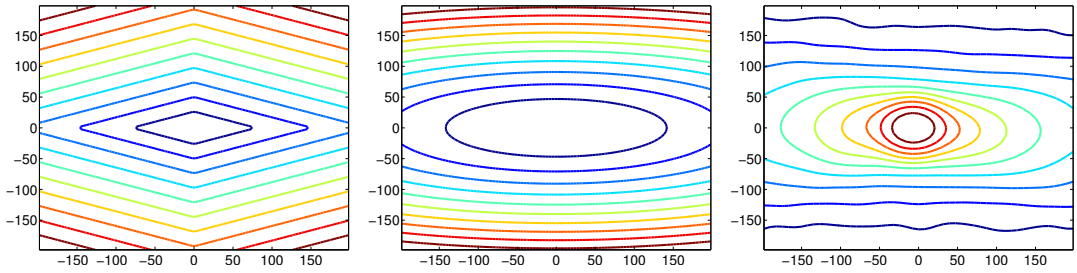


Figure 3.5: Equiprobable contours of the distribution of filter coefficient vectors $\boldsymbol{x}_k[\boldsymbol{n}]$, corresponding to the filter shown in Fig. 3.3. Contours are shown along the two major eigen-vectors of the covariance matrix $\boldsymbol{\Sigma}_k$, for a multi-variate Laplace distribution (Left), a radial exponential distribution (Center), and the empirical distribution computed using kernel density estimation (Right). The empirical distribution has elliptical contours, making the radial exponential distribution a more appropriate choice for modeling these coefficients.

independent under (3.5). Figure 3.5 (right) shows the equiprobable contours of the actual empirical distribution of $\boldsymbol{x}_k$ computed using kernel density estimation, and we see that these contours are indeed ellipsoidal indicating that the choice of the "radial exponential" distribution for $p(\boldsymbol{x}_k)$ in (3.5) is appropriate.

3.4.2 Learning Model Parameters

Having defined a parametric statistical model for colors in a canonical image, $p(\boldsymbol{x}) = \prod_k \prod_{\boldsymbol{n}} p\left(\boldsymbol{x}_k[\boldsymbol{n}]\right)$, we now require the ability to fit the model parameters $\boldsymbol{\Sigma}_k$ given a training set of canonical images. We treat each sub-band separately during this training step, so the problem amounts to inferring the entries of a 3×3 matrix $\boldsymbol{\Sigma}_k$ given the correspond sub-band coefficients in the training set, $\{\boldsymbol{x}_{k,t}\}_{t=1}^{T}$. Although, $\boldsymbol{\Sigma}_k$ corresponds to the covariance matrix of $\boldsymbol{x}_k$, the maximum likelihood (ML) estimate of $\boldsymbol{\Sigma}_k$ is *not* the empirical covariance of $\{\boldsymbol{x}_{k,t}\}$. The log-likelihood of the training set coefficients as a function of $\boldsymbol{\Sigma}_k$ is

$$J_k(\boldsymbol{\Sigma}) = \sum_t \log p\left(\boldsymbol{x}_{k,t}|\boldsymbol{\Sigma}\right) = -\frac{T}{2}\log\det(\boldsymbol{\Sigma}) - \sum_t 2\sqrt{\boldsymbol{x}_{k,t}^T \boldsymbol{\Sigma}^{-1}\boldsymbol{x}_{k,t}}. \tag{3.6}$$

The ML estimate of $\boldsymbol{\Sigma}_k$ is therefore given by

$$\boldsymbol{\Sigma}_k = \arg\max\ J_k(\boldsymbol{\Sigma}). \tag{3.7}$$

Unfortunately, this does not have a closed-form solution and we therefore propose an iterative procedure to compute $\boldsymbol{\Sigma}_k$.

We begin the iterations by initializing $\boldsymbol{\Sigma}_k$ to the identity matrix. Then at each iteration, based on the current estimate $\boldsymbol{\Sigma}_k^*$ of $\boldsymbol{\Sigma}_k$, we define an approximation to $J_k(\cdot)$ as

$$J_k^*(\boldsymbol{\Sigma}|\boldsymbol{\Sigma}_k^*) = -\frac{T}{2}\log\det(\boldsymbol{\Sigma}) - \sum_t 2\frac{\boldsymbol{x}_{k,t}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{x}_{k,t}}{\sqrt{\boldsymbol{x}_{k,t}^T\boldsymbol{\Sigma}_k^{*-1}\boldsymbol{x}_{k,t}}}. \tag{3.8}$$

Accordingly, we update $\boldsymbol{\Sigma}_k$ in the following iteration as

$$\boldsymbol{\Sigma}_k = \arg\max_{\boldsymbol{\Sigma}} J_k^*(\boldsymbol{\Sigma}|\boldsymbol{\Sigma}_k^*) = \frac{4}{T}\sum_t \frac{\boldsymbol{x}_{k,t}\boldsymbol{x}_{k,t}^T}{\sqrt{\boldsymbol{x}_{k,t}^T\boldsymbol{\Sigma}_k^{*-1}\boldsymbol{x}_{k,t}}}. \tag{3.9}$$

Note that true ML estimate is a fixed point of these iterations, and in practice, we find that the procedure converges quickly (usually in less than five iterations).

3.5 Maximum-Likelihood Illuminant Estimation

Once we have learned the parameters of the prior model, we are ready to infer the illuminant $\boldsymbol{M}$ for an input color image $\boldsymbol{y}[\boldsymbol{n}]$. As per our model in (3.3), we assume that $\boldsymbol{y}[\boldsymbol{n}] = \boldsymbol{M}\boldsymbol{x}[\boldsymbol{n}]$, where $\boldsymbol{M}$ is a diagonal 3×3 matrix with positive entries. Since convolution is a linear operation, it follows that $\boldsymbol{y}_k[\boldsymbol{n}] = \boldsymbol{M}\boldsymbol{x}_k[\boldsymbol{n}]$, where $\boldsymbol{y}_k[\boldsymbol{n}]$ are the color sub-band coefficients of $\boldsymbol{y}[\boldsymbol{n}]$ for filter f_k as in (3.4).

We begin by noting that under our prior model for $\boldsymbol{x}_k$, the likelihood of $\boldsymbol{y}_k$ conditioned on the illuminant $\boldsymbol{M}$ is given by

$$\begin{aligned} p(\boldsymbol{y}_k|\boldsymbol{M}) &= \frac{1}{\pi\sqrt{\det(\boldsymbol{M}\boldsymbol{\Sigma}_k\boldsymbol{M})}}\exp\left(-2\sqrt{\boldsymbol{y}_k^T(\boldsymbol{M}\boldsymbol{\Sigma}_k\boldsymbol{M})^{-1}\boldsymbol{y}_k}\right) \\ &\propto \frac{1}{\det(\boldsymbol{M})}\exp\left(-2\sqrt{\boldsymbol{y}_k^T(\boldsymbol{M}\boldsymbol{\Sigma}_k\boldsymbol{M})^{-1}\boldsymbol{y}_k}\right). \end{aligned} \tag{3.10}$$

Therefore, the observed coefficients $\boldsymbol{y}_k[\boldsymbol{n}]$ also have a radial exponential distribution, with the covariance matrix $\boldsymbol{M}\boldsymbol{\Sigma}_k\boldsymbol{M}$. Figure 3.6 compares the covariance matrices $\boldsymbol{\Sigma}_k$ and $\boldsymbol{M}\boldsymbol{\Sigma}_k\boldsymbol{M}$ for a

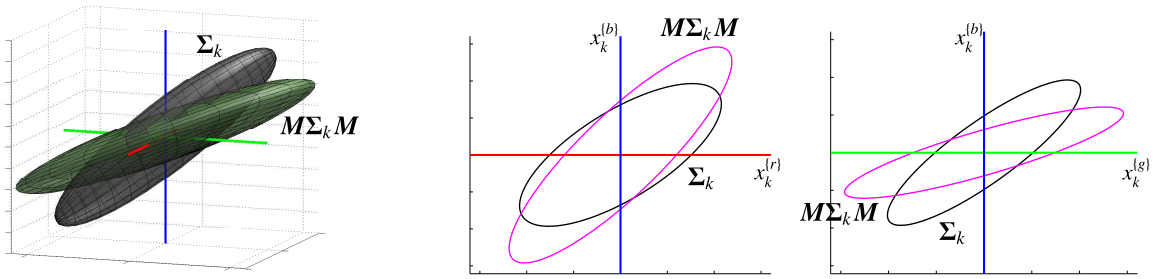


Figure 3.6: Covariance matrices $\boldsymbol{\Sigma}_k$ and $\boldsymbol{M}\boldsymbol{\Sigma}_k\boldsymbol{M}$, for coefficients observed under a canonical illumination and a typical illuminant $\boldsymbol{M}$. (Left) Ellipsoids corresponding to the covariance matrices, and (Right) their projections onto the R-B and G-B planes. The illuminant causes a skew in the shape of the covariance matrix, which serves as a cue for our estimation method.

typical illuminant $\boldsymbol{M}$. This difference between the kth sub-band coefficient distributions for the input image $\boldsymbol{y_k}$ and the canonical color image $\boldsymbol{x_k}$, as embodied by these covariance matrices, is the fundamental cue that we will exploit for illuminant estimation.

For notational convenience, we define $\boldsymbol{m} = [m_1, m_2, m_3]$ and $\boldsymbol{w} = [w_1, w_2, w_3]$ to be the diagonal elements of $\boldsymbol{M}$ and $\boldsymbol{M}^{-1}$ respectively, where $m_i = w_i^{-1}$. Note the following identities for any $\boldsymbol{y} \in \mathbb{R}^3$ and $\boldsymbol{S} \in \mathbb{R}^{3\times3}$:

$$\boldsymbol{M}^{-1}\boldsymbol{y} = \boldsymbol{w}\,\mathrm{diag}(\boldsymbol{y}), \qquad \mathrm{diag}(\boldsymbol{y})\,\boldsymbol{S}\,\mathrm{diag}(\boldsymbol{y}) = \left(\boldsymbol{y}\boldsymbol{y}^T\right) \circ \boldsymbol{S}, \tag{3.11}$$

where $\mathrm{diag}(\boldsymbol{y}) \in \mathbb{R}^{3\times3}$ refers to a diagonal matrix whose entries correspond to $\boldsymbol{y}$, and $\boldsymbol{A} \circ \boldsymbol{B}$ denotes the Hadamard or "entry-wise" product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. Therefore, the ML estimate of $\boldsymbol{M}$, conditioned on all observed coefficients $\{\boldsymbol{y}_k[\boldsymbol{n}]\}_{k,\boldsymbol{n}}$, is given by

$$\begin{aligned}
\hat{\boldsymbol{M}}_{ML} &= \arg \max_{\boldsymbol{M}\in\mathrm{diag}(\mathbb{R}^3)} \sum_{k,\boldsymbol{n}} \log p\left(\boldsymbol{y}_k[\boldsymbol{n}]|\boldsymbol{M}\right) \\
&= \arg\min \sum_{k,\boldsymbol{n}} \left[\log\det(\boldsymbol{M}) + 2\sqrt{\boldsymbol{y}_k^T[\boldsymbol{n}](\boldsymbol{M}\boldsymbol{\Sigma}_k\boldsymbol{M})^{-1}\boldsymbol{y}_k[\boldsymbol{n}]}\,\right] \\
&= \arg\min N\log m_1 m_2 m_3 + \sum_{k,\boldsymbol{n}} 2\sqrt{\boldsymbol{w}^T\left[(\boldsymbol{y}_k[\boldsymbol{n}]\boldsymbol{y}_k^T[\boldsymbol{n}]) \circ \boldsymbol{\Sigma}_k^{-1}\right]\boldsymbol{w}},
\end{aligned} \tag{3.12}$$

where $N = \sum_{k,\boldsymbol{n}} 1$ is the total number of coefficients across all bands. Once we have solved for the ML estimate $\hat{\boldsymbol{M}}_{ML}$ above, the canonical image $\boldsymbol{x}[\boldsymbol{n}]$ can be computed simply as $\boldsymbol{x}[\boldsymbol{n}] = \hat{\boldsymbol{M}}_{ML}^{-1}\boldsymbol{y}[\boldsymbol{n}]$.

We propose an iterative algorithm to solve (3.12) using a similar approach as the training method in Sec. 3.4.2. We begin by initializing $\boldsymbol{M}$ to the identity. Based on the estimate $\boldsymbol{M}^*$ of the illuminant at every iteration, we approximate the cost function in (3.12) as

$$\begin{aligned} L(\boldsymbol{M}|\boldsymbol{M}^*) &= N \log m_1 m_2 m_3 + \sum_{k,\boldsymbol{n}} 2 \frac{\boldsymbol{w}^T \left[(\boldsymbol{y}_k[\boldsymbol{n}] \boldsymbol{y}_k^T[\boldsymbol{n}]) \circ \boldsymbol{\Sigma}_k^{-1} \right] \boldsymbol{w}}{\sqrt{\boldsymbol{y}_k^T[\boldsymbol{n}] (\boldsymbol{M}^* \boldsymbol{\Sigma}_k \boldsymbol{M}^*)^{-1} \boldsymbol{y}_k^T[\boldsymbol{n}]}} \\ &= N \left[\log m_1 m_2 m_3 + \frac{1}{2} \boldsymbol{w}^T \boldsymbol{A}^* \boldsymbol{w} \right], \end{aligned} \tag{3.13}$$

where $\boldsymbol{A}^*$ is a 3×3 symmetric matrix given by

$$\boldsymbol{A}^* = \frac{4}{N} \sum_k \left(\sum_{\boldsymbol{n}} \frac{\boldsymbol{y}_k[\boldsymbol{n}] \boldsymbol{y}_k^T[\boldsymbol{n}]}{\sqrt{\boldsymbol{y}_k^T[\boldsymbol{n}] (\boldsymbol{M}^* \boldsymbol{\Sigma}_k \boldsymbol{M}^*)^{-1} \boldsymbol{y}_k^T[\boldsymbol{n}]}} \right) \circ \boldsymbol{\Sigma}_k^{-1}. \tag{3.14}$$

In the next iteration, the illuminant estimate $\boldsymbol{M}$ is set as

$$\boldsymbol{M} = \arg\min L(\boldsymbol{M}|\boldsymbol{M}^*). \tag{3.15}$$

This minimization must also be done iteratively. There is a closed-form solution to this expression for each element of m_i of $\boldsymbol{m}$ when the others elements $m_{j \neq i}$ are fixed. This is given by

$$\begin{aligned} &\frac{\partial}{\partial m_i} L(\boldsymbol{M}|\boldsymbol{M}^*) = 0 \Rightarrow m_i^2 - \left(\sum_{j \neq i} \frac{A_{ji}^*}{m_j} \right) m_i - A_{ii}^* = 0 \\ &\Rightarrow m_i = \frac{1}{2} \left(\sum_{j \neq i} \frac{A_{ji}^*}{m_j} + \sqrt{\left(\sum_{j \neq i} \frac{A_{ji}^*}{m_j} \right)^2 + 4 A_{ii}^*} \right). \end{aligned} \tag{3.16}$$

We apply (3.16) iteratively on each channel i, keeping the others fixed. Once we have converged to a solution for (3.15) through these "inner" iterations, we update the approximate cost as per (3.13).

3.6 Illuminant Prior

Implicit in the maximal likelihood estimation described in the previous section is the notion that all illuminants are equally likely. In this section, drawing inspiration from existing color constancy algorithms [53, 55] that try to leverage *a-priori* knowledge of illuminant statistics, we

develop a strategy to incorporate an illuminant prior in to the estimation process. We choose an illuminant prior distribution $p(\boldsymbol{M})$ of the form

$$p(\boldsymbol{M}) = \left(\frac{2}{\pi}\right)^{3/2} \frac{1}{(m_1 m_2 m_3)^2 \sqrt{\det(\boldsymbol{Q})}} \exp\left(-\frac{1}{2}\boldsymbol{w}^T \boldsymbol{Q}^{-1} \boldsymbol{w}\right), \tag{3.17}$$

where $\boldsymbol{Q}$ is a 3×3 positive-definite matrix, and $\boldsymbol{m}, \boldsymbol{w} > 0$ are equal to the diagonal entries of $\boldsymbol{M}$ and $\boldsymbol{M}^{-1}$ as before. Note that this corresponds to a multivariate Gaussian distribution on $\boldsymbol{w}$ (restricted to positive values), and is a conjugate prior to the likelihood expression in (3.12).

We can use our prior model to describe the statistics of scene illuminants in general, or those restricted to a category of illuminants (such as indoor v.s. outdoor) that the input image $\boldsymbol{y}[\boldsymbol{n}]$ is known to belong to. Ideally, the covariance parameter $\boldsymbol{Q}$ would be computed as a sample covariance of a training set $\{\boldsymbol{w}_t\}$ of illuminant coefficients. However, only the relative spectral radiance of illuminants is usually available, meaning the absolute luminance of the illuminants in the training set is unknown. Therefore, we compute the sample covariance $\boldsymbol{Q}'$ over the set of illuminant vectors normalized such that $\|\boldsymbol{m}_t\| = 1$ during training. During estimation, we first compute the ML estimate $\hat{\boldsymbol{M}}_{ML}$ of (3.12) without the illuminant prior, and then reweight $\boldsymbol{Q}'$ covariance parameter as

$$\boldsymbol{Q} = \|\boldsymbol{m}_{ML}\|^2 \boldsymbol{Q}'. \tag{3.18}$$

This reweighting has the effect of normalizing all elements of the training set $\{\boldsymbol{m}_t\}$ to have the same norm as the initial estimate $\|\boldsymbol{m}_{ML}\|$.

Let $\boldsymbol{y}_k[\boldsymbol{n}]$ be the kth sub-band coefficients of input image as before. The final illuminant estimate incorporating $p(\boldsymbol{M})$ is obtained by solving

$$\hat{\boldsymbol{M}} = \arg\max \left[\sum_{k,\boldsymbol{n}} \log p(\boldsymbol{y}_k[\boldsymbol{n}] | \boldsymbol{M}) \right] + \alpha \log p(\boldsymbol{M}). \tag{3.19}$$

Here, α is a scalar parameter that weights the relative contribution of the prior to the image evidence, and is learned using cross-validation to minimize estimation error on a training set. While for the special case of $\alpha = 1$, the above formulation corresponds to maximum a-posteriori (MAP) estimation, we find that a larger weight on the prior yields better estimates in practice.

We solve the minimization problem in (3.19) using an approach similar to that for (3.16). We define an approximated cost as

$$L_p(\boldsymbol{M}|\boldsymbol{M}^*) = (N + 2\alpha)\left[\log m_1 m_2 m_3 + \frac{1}{2}\boldsymbol{w}^T \boldsymbol{A}_p^* \boldsymbol{w}\right], \tag{3.20}$$

where $\boldsymbol{A}_p^*$ is a 3×3 matrix given by

$$\boldsymbol{A}_p^* = \frac{N\boldsymbol{A}^* + \alpha\boldsymbol{Q}}{N + 2\alpha}, \tag{3.21}$$

with $\boldsymbol{A}^*$ defined as per (3.14). Applying (3.16) iteratively to $\boldsymbol{A}_p^*$ (instead of $\boldsymbol{A}^*$ as was done in the previous section) solves $\boldsymbol{M} = \arg\min L_p(\boldsymbol{M}|\boldsymbol{M}^*)$. If $\boldsymbol{M}^*$ is set to $\boldsymbol{M}_{ML}$, we find that further iterations to update L_p or $\boldsymbol{Q}$ are unnecessary.

3.7 Experimental Evaluation

We evaluate the proposed method on the 568 color images collected by Gehler *et al.* [55], of which 246 were labeled as captured indoors, and 322 as captured outdoors. Each image has a color checker chart at manually marked co-ordinates which serves as ground truth, and is masked out during evaluation. We use the version of the database made available by Shi and Funt [69]. It consists of 12-bit linear images in the sensor color space generated directly from the RAW captured data — without using the camera's auto-white balance, gamma-correction or any demosaicking (the sensor data is sub-sampled to provide a trichromatic vector at each pixel). While the rest of this section describes performance on this database, we also include results for another other common databases in the appendix.

A recent survey [70] provides comprehensive evaluation of a large number of color constancy methods on this database, allowing us to conveniently compare them to the proposed algorithm. Performance is measured in terms of the angular deviation between the estimated and true illuminant, where the later is computed from the brightest grey patch in the color chart. Using the same error metric and ground truth, we compare the proposed method to the state of the

art: grey-world [50], grey-edge [56], gamut-mapping [52], generalized-gamut mapping [57] and Rosenberg [54, 55].

3.7.1 Implementation Details

We implement the proposed method with second-derivative Gaussian filters at three different scales ($\sigma = 1, 2$ and 4). We report performance both with and without the illumination prior. For the case with the prior, we evaluate a general prior over all illuminants, as well as separate priors for the indoor and outdoor images. To train the parameters of our model $\{\boldsymbol{\Sigma}_k\}, \boldsymbol{Q}'$ and α, we use three-fold cross validation where the database is split into three equal folds, and when testing on each fold, we use the remaining images for training. A MATLAB implementation of the algorithm is available for download [23].

Since the authors of [70] have made the estimated illuminants and errors for all evaluated methods available, we report performance for most other methods directly from their data. However, we incorporate an important enhancement to the evaluation of grey-world and grey-edge that improves their performance. While grey-world is often interpreted as assuming the mean color of an image to be along $[1, 1, 1]$ (as is the case in [70]), we posit that the mean is along some trichromatic unit vector $\hat{\boldsymbol{g}}$. The illuminant estimate is then given by $\hat{\boldsymbol{m}} = [\text{diag}(\hat{\boldsymbol{g}})]^{-1}\text{avg}(\boldsymbol{y}(\boldsymbol{n}))$, where $\text{avg}(\boldsymbol{y}(\boldsymbol{n}))$ is the trichromatic average of the observed image. We apply the same interpretation to grey-edge, and learn the vector $\hat{\boldsymbol{g}}$ for each algorithm from a training set of images as the mean of the unit-vectors corresponding to the pixel or edge averages of each image. We use three-fold cross validation for this training, and to pick the optimal smoothing and norm parameters for grey-edge as described in [70].

3.7.2 Results

We report the mean and median errors for all algorithms, as well as the "Worst-25%" error which is a measure of robustness and refers to the mean of the 25% highest error values (note

Table 3.1: Angular error quantiles on all 568 images from the "Color Checker" database [55]

Method	Mean	Median	Worst 25%
Grey-world [50]	4.6°	4.1°	8.7°
Grey-edge (1^{st} Order) [56]	4.1°	3.5°	8.0°
Grey-edge (2^{nd} Order) [56]	4.0°	3.4°	7.8°
Gamut-mapping (σ = 5) [52]	4.1°	2.5°	10.3°
Generalized-gamut (σ = 5,*1-jet*) [57]	4.1°	2.5°	10.3°
Rosenberg [55]	4.8°	3.5°	10.5°
Proposed (ML Estimate)	3.7°	3.0°	7.6°
Proposed (with general Prior)	3.6°	3.0°	7.4°
Proposed (with category-wise Prior)	3.1°	2.3°	6.5°

Table 3.2: Angular error quantiles on 246 indoor images from the "Color Checker" database [55]

Method	Mean	Median	Worst 25%
Grey-world [50]	5.8°	5.5°	9.9°
Grey-edge (1^{st} Order) [56]	4.8°	4.1°	8.8°
Grey-edge (2^{nd} Order) [56]	4.7°	4.0°	8.5°
Gamut-mapping (σ = 5) [52]	5.5°	4.3°	12.3°
Generalized-gamut (σ = 5,*1-jet*) [57]	5.5°	4.4°	12.4°
Rosenberg [55]	6.5°	5.9°	12.0°
Proposed (ML Estimate)	4.2°	3.6°	8.1°
Proposed (with general Prior)	4.2°	3.6°	7.9°
Proposed (with category-wise Prior)	4.1°	3.7°	7.8°

that these may correspond to different images for different methods). Table 3.1 shows quantiles for the entire database, while Tables 3.2 and 3.3 report values separately for the indoor and outdoor sets respectively. Additionally, Figures 3.7 and 3.8 show examples of indoor and outdoor images respectively, corrected according to the illuminant estimates of various methods.

We first look at the relative performance of the ML estimator, without any prior information. Over the entire database, we find that it has the lower mean and worst 25% error values

Table 3.3: Angular error quantiles on 322 outdoor images from the "Color Checker" database [55]

Method	Mean	Median	Worst 25%
Grey-world [50]	3.7°	3.1°	7.1°
Grey-edge (1^{st} Order) [56]	3.5°	2.9°	7.1°
Grey-edge (2^{nd} Order) [56]	3.5°	2.8°	7.1°
Gamut-mapping (σ = 5) [52]	3.1°	1.8°	7.9°
Generalized-gamut (σ = 5,*1-jet*) [57]	3.1°	1.8°	7.9°
Rosenberg [55]	3.5°	2.4°	7.9°
Proposed (ML Estimate)	3.3°	2.5°	7.1°
Proposed (with general Prior)	3.2°	2.4°	6.9°
Proposed (with category-wise Prior)	2.3°	1.9°	4.6°

amongst all methods. The gamut-based methods have lower median errors, largely due to their superior performance on the outdoor images that were captured in daylight. They have the lowest mean and median errors for those images, although the proposed ML estimator appears to be more robust with a lower worst 25% error. However, the gamut-based methods perform worse on indoor images and have the highest error quantiles on that set after Rosenberg. In general, all algorithms show poorer performance on the indoor set than on outdoor images, indicating that indoor scenes are more likely to contain regions that act as outliers. However, the ML estimator performs noticeably better than the state of the art, with the lowest values for all error quantiles on indoor images.

Next, we note that an illuminant prior, even a general one over all images, leads to an improvement in overall performance. The most significant improvement is to the worst 25% quantile for all sets, indicating that the prior's major contribution is in making the estimation process more robust. It is worth noting here that while the performance improves overall, estimation errors on individual images may increase as is the case for some examples in Figs. 3.7-3.8.

When the illuminant prior is defined separately over sets of indoor and outdoor images (and it is possible that these labels may be obtained during estimation from a light-meter on the camera or a scene-classification algorithm), the improvement in overall performance is more dra-

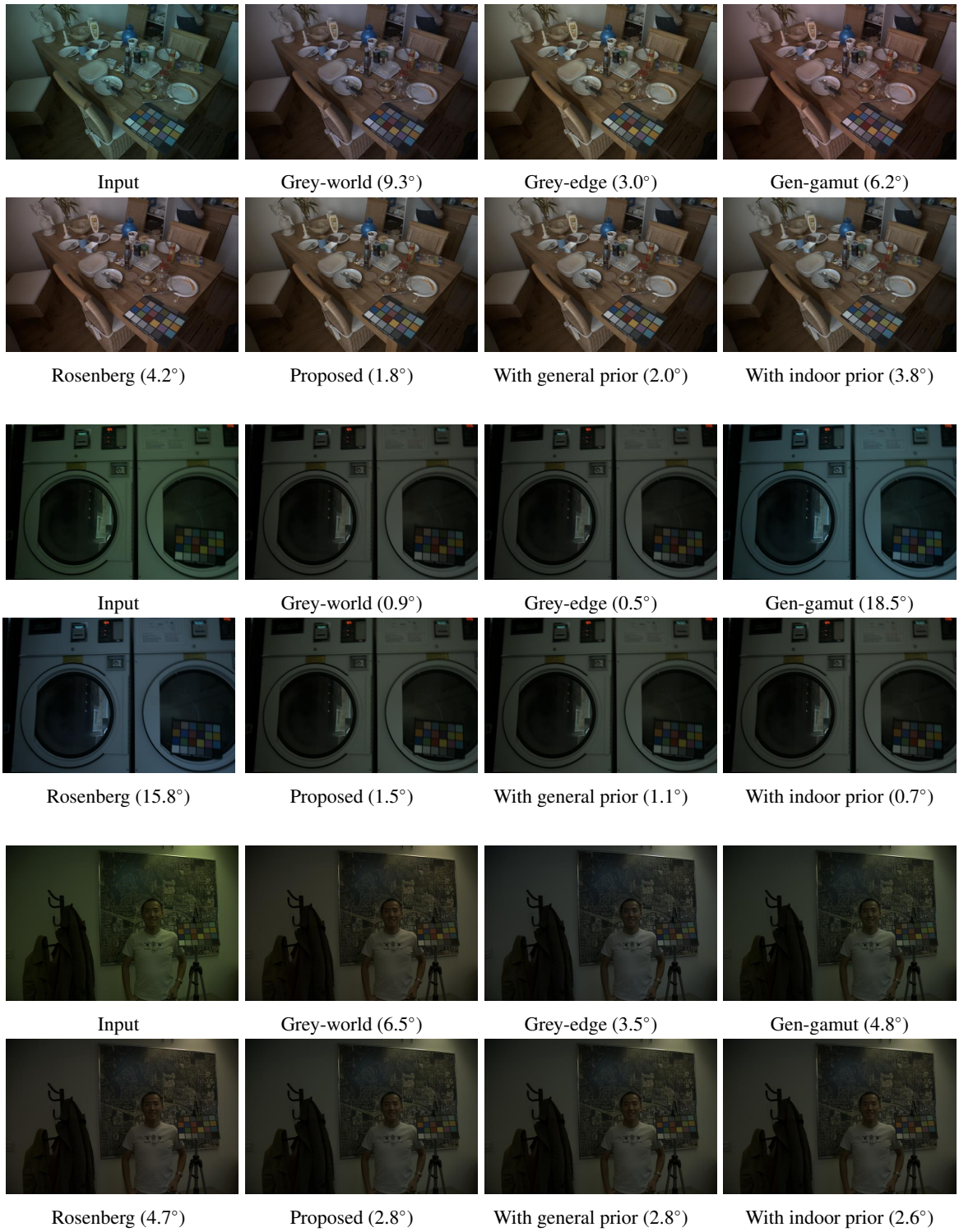


Figure 3.7: Indoor images from the color checker database, corrected using different algorithms. Angular errors for estimated illuminant are indicated below each image.

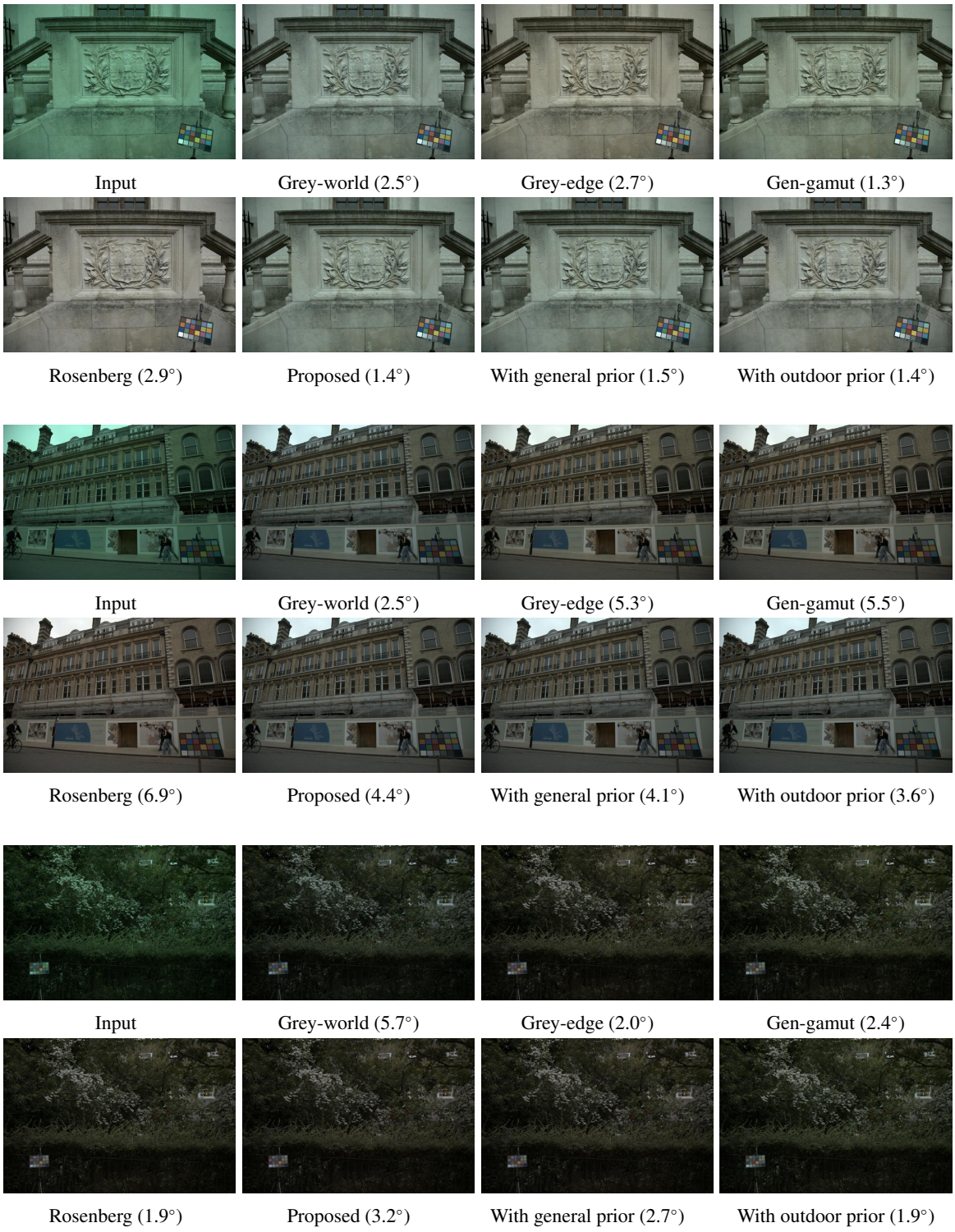


Figure 3.8: Outdoor images from the color checker database, corrected using different algorithms. Angular errors for estimated illuminant are indicated below each image.

Table 3.4: Performance of grey-edge and proposed method on the "Color Checker" database [55], with both methods using second-derivative Gaussian filters at $\sigma = 1$.

	Mean	Median	Worst 25%
Grey-edge	4.0°	3.4°	7.8°
Proposed (ML Estimate)	3.9°	3.2°	7.7°
Proposed (with gen. prior)	3.7°	3.0°	7.5°

matic. However, we note that most of this improvement is on the outdoor images, which is likely because of relatively less variability in the set of daylight illuminants.

While the spatio-spectral model allows a natural way to combine cues across sub-bands at multiple scales, we seek to gauge how well the proposed estimator succeeds at exploiting the information in each sub-band. Therefore, we perform a more direct comparison of our method to the grey-edge algorithm, by restricting the former to work on the same set of coefficients as the latter. Table 3.4 shows the error quantiles of the two methods (over all images) when both use second-derivative Gaussian filters at scale $\sigma = 1$, which corresponds to the best choice for grey-edge. We find that the proposed method yields better estimates even when not using an illuminant prior, and appears to benefit from accurately modeling the statistical structure of sub-band coefficients.

3.8 Discussion

In this chapter, we described a computational color constancy algorithm based on a model that effectively leveraged the spatial dependencies among pixels in a color image. We first decomposed the image using a set of spatially-decorrelating filters, and then analyzed the statistics of their coefficients. We found that these color coefficients are "better-behaved" than colors of individual pixels, and enable color constancy techniques that are quite accurate and efficient. We described two approaches for estimation: a maximum likelihood framework for when no prior information about the illuminant is available; and for cases when one knows statistics about likely illuminants a-priori, we introduced an illuminant model that can be incorporated during estimation for increased

robustness.

While we confined the bulk of our discussion to computational color constancy, work in this domain has often been motivated by and been an attempt to explain the way in which the human visual system achieves color constancy [71, 72]. Therefore, the relative success of our method naturally raises questions about whether the human visual system employs related processing for adaptation and color constancy. Indeed, psychophysical experiments have shown a strong interaction between the spatial orientation and frequency of a stimulus and the chromatic adaptation it induces [73, 74]. The mechanisms that govern these interactions are however poorly understood [73], and the machinery developed in this paper might be useful in creating experiments to analyze them further. Separate experiments have linked the textures of familiar objects to human color perception [75, 76], and while texture is currently thought to affect color perception through object memory, it is worth exploring the contribution of spatial correlations to this effect.

On the computational side, interesting avenues for future research include relaxing the two main assumptions made in most color constancy algorithms discussed in this chapter: spatially-constant illuminant spectrum and diagonal transforms for chromatic adaptation. Even when there is a single illuminant in the scene, mutual illumination causes the "effective" illuminant spectrum to vary from point to point. A first step that could lead to improvement would be to remove regions that appear to be outliers to the general statistics of the image, in the hope that these are likely to correspond to areas with significant inter-reflections. Alternatively, one could explore variants of the proposed method that assume the scene to be lit by a mixture of n illuminants. The problem would then be to estimate the colors of each of these illuminants, and their relative contributions at each pixel in the image.

The problem of moving beyond diagonal transforms is interesting as well. One path is to estimate general linear transforms under appropriate constraints. Another is to consider a set of registered training images taken under different illuminants, and then "learn" the functional form of the map between corresponding pixels. The parameters of this map can then be estimated from a

test image to do color constancy.

As was mentioned in Sec. 3.1, one of the reasons we seek color constancy is to be able to use color as a stable descriptor. A robust illuminant estimation framework allows us to now engage the problem of using color reliably in recognition and other visual tasks. Preliminary work by Owens *et al.* [77] shows that object matching across multiple scenes and illuminant estimation can be done jointly. However, it remains a challenge to be able to do color constancy with images captured using consumer digital cameras that do not provide linear (RAW) image data. In this work and in most color constancy algorithms, it is assumed that training and testing are performed with linear images taken from the same or similar cameras. This is important, because as shown in [78], the color spaces and non-linear processing done in cameras can vary significantly, affecting both the sub-band statistics in $\{\boldsymbol{\Sigma}_k\}$ and the illuminant statistics in $\boldsymbol{Q}$. Saenko *et al.* [79] look at the problem of adapting visual category models learned from one camera to another. It would be interesting to investigate whether the same can be done for color constancy in a way that would allow training data from one camera or linear images to be adapted to a new domain, with minimal additional information.

We end this discussion by noting that the model introduced in this chapter sought to encode spatio-spectral dependencies in trichromatic images, which make just three spectral measurements of the incident radiance at each pixel. In the next chapter, we study and model these dependencies in hyperspectral images, *i.e.* those that have a higher *spectral resolution* and therefore carry a far richer description of color.

Appendix: Additional Results

In this appendix, we report performance on another database considered in [70]. The "Grey Ball" database[80] is a collection of 11355 images captured using a video camera. A diffuse grey sphere is attached to the camera and is present at the same location in all images, serving

Table 3.5: Performance of various methods on "Grey Ball" set [80]

Method	Mean	Median	Worst 25%
Grey World [50]	12.2°	9.7°	25.4°
Grey Edge (1^{st} Order) [56]	10.8°	9.2°	20.9°
Grey Edge (2^{nd} Order) [56]	11.4°	9.9°	21.5°
Gamut Mapping [52]	11.8°	8.9°	24.9°
Generalized Gamut Mapping [57]	11.8°	8.9°	24.9°
Proposed (ML Estimate)	10.3°	8.9°	20.3°

as ground truth. It is pioneering in being the first large database of real world images captured to evaluate color constancy methods. However, the database is highly correlated with consecutive frames often having very little change in scene content and illumination. Furthermore, the ground truth information is unreliable in some cases since the grey ball is close to the camera and may be lit by a different illuminant than the scene being captured (the color charts in the database in [55] are placed within the scene to avoid this). As such, the error values on this database need to be interpreted with caution.

The original database includes gamma-corrected (sRGB) images, and we follow the procedure in [70] by measuring performance after applying approximate inverse gamma-correction (by raising all intensity values to the power 2.2) to the images and recomputing the ground truth accordingly. We also adopt the approach suggested in [70] for cross-validation to avoid correlated images appearing in the training and testing sets: since the database is divided into 15 videos, we carry out estimation on the frames of each video using using the remaining videos as training.

Table 3.5 shows error quantiles of the proposed method and compares them to those of other algorithms. We use the evaluation results from [70] as an indicator of the performance of state-of-the-art methods, but again, we differ in applying the learning step to grey-world and grey-edge as described in Sec. 3.7.1. We only report errors for the ML estimator, since the prior-based estimates have identical error quantiles. Due to lower resolution, compression artifacts, and an approximate

gamma compensation, this database proves to be significantly more challenging with all algorithms showing larger error values than for the color checker database in Sec. 3.7.2 (although, this could partly be due to the unreliability of ground truth as discussed above). Nevertheless, we find that the proposed method yields more accurate estimates than other algorithms, with the lowest values for all quantiles.

4

Hyperspectral Statistics

"We never really perceive what color is physically."

— Josef Albers

In this chapter, we explore image statistics in a new domain. Hyperspectral images provide higher spectral resolution than typical RGB images by including a large number of irradiance measurements in the visible spectrum, at every pixel. This additional spectral resolution is potentially useful for visual tasks, such as segmentation, recognition, and relighting. We present a new database of fifty hyperspectral images of natural indoor and outdoor scenes, and use this database to explore hyperspectral statistics. Vision systems that seek to capture and exploit hyperspectral data should benefit from statistical models of natural hyperspectral images. Additionally, these models provide greater access (than models in RGB) to information about the spatio-spectral structure of scene radiance. We derive an optimized joint spatio-spectral basis for representing hyperspectral image patches. Then, we explore the statistical properties of the coefficients in this basis, and investigate the dependencies between different coefficients.

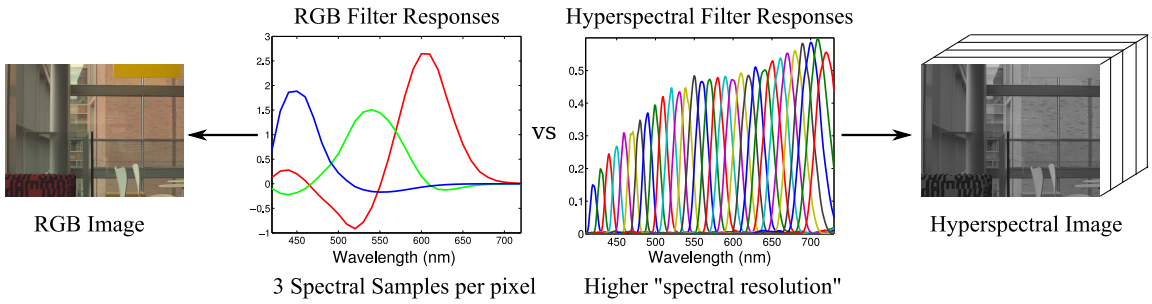


Figure 4.1: Typical digital cameras (Left) make just three measurements of the incident radiance, to yield trichromatic images with three recorded values at each pixel (*eg.* RGB). Hyperspectral cameras (Right) seek higher spectral resolution and make a larger number of measurements at each pixel (*eg.* corresponding to multiple narrow wavelength-band filters in the visible spectrum).

4.1 Introduction

Most cameras are concerned with capturing images and videos for display to humans, and therefore capture three spectral measurements (red, green, blue) to match human trichromacy. However, the spectral power density of the incident visible radiance is a continuous function of wavelength which carries information beyond these measurements. Since this information can be of use to vision systems, researchers have looked at acquiring and processing "hyperspectral" images, meaning those that provide a dense spectral sampling at each pixel (see Fig. 4.1). These images have proven useful in many domains, including remote sensing [81–85], medical diagnosis [86–88], and biometrics [89], and it seems likely that they will be beneficial for inference tasks on everyday scenes as well.

In this chapter, we establish the basic statistical structure of hyperspectral images of "real-world" scenes, such as offices, streetscapes, and parks, that we encounter in everyday life. Such a characterization gives us a more complete understanding of the spatio-spectral structure of scene radiance, than comparable analysis of RGB data. Furthermore, when developing vision systems that acquire and exploit hyperspectral imagery, we can benefit from knowledge of the underlying statistical structure. By modeling the inter-dependencies that exist in the joint spatio-spectral domain, we should be able to build, for example, more efficient systems for capturing hyperspectral

images and videos, that are able yield accurate reconstructions from fewer measurements and are able to deal with noise and other imperfections in the capture process. Furthermore, it is important to understand these dependencies when developing algorithms for visual inference tasks, such as segmentation and recognition.

While previous analyses have separately considered the spectral statistics of point samples [90–92], we consider the spatial and hyperspectral dimensions *jointly* to uncover additional structure. Using a new collection of fifty hyperspectral images captured with a time-multiplexed 31-channel camera, we evaluate difference choices of spatio-spectral bases for representing hyperspectral image patches and find that a separable basis is appropriate. Then, we characterize the statistical properties of the coefficients in this basis and describe models that capture these properties effectively.

4.2 Related Work

Our work is motivated by successes in analyzing and modeling the statistical properties of greyscale [9, 93, 94] and trichromatic [95–99] images. In the previous chapters, we have seen how such models can enable accurate visual inference— for estimating motion blur and illuminant spectra. But image statistics have found applications in a variety of domains. They have proven useful for inferring accurate images from noisy and incomplete measurements, with applications in denoising [13, 100], restoration [6, 101] and demosaicking [15, 58], and have also use as building blocks for higher-level visual tasks such as segmentation and object detection [102–104]. Our goal here is to develop comparable models for hyperspectral data, by considering the joint statistics of variations with respect to space and wavelength.

The study in this chapter is enabled by recent advances in hyperspectral capture systems, such as those based on spatial-multiplexing with generalized color filter arrays [105], spatial-multiplexing with a prism [106], time-multiplexing with liquid crystal tunable filters [107, 108],

and time-multiplexing with varying illumination [109, 110]. Prior to these advances, studies of real-world spectra have been limited to collections of point samples, such as those collected by a spectrometer. These studies have suggested, for example, that the spectral reflectances of "real-world" materials are smooth functions that can be represented with 6-8 principal components [91, 92, 96] (or a suitable sparse code [90]), and that the spectra of daylight and other natural illuminants can be represented with even fewer principal components [111]. One of our goals in this study is to move beyond point samples, and to investigate the properties of variations in spectral distributions within spatial neighborhoods.

We expect that accurate statistical models will aid in the design of efficient hyperspectral acquisition systems. Many proposed acquisition methods seek to reconstruct full spectral images from a reduced set of measurements based on assumptions about the underlying statistics [105, 110]. Such methods are likely to benefit from accurate statistical models that are learned from real-world hyperspectral data. These models may also prove useful for other applications, such as relighting, segmentation, and recognition.

Other hyperspectral datasets that are related to the one introduced here include those of Hordley *et al.* [108] and Yasuma *et al.* [105]. These datasets include 22 and 32 hyperspectral images, respectively, and they are focused on objects captured with controlled illuminants in laboratory environments. More related is the database of 25 hyperspectral images of outdoor urban and rural scenes captured by Foster *et al.* [107]. A primary aim of our work has been to capture and analyze a larger database that includes both indoor and outdoor scenes.

4.3 Hyperspectral Image Database

To enable an empirical analysis of the joint spatio-spectral statistics of real-world hyperspectral scenes, a commercial hyperspectral camera (Nuance FX, CRI Inc.) was used to collect a database of fifty images under daylight illumination, both outdoors and indoors. As illustrated in

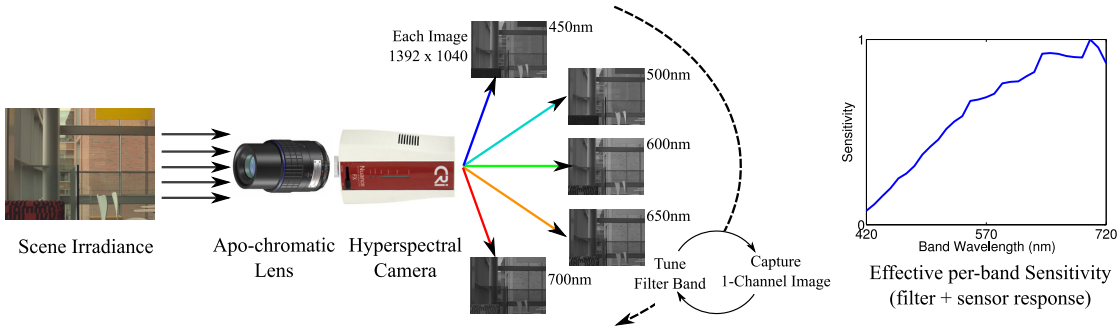


Figure 4.2: Commercial hyperspectral camera (Nuance FX, CRI Inc.) with apo-chromatic lens used to capture images in the database. The camera is equipped with a greyscale sensor and a liquid crystal tunable filter that can be tuned to have 10nm wavelength pass bands centered at steps on 10nm (see Fig. 4.1). (Left) A 31-channel hyperspectral image is captured through time-multiplexed measurements by tuning the filter sequentially to each of these bands and capturing an image stack. (Right) The effective per-band sensitivity is a combination of the filter transmittance and wavelength-dependent sensor efficiency.

Fig. 4.2, the camera uses an integrated liquid crystal tunable filter and is capable of acquiring a hyperspectral image by sequentially tuning the filter through a series of thirty-one narrow wavelength bands, each with approximately 10nm bandwidth and centered at steps of 10nm from 420nm to 720nm. Figure 4.2 (right) shows the relative sensitivity of the camera for each wavelength band, accounting for both the quantum-efficiency of the 12-bit greyscale sensor and the per-band transmittance of the effective filters. The camera is equipped with an apo-chromatic lens (CoastalOpt UV-VIS-IR 60mm Apo Macro, Jenoptik Optical Systems, Inc.) and in all cases we used the smallest viable aperture setting. The combination of the apo-chromatic lens and the avoidance of a mechanical filter wheel allows us to acquire images that are largely void of chromatic aberration and mis-alignment. To avoid contaminating the statistics by having different per-band noise levels, we did not vary the exposure time across bands or normalize the captured bands with respect to sensitivity. Therefore, all results in this chapter must be interpreted relative to the camera sensitivity function, with the exception of Sec. 4.4.2 which discusses "camera-independent" statistics (after making specific assumptions).

Due to the use of small apertures and the low transmittance of individual bands, the total

acquisition times for an entire image (*i.e.*, all wavelength bands) are high and vary from fifteen seconds to over a minute. Accordingly, all images were captured using a tripod and by ensuring minimal movement in the scene. Since having a perfectly static scene is often not feasible, in the interest of having a diverse dataset, we have captured images with movement in some regions— but these regions (and other areas affected by dust, *etc.*) are masked out manually before analysis. We note that as a result, any regions with people in the captured scenes are masked out, and our analysis does not include samples of human skin tones.

The captured dataset includes images of both indoor and outdoor scenes featuring a diversity of objects, materials and scale (see Fig. 4.3 for a few example images rendered in sRGB). We have also captured twenty-five additional images taken under artificial and mixed illumination, and while these are not used for the analysis presented in this work, they have been made available to researchers along with the fifty natural illumination images [24]. We believe the database to be a representative sample of real-world images, capturing both pixel-level material statistics and spatial interactions induced by texture and shading effects. In addition to the analysis here, these images may be useful "ground truth" to design and evaluate methods for various acquisition and vision tasks that use hyperspectral data. They can also be used to synthetically generate ground truth RGB images that correspond to true trichromatic measurements at each pixel, rather than those estimated through demosaicking.

4.4 Spatio-Spectral Representation

We begin by exploring efficient representations for hyperspectral images. As is common practice with greyscale and RGB images, we first divide the entire image into patches and consider the properties of each patch independently. Let $X[\boldsymbol{n}, l]$ be a random $P \times P$ hyperspectral image patch, where $\boldsymbol{n} \in \{1, \ldots P\}^2$ and $l \in \{1, \ldots 31\}$ index pixel location and spectral band respectively. For the rest of this chapter, we choose the patch size $P = 8$, but the results and conclusions from

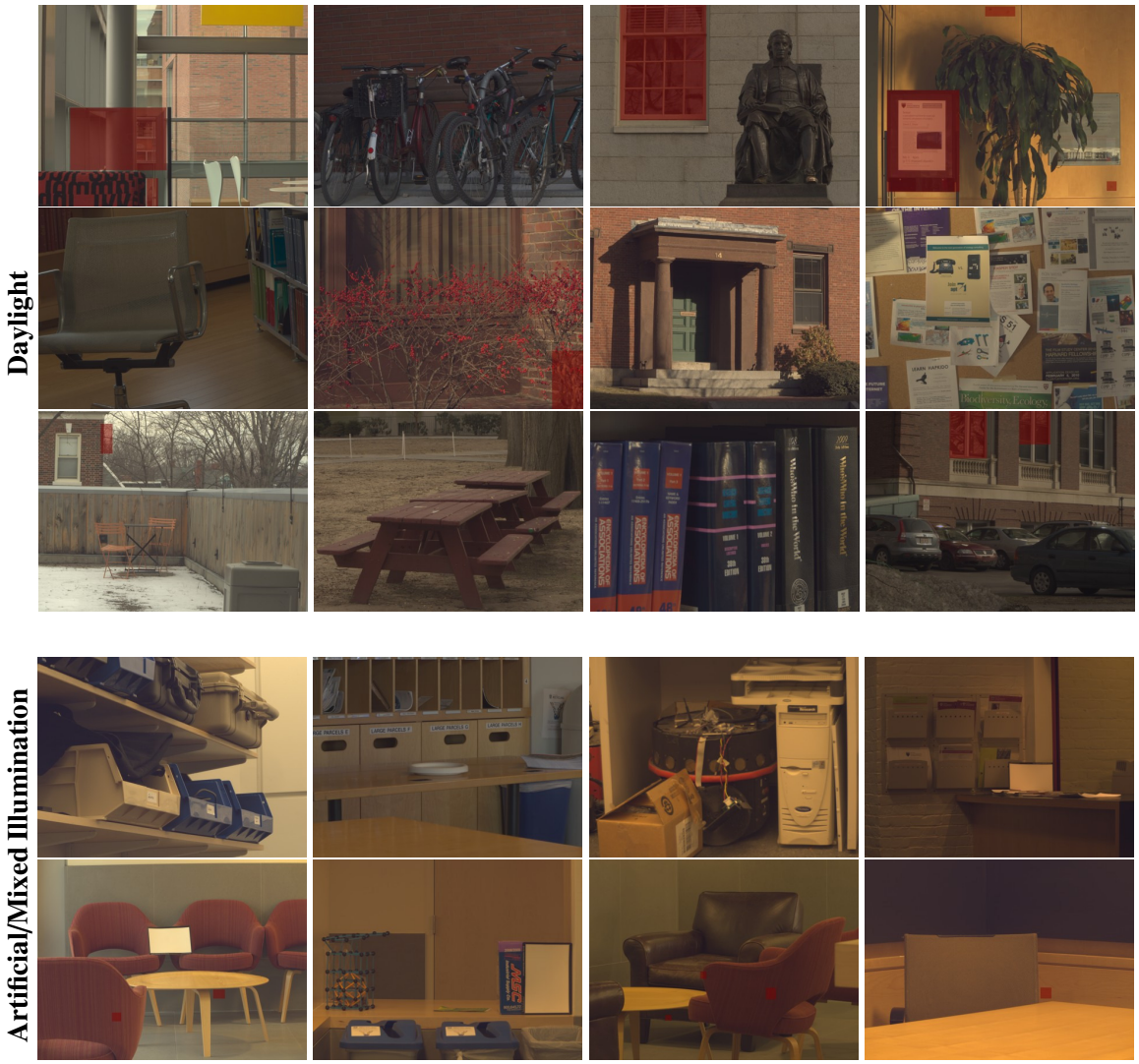


Figure 4.3: Example scenes from the captured database, rendered here in sRGB. The database contains a diverse set of images of outdoor and indoor scenes, with fifty images illuminated by daylight and an additional twenty-five under artificial and mixed illumination. Regions with unreliable data (due to movement during exposure, *etc.*) were labeled manually, and are shown here with red masks.

different choices of P are qualitatively the same.

Since X is high-dimensional, we seek a representation that allows analysis in terms of a smaller number of components. Formally, we wish to find an optimal orthonormal basis set $\{V_i\}$ and express X in terms of scalar coefficients x_i as

$$X[\boldsymbol{n}, l] = \mu[\boldsymbol{n}, l] + \sum_i x_i V_i[\boldsymbol{n}, l], \tag{4.1}$$

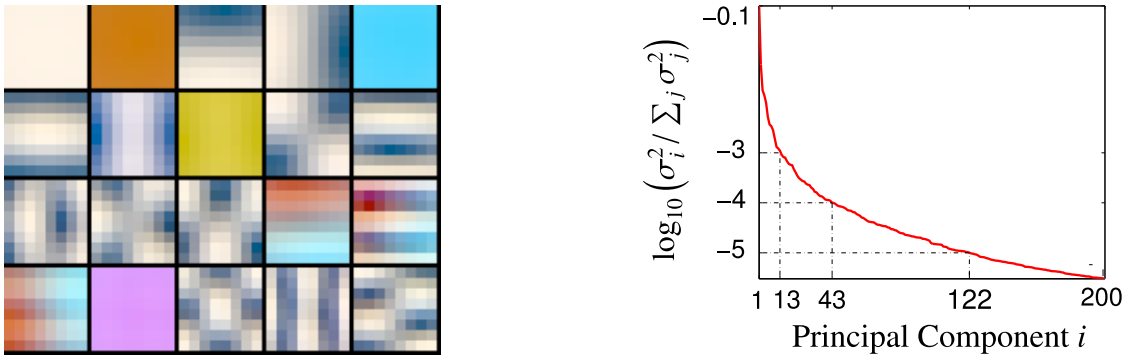


Figure 4.4: General basis for 8×8 hyper-spectral patches learned across the database. Left: most significant basis vectors (in reading order) rendered in RGB. Right: normalized variance of the coefficients for the first 200 basis vectors. The variance decays rapidly indicating that a small proportion of components are sufficient for accurate reconstruction.

where $\mu[\boldsymbol{n}, l]$ is the "mean patch" and

$$x_i = \langle X - \mu, V_i \rangle_{\boldsymbol{n},\, l} = \sum_{\boldsymbol{n},l} V_i[\boldsymbol{n}, l]\, (X[\boldsymbol{n}, l] - \mu[\boldsymbol{n}, l]). \tag{4.2}$$

We learn a set of general basis vectors using principal component analysis (PCA) on patches cropped from images in the database, through eigen-decomposition of their empirical covariance matrix. This corresponds to choosing a basis that ensures maximal "energy compaction", *i.e.* the set of basis vectors is chosen such that every truncated sub-set $\{V_{i'}\}_{i'=1}^{i}$ minimizes the expected "reconstruction error" from that subset, defined as

$$\epsilon(i) = \mathbb{E}\left\| \left(\mu[\boldsymbol{n}, l] + \sum_{i'=1}^{i} x_{i'} V_{i'}[\boldsymbol{n}, l] \right) - X[\boldsymbol{n}, l] \right\|^2. \tag{4.3}$$

Figure 4.4 shows the top twenty components rendered in RGB, as well as the variance for the top 200 components. We see that the first two V_i essentially correspond to spatially-constant "DC" components with distinct spectra, followed by vertical and horizontal derivative components. We also find that there is a steep fall off in variance indicating that X can be described accurately by a relatively small number of coefficients. Indeed, the first 20 basis vectors (out of a total of 1984) account for 99% of the total variance.

4.4.1 Separable Basis Components

Notice that the basis components in Fig. 4.4 has sets of vectors with similar spatial patterns but different spectra or "colors". Therefore, we next investigate the efficiency of a separable basis $\{V_i\}$, that is formed as a Cartesian product of two orthonormal basis sets $\{S_j[\boldsymbol{n}]\}_{j=1}^{P^2}$ and $\{C_k[l]\}_{k=1}^{31}$, that span the space of monochrome $P \times P$ spatial patches and the space of 31-channel spectral distributions, respectively. Every V_i can therefore be expressed as $V_i[\boldsymbol{n}, l] = V_{jk}[\boldsymbol{n}, l] = S_j[\boldsymbol{n}]C_k[l]$, for some values of j and k, and we denote the coefficient along V_{jk} as x_{jk}, where $x_{jk} = \left\langle X - \mu, S_j C_k \right\rangle_{\boldsymbol{n},\, l}$. Note that by construction, the vectors $\{V_{jk}\}$ are orthonormal since

$$\begin{aligned}\left\langle V_{jk}, V_{j'k'} \right\rangle_{\boldsymbol{n},\, l} &= \sum_{\boldsymbol{n},l} S_j[\boldsymbol{n}]C_k[l]\, S_{j'}[\boldsymbol{n}]C_{k'}[l] = \left(\sum_{\boldsymbol{n}} S_j[\boldsymbol{n}]S_{j'}[\boldsymbol{n}]\right)\left(\sum_{l} C_k[l]C_{k'}[l]\right) \\ &= \left\langle S_j, S_{j'} \right\rangle_{\boldsymbol{n}} \langle C_k, C_{k'} \rangle_l = \begin{cases} 1 & \text{if } j = j', k = k', \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{4.4}$$

As before, we use a training set to learn the optimal separable basis $\{V_{jk}\}$, which corresponds to learning the optimal spatial and spectral components $\{S_j\}$ and $\{C_k\}$. Let us assume that we know the optimal spatial basis, and only have to estimate the spectral components $\{C_k\}$. For energy compaction, we wish to choose these components such that for every k, the reconstruction error, when using components composed of all $\{S_j\}$ and the truncated set $\{C_{k'}\}_{k'=1}^{k}$, is minimized. This error is given by

$$\begin{aligned}\epsilon^c(k) &= \mathbb{E}\left\| \left(\mu[\boldsymbol{n}, l] + \sum_{j=1}^{P^2}\sum_{k'=1}^{k} x_{jk'} S_j[\boldsymbol{n}]C_{k'}[l]\right) - X[\boldsymbol{n}, l]\right\|^2 \\ &= \mathbb{E}\|X - \mu\|^2 - \sum_{j}\sum_{k'=1}^{k} \mathbb{E}\, \left\langle X - \mu, S_j C_{k'} \right\rangle_{\boldsymbol{n},\, l}^2 \\ &= \mathbb{E}\|X - \mu\|^2 - \sum_{\boldsymbol{n}}\sum_{k'=1}^{k} \mathbb{E}\, \langle X - \mu, C_{k'} \rangle_l^2 \,. \end{aligned} \tag{4.5}$$

We note the reconstruction error is independent of the choice of $\{S_j\}$, and therefore the optimal spectral components $\{C_k\}$ can be computed using PCA on the spectral distributions of individual pixels. This reasoning also holds when computing the optimal spatial basis $\{S_j\}$, which can

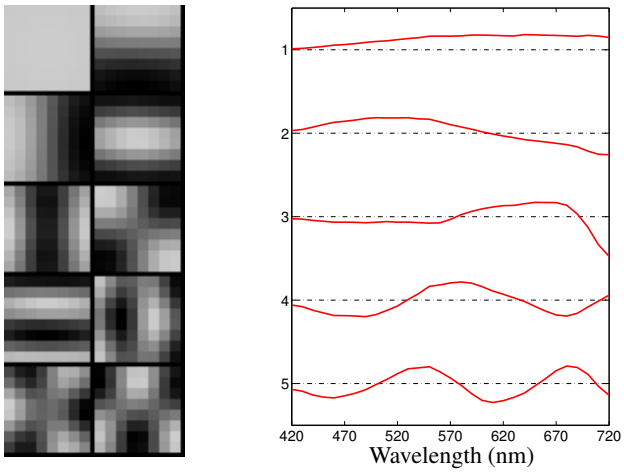


Figure 4.5: Learned separable basis, with most significant (Left) spatial components $\{S_j[\boldsymbol{n}]\}$, and (Right) spectral components $\{C_k[l]\}$. The overall separable basis is constituted by taking a Cartesian product of these two sets, *i.e.* by forming basis vectors as $S_j[\boldsymbol{n}]C_k[l]$, for ever pair (j,k).

be learned through PCA on monochrome patches pooled across all bands. Figure 4.5 shows the first few spatial and spectral components, $\{S_j\}$ and $\{C_k\}$, learned in this manner from the database. The spatial components correspond to a DCT-like basis used commonly for modeling greyscale images, with S_1 corresponding to the "DC" component. The spectral components in turn resemble a Fourier basis scaled by the camera's sensitivity function (see Fig. 4.2), where C_1 can be loosely interpreted as the average brightness or "luminance".

Given the individual spatial and spectral bases, we can compute the energy or variance for every combination (*i.e.* $\mathbb{E}\ x_{jk}^2$) and sort these combinations according to this variance. This establishes a unique map from every pair (j,k) to an index $i \in \{1,\ldots P^2 \times 31\}$, such that the overall reconstruction error $\epsilon(i)$ is minimized. In Fig. 4.6, we compare this reconstruction error to that of the general joint basis. We find that the two have near-identical reconstruction errors for the same number of components, indicating that separable basis is equally efficient. A wavelet-based separable basis, with the spatial components $\{S_j\}$ corresponding to Haar wavelets, is also included for comparison.

Figure 4.7 shows the map from the pairs (j,k) to index i (truncated to $i \leq 15$ for clarity),

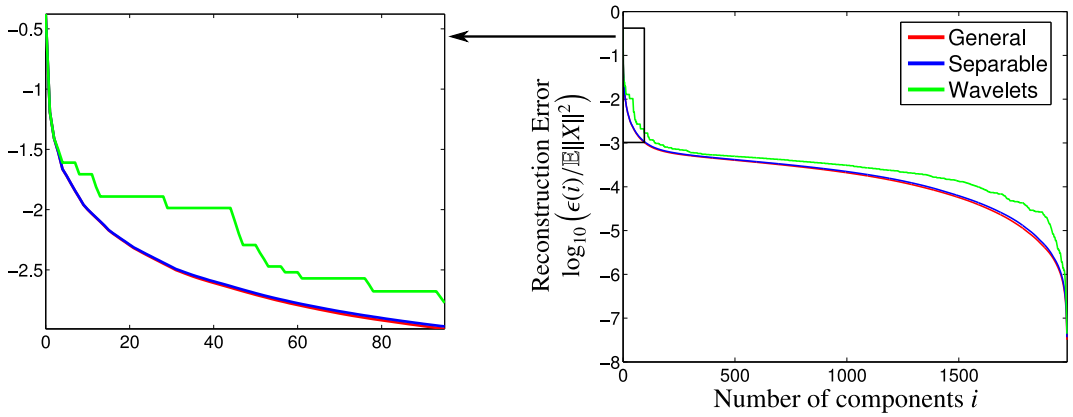


Figure 4.6: Comparison of different spatio-spectral basis sets in terms relative reconstruction error using a limited number of components. The figure compares a general basis set to one restricted to having separable spatial and spectral components. The separable basis has near identical reconstruction error to the general basis, indicating that it is equally efficient. The efficiency of a separable set with Harr wavelets as the spatial basis is also shown.

and illustrates the relative importance of different combinations of the spatial and spectral basis vectors in Fig. 4.5. We see that the combinations of the first spectral component (*i.e.* the luminance) with various spatial components typically have higher variance than vectors involving higher spectral components. Fig. 4.8 provides another look at the variance in this separable basis. For each of the top spatial components, it plots the variance for their combination with each of the top spectral components. We find that the these curves show similar decays along the spectral dimension, indicating that the total variance in the different spatial components is distributed in similar proportions amongst their spectral coefficients.

4.4.2 Camera-independent Basis

So far, we have looked at representations for patches relative to the camera's sensitivity function shown in Fig. 4.2. Since this function is known, it is possible to compute the corresponding statistics for hyperspectral images captured by a different device with a different sensitivity, after making appropriate assumptions about the observation noise in the database. As a specific case of this, we look at properties of images captured by a hypothetical camera that has a flat sensitivity

Spatial index j

Spectral index k	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
1	1	3	4	6	7	9	10	11	12	13	15
2	2	14									
3	5										
4	8										

Figure 4.7: Relative importance of different combinations of the spatial and spectral components S_j and C_k, in terms of variance. Shown here is the map from every pair (j,k) to the index i of the corresponding joint basis vector V_{jk}. Note that combinations of the first spectral component C_1 (*i.e.* luminance) with various S_j often rank higher than the combinations of S_1 (*i.e.* DC) with the higher spectral components $C_k, k > 1$.

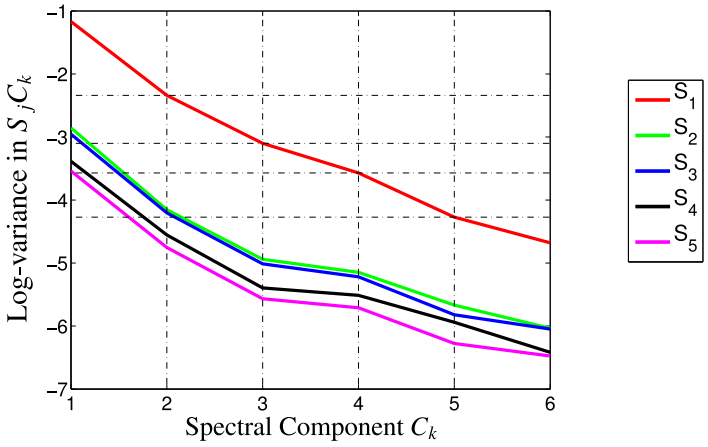


Figure 4.8: Variances in combinations with different spectral components, for the first five spatial components. The horizontal grid lines correspond to the values of the DC component S_1. Note that the different S_i have similar decays along the spectral dimension.

function. These can be interpreted as the properties of the underlying scene itself, without varying attenuation applied to the different wavelength bands.

Formally, we relate the captured hyperspectral patch $X[\boldsymbol{n}, l]$ to the true un-attenuated version $X_t[\boldsymbol{n}, l]$ as

$$X[\boldsymbol{n}, l] = s[l]X_t[\boldsymbol{n}, l] + z[\boldsymbol{n}, l], \tag{4.6}$$

where $s[l]$ is the known camera sensitivity, and $z[\cdot]$ is observation noise. Following the analysis

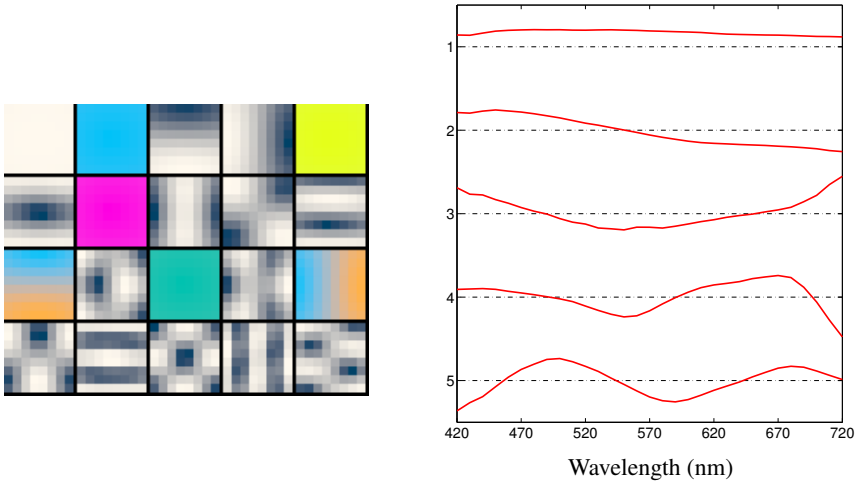


Figure 4.9: Basis vectors for hyperspectral patches from a camera with uniform sensitivity. Left: Most significant joint basis vectors, rendered in sRGB. Right: Spectral basis vectors $\{C_{tk}\}$ that, combined with the spatial vectors $\{S_j\}$ in Fig. 4.5, define an efficient separable basis.

above, we seek to find the optimal basis for X_t through an eigen-decomposition of the covariance matrix $\mathbb{E}X_tX_t^T$. We assume white Gaussian noise, $z[\boldsymbol{n}, l] \overset{\text{i.i.d}}{\sim} \mathcal{N}(0, \sigma_z^2)$, which gives us the following relation between the covariances of X and X_t:

$$\mathbb{E}X_t^2[\boldsymbol{n}, l] = s^{-2}[l]\left(\mathbb{E}X^2[\boldsymbol{n}, l] - \sigma_z^2\right),$$

$$\mathbb{E}X_t[\boldsymbol{n}, l]X_t[\boldsymbol{n}', l'] = s^{-1}[l]s^{-1}[l']\mathbb{E}X[\boldsymbol{n}, l]X[\boldsymbol{n}', l'], \quad \text{if } n \neq n' \text{ or } l \neq l'. \tag{4.7}$$

We use the values of $\mathbb{E}XX^T$ estimated from our database, and we set the noise variance σ_z^2 to be equal to half of its lowest eigen-value, *i.e.*, half the variance along the least significant basis vector. We can now compute the covariance matrix for X_t, and the optimal basis vectors thus obtained through PCA are shown in Fig. 4.9 (left).

Since X was shown in Sec. 4.4 to be represented efficiently using a separable basis and the camera sensitivity is the same for all pixels, it follows that the basis for X_t is also separable, and composed of the same spatial basis $\{S_j[\boldsymbol{n}]\}$ as for X and a spectral basis $\{C_{tk}[l]\}$ shown in Fig. 4.9 (right). As expected, from comparing Fig. 4.9 to Fig. 4.5, we find that spectral basis vectors $\{C_{tk}[l]\}$ for X_t represent an orthogonalized version of $\{s^{-1}[l]C_k[l]\}$.

We can also use our estimates of the covariance matrix of X_t to explore how efficient the human cone responses are at capturing the variance in the scenes in our database. We find that the sub-space spanned by the CIE XYZ vectors (designed to match the spectral response of human visual system) account for 77.22% of the total variance in X_t. In comparison, the first three eigenvectors $\{C_{tk}\}_{k=0}^{2}$ account for 99.14% of the total variation. However, it is important to remember that human cone responses are restricted to have non-negative responses at all wavelengths (as is true with any set of sensors). Also, the human visual system is likely to have evolved in different environments, and to be optimal for capturing spectral information helpful for specific discriminative tasks.

Note that the assumption made in this section about the noise being i.i.d Gaussian, and the manner in which its variance is chosen are approximations. Indeed, observation noise in typical camera sensors is known to be Poisson distributed [112]. Therefore, while this analysis is helpful in building intuition, we return to working with the camera-relative bases in subsequent sections.

4.5 Coefficient Models

Having identified a separable spatio-spectral basis we now explore statistical models for coefficients in this basis. We look at distributions for each coefficient individually, as well as joint models for different spectral coefficients along the same spatial basis.

4.5.1 Modeling Individual Coefficients

Let x_{jk} be the coefficient of X in the basis component $V_{jk} = S_j[\boldsymbol{n}]C_k[l]$. We begin by looking at empirical distributions of the "DC" coefficients x_{1k} in Fig. 4.10 (top). As was the case for per-pixel distributions in Chapter 3, we find that these distributions differ qualitatively from those of the other coefficients shown in Fig. 4.10 (bottom), and exhibit comparatively less structure. In applications with greyscale and RGB images, DC coefficients are found to be poorly described by

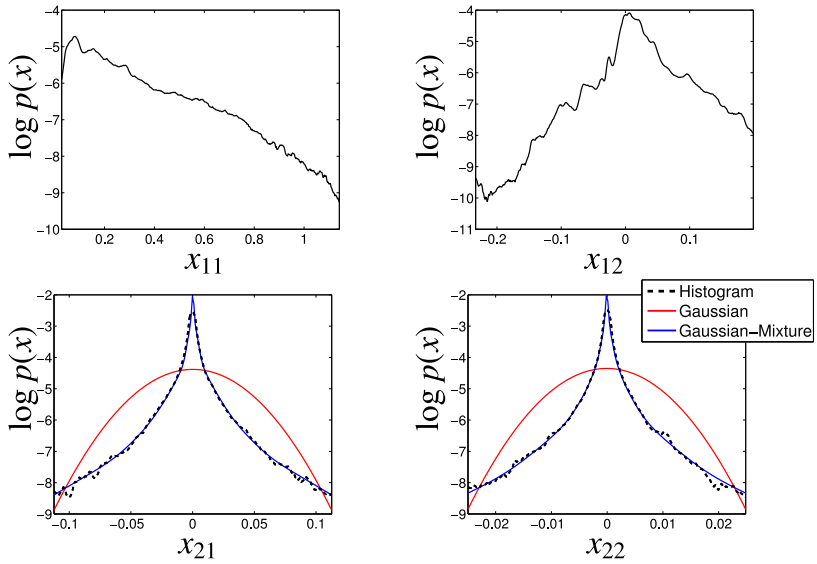


Figure 4.10: Empirical Histograms of coefficients in the separable basis. Histograms of (Top) DC coefficients corresponding to the first two spectral components (*i.e.* x_{11} and x_{12}); and (Bottom) coefficients along the second spatial component (x_{21} and x_{22}). The DC distributions are relatively unstructured compared to those for higher spatial components. The latter are symmetric and heavy-tailed (a Gaussian distribution with the same variance is shown for comparison), and can be accurately described by a Gaussian-mixture.

standard probability distributions and are often simply modeled as being uniform (for example, they were not used for inference in Chapters 2 and 3), and we do the same here.

The statistics of the higher spatial coefficients (x_{jk} for $j > 1$) are more interesting. Figure 4.10 (bottom) shows empirical distributions of x_{21} and x_{22} (the second spatial component). We see that these distributions are zero-mean, uni-modal, symmetric, and more kurtotic than a Gaussian distribution with the same variance, with heavier tails and a higher probability mass near zero. This matches intuition from greyscale and RGB image analysis that higher spatial sub-band coefficients are "sparse". While a variety of parametric forms could be used to model the distributions of these coefficients (generalized Gaussians, Gaussian Scale Mixtures, *etc.*), we choose to use a finite mixture of zero-mean Gaussians that has the advantage of allowing tractable inference, while also being able to accurately express a variety of distribution shapes with the appropriate choice of model parameters.

Formally, we define

$$p(x_{jk}) = \sum_{z=1}^{Z} p(z_{jk} = z)\mathcal{N}\left(x_{jk}|0, \sigma^2_{jk,z}\right), \tag{4.8}$$

where $z_{jk} \in \{1, \ldots Z\}$ is a latent index variable indicating that x_{jk} is distributed as a Gaussian with the corresponding variance $\sigma^2_{jk,z}$. Without loss of generality, we assume that the mixture components are sorted by increasing variance. The model parameters $\{p(z_{jk} = z), \sigma^2_{jk,z}\}_z$ are estimated from the database using Expectation Maximization (EM) [48], and in practice we find that mixtures of eight Gaussians (*i.e.* $Z = 8$) are able to fit the empirical distributions for all coefficients with reasonable accuracy. These fits for the coefficients x_{21} and x_{22} are shown superimposed on the empirical histograms in Fig. 4.10 (bottom).

4.5.2 Joint Models

Since the spatio-spectral basis vectors have been estimated through PCA, it follows that x_{jk} and $x_{jk'}$ will be uncorrelated for $k \neq k'$, *i.e.* $\mathbb{E}(x_{jk}x_{jk'}) = 0$. However, given the model for individual coefficients in (4.8), this does not necessarily imply that they will be independent. Indeed, different spatial coefficients at the same spatial location in greyscale images are known to be related [14]. We now demonstrate that different spectral coefficients along the same spatial basis are also mutually dependent, and propose a model that encodes these dependencies. This expands on our intuition in Chapter 3 that sub-band color coefficient vectors have ellipsoidal equi-probability contours.

We begin by examining whether knowing the value of the mixture index z_{jk} carries any information about the statistics of the coefficient $x_{jk'}$ for a different spectral component k' along the same spatial basis j, as illustrated in Fig. 4.11 (left). We define $\sigma^2_{jk'|z_{jk}}(z)$ to be the variance of $x_{jk'}$ conditioned on the mixture index z_{jk} being z, and estimate it from a set of training patches $\{X^i\}$ from the database, as

$$\sigma^2_{jk'|z_{jk}}(z) = \frac{\sum_i p(z_{jk} = z|x^i_{jk})(x^i_{jk'})^2}{\sum_i p(z_{jk} = z|x^i_{jk})}, \tag{4.9}$$

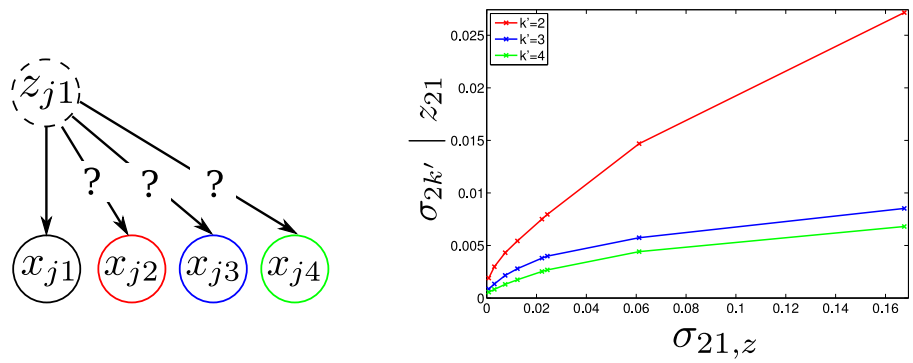


Figure 4.11: Test for coefficient independence. (Left) We seek to determine whether the index variable for one spectral coefficient has any effect on the statistics of other coefficients for the same spatial basis. (Right) Expected standard deviation of $x_{2k'}$ conditioned on the index variable z_{21}. We find that when x_{21} belongs to a mixture component having higher standard deviation $\sigma_{21,z}$ (horizontal axis), the other spectral components $x_{2k'}$ have higher standard deviations $\sigma_{2k'|z_{21}}(z)$ (vertical axis) as well. This implies that the different spectral coefficients are not independent, because if they were, these curves would be horizontal.

where $p(z_{jk} = z|x^i_{jk})$ is computed for every training coefficient as

$$p(z_{jk} = z|x^i_{jk}) = \frac{p(z_{jk} = z)\mathcal{N}(x^i_{jk}|0, \sigma^2_{jk,z})}{\sum_{z'} p(z_{jk} = z')\mathcal{N}(x^i_{jk}|0, \sigma^2_{jk,z'})}. \tag{4.10}$$

Figure 4.11 (right) shows these variances for different coefficients $x_{2k'}$ conditioned on the mixture index z_{21} for the first spectral coefficient, and compares them to the corresponding mixture component variances $\sigma^2_{21,z}$. If the coefficients were independent, the value of one index variable would have no effect on the variance of other coefficients. However, we see that variance $\sigma^2_{jk'|z_{jk}}$ does indeed vary with the value of z_{jk}, which implies that the different spectral coefficients are mutually dependent. Specifically, when the first spectral coefficient x_{21} belongs to a mixture component having higher variance, the expected variances of the other spectral coefficients $\{x_{2k'}\}$ increase as well.

To capture this relationship, we update the model in (4.8) by including a joint distribution $p(\{z_{jk}\}_k)$ on the mixture indices corresponding to different spectral coefficients along the same spatial basis as

$$p(\{x_{jk}\}_k) = \sum_{z_1,z_2,\ldots} p(\{z_{jk} = z_k\}_k) \prod_k \mathcal{N}(x_{jk}|0, \sigma_{jk,z_k}). \tag{4.11}$$

Note that this is only one possible choice for a joint model. Other choices include using a radial-exponential distribution (as was done in Chapter 3) or a multi-variate Gaussian-mixture. While the former allows only the variance of the distribution to be tuned but with a fixed shape, the latter is more expressive but has a much larger number of parameters (it needs to learn a covariance matrix across all spectral dimensions for each mixture component). The model in (4.11) offers a reasonable compromise between the number of parameters and control over distribution shape. Furthermore, the marginal distribution for any single coefficient under this model is the same as the one in (4.8). Therefore, to fit this model, we first learn $\{p(z_{jk} = z), \sigma_{jk,z}\}$ for each coefficient x_{jk} individually as before. We can then estimate the joint distribution of the indices $p(\{z_{jk}\}_k)$ from the set of training patches $\{X^i\}$ as

$$p(\{z_{jk} = z_k\}_k) \propto \sum_i \prod_k p(z_{jk} = z_k | x^i_{jk}). \tag{4.12}$$

This allows us to learn joint distributions over any subset of coefficients depending on the application, and models for overlapping sets are guaranteed to be consistent (*i.e.* have the same marginal distributions, *etc.*).

Having fit this model, we can use the learned joint distribution of the mixture indices $p(\{z_{jk}\}_k)$ to reason about the relationships between the corresponding coefficients $\{x_{jk}\}_k$. Figure 4.12 shows the estimated conditional distributions $p(z_{2k'}|z_{2k})$ for different pairs of spectral coefficients along the spatial basis S_2. As expected, these distributions are different from the corresponding marginal distributions $p(z_{2k'})$ (also shown for comparison). We find that conditioned on the mixture index z_{2k} having a value corresponding to higher mixture component variance, the index $z_{2k'}$ for a different spectral coefficient $x_{2k'}$ is more likely to correspond to higher variance mixture component as well, which is consistent with our observations in Fig. 4.11. Therefore, observing a high magnitude value for one coefficient makes a high value for another spectral coefficient along the same spatial basis more likely. This joint model can be exploited during inference, for example, when estimating a hyperspectral image from noisy or incomplete observations.

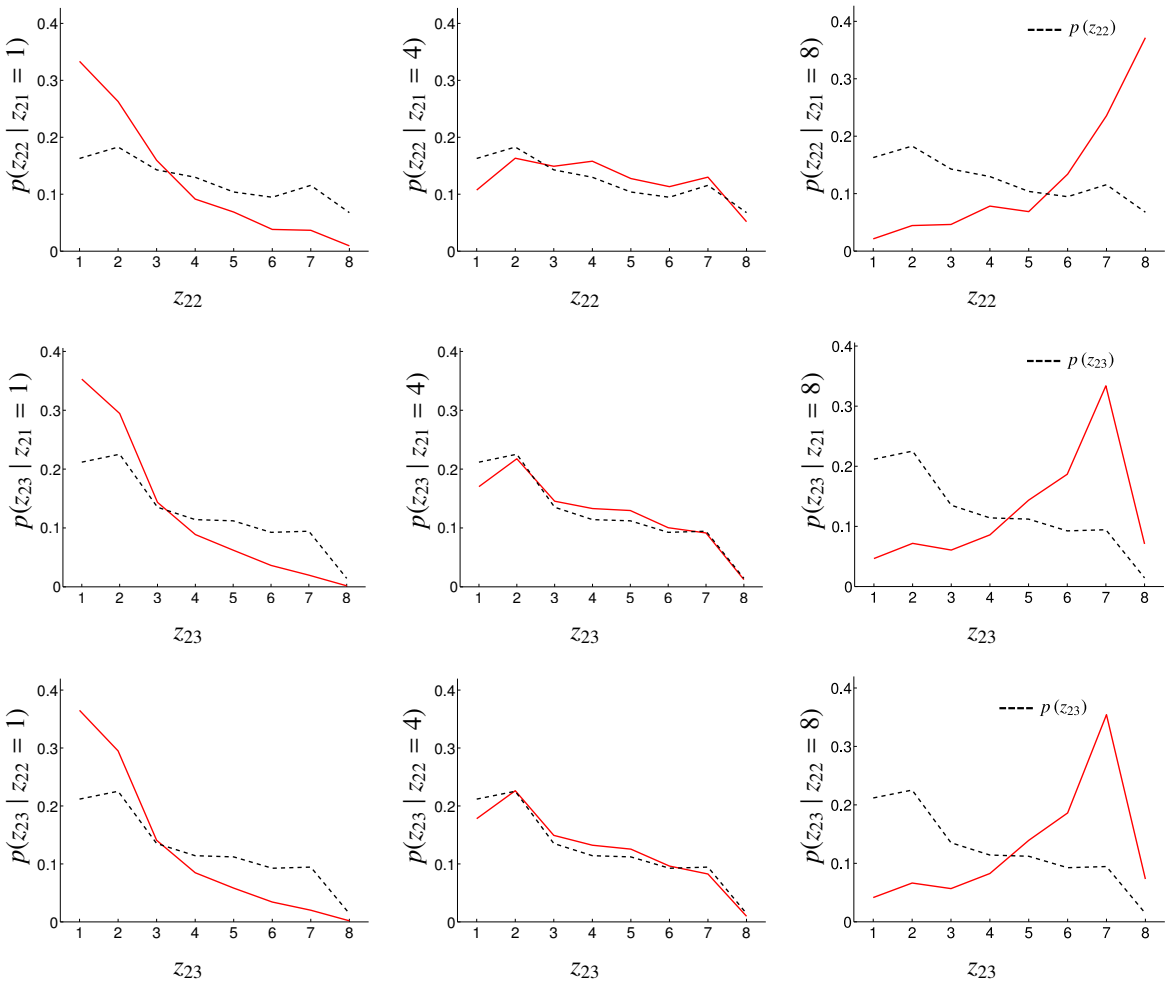


Figure 4.12: Relationship between mixture indices. Conditional distributions of the mixture indices $p(z_{2k'}|z_{2k})$ for different pairs of spectral coefficients along the same spatial basis S_2. Knowing the value of the mixture index z_{2k} for one spectral coefficient changes the distribution of the index $z_{2k'}$, corresponding to a different spectral coefficient, from the marginal distribution $p(z_{2k'})$ (shown with dotted black line for comparison). Broadly, these graphs suggest that higher/lower magnitudes of one coefficient make higher/lower magnitudes respectively for other coefficients, along the same spatial basis, more likely.

4.6 Discussion

In this chapter, we examined the joint spatial and spectral statistics of hyperspectral images using a new database of real-world scenes. We found that a separable basis, composed of independent spatial and spectral components, serves as an efficient representation for hyperspectral patches, and we studied the relative variance in these components. We then explored the statisti-

cal properties of coefficients in this basis and found that higher-frequency spatial components are accurately described by Gaussian mixture models. We also established that for the same spatial subband, different spectral coefficients are mutually dependent, and we described a joint distribution for mixture indices for different coefficients that encodes these dependencies.

A natural application of the statistical characterization described here is in hyperspectral imaging. In the future, acquisition systems should exploit the interdependencies and correlations between different spatio-spectral components, so as to efficiently acquire hyperspectral images with fewer measurements. General color filter array patterns (such as those proposed in [105]) can be designed to trade off spatial and spectral accuracy based on the relative variances of different components, and reconstruction methods can use the joint coefficient models during estimation. Similarly, these statistics are likely to be useful when estimating "clean" hyperspectral images from observations degraded by noise, blur, chromatic aberration, *etc.* The database can be used as "ground truth" to evaluate different strategies for these applications.

This chapter presents a first look at spatio-spectral statistics and representations for hyperspectral images. Further research in this direction should include studies on the statistics of specific classes of objects or regions in hyperspectral images, and ways to leverage these statistics for vision applications. In addition to hyperspectral object models for recognition, understanding the difference in the statistics of homogenous regions with variations due to shading, relative to those of regions that include material boundaries, may be useful for segmentation and recovering "intrinsic images" [16]. Furthermore, these models can be expanded to include wavelengths beyond the visible spectrum, for example, to study the redundancies in the information carried by the infra-red and long wavelength visible bands.

Other avenues of future work include looking at representations derived using more sophisticated techniques such as independent component analysis and fields of experts [9], with the choice of representation likely to be geared towards specific vision tasks. It would also be interesting to evaluate the utility of sparse codes, which have been previously proposed to describe spatial and

spectral components independently in hyperspectral data [110]. Our observation about the mutual dependence between spectral coefficients for different spatial bands suggests that it would be useful to consider joint spatio-spectral coding strategies.

5

Conclusion

In this dissertation, we focused on the utility of statistical inference for problems in computer vision. When the observed image does not uniquely determine the parameters we seek to estimate, we showed that knowledge of the statistical properties of natural images, as well as those of other scene parameters, can be used to arrive at a solution. We discussed the need for choosing a modeling strategy that is adapted to the task at hand, and noted how this often involves making assumptions to simplify inference, while leveraging the statistical properties that are useful for that particular estimation problem.

We used this intuition in developing novel estimation algorithms for two vision applications. The first looked at estimating the parameters of spatially-varying motion blur in an image. Since blur acts on the texture content of images (but uniformly across color channels), we used a model that encoded the properties of sharp edges in greyscale, taking care to account for the arbitrary variation in the contrast of these edges from region to region. We found that we also had to use a per-pixel color model in conjunction with this edge model to yield robust estimates.

The second application, color constancy, has a very different formulation. Here, a spatially-uniform illuminant affected the colors of the observed scene at the pixel level, and therefore the statistics of color in natural images had to be central to our approach. But we found that instead of modeling the statistics of individual pixels, the colors of image derivatives were more informative and easier to encode efficiently. Our proposed algorithm for this task also included an example of modeling the statistical properties of other scene parameters, in this case the illuminant color, to improve accuracy.

Hence, we introduced two very different statistical models for the same class of signals, *i.e.* canonical images of natural scenes. Neither of these models is strictly more accurate than the other, it is just that they encode different aspects of image statistics, and were chosen to allow efficient and accurate inference for specific tasks. In addition to providing robust estimates for their respective applications, these algorithms helped to illustrate the *design choices* that need to be made when crafting a statistics-based algorithm for visual inference. Besides choosing appropriate

image models for each application, these design choices included using a convenient local Fourier representation for blur analysis, and a conjugate form for the illuminant distribution in the color constancy method.

While other inference tasks are likely to need image models that are specifically adapted, the models we proposed for the blur and illuminant estimation tasks can serve as useful starting points. For example, the image model used for blur estimation essentially provides a mathematical framework that favors image gradients being sharp, or spatially localized. While we used this model for the case when this sharpness was affected by convolution with a blur kernel, the same framework is likely to be useful for analyzing, say, the effect of a spatially smooth shading map being multiplied with the underlying sharp material boundaries.

The spatio-spectral approach used for the color constancy application, that jointly modeled the statistical properties of color and texture, is also likely to be useful for other tasks that deal with color and material properties. That is why in Chapter 4, we did away with the constraint of working with three channel images and explored these joint statistics for hyperspectral data. As described in that chapter, these statistics are directly useful when designing hyperspectral cameras and vision systems that use hyperspectral images as input. But the proposed representations and statistical models for hyperspectral data can also be used to obtain deeper insights into the properties of regular RGB images, which can be thought of as "spectrally downsampled" projections of their hyperspectral counterparts.

Current research trends in computer vision indicate that image statistics will have a crucial role to play as we move to new modalities of imaging. The emerging field of *Computational Photography* provides new and exciting opportunities to deploy image models, as we consider inference on observations not restricted to being images taken by traditional cameras. With the goal of recording more than just a projection of the light incident on the sensor plane, camera systems have been proposed that modify the capture process in a variety of ways— using coded apertures [41], programmable shutters [113], diffusion films [114], camera motion [115], *etc.*

In Chapter 2, we used blur in images from traditional cameras to reason about object motion. Computational cameras are often designed to deliberately introduce blur and other image distortions into the capture process in a controlled manner, using the resulting observations to either extract information that would not be present in a traditional image (such as depth, higher dynamic range,*etc.*), or to compensate for some other form of degradation during capture (such as defocus or motion blur, noise, *etc.*). To make optimal use of this flexibility to change the acquisition process itself, appropriate statistical models should be used to design the best capture strategy, and for estimation on the acquired data.

Bibliography

[1] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transfom," *IEEE Transactions on Computers*, vol. 100, no. 23, pp. 90–93, 1974.

[2] A. Chakrabarti and K. Hirakawa, "Effective separation of sparse and non-sparse image features for denoising," in *Proceedings of the IEEE International Conference on Audio, Speech and Signal Processing (ICASSP)*, 2008.

[3] P. Burt and E. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.

[4] S. Mallat, *A wavelet tour of signal processing*. Academic Press, 1999.

[5] E. Simoncelli and W. Freeman, "The steerable pyramid: A flexible architecture for multi-scale derivative computation," in *Proceedings of the IEEE Conference on Image Processing (ICIP)*, 1995.

[6] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, 2007.

[7] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proceedings of the Asilomar Conference on Signals, Systems and Computers*, 1993.

[8] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1999.

[9] S. Roth and M. Black, "Fields of experts: A framework for learning image priors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[10] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001.

[11] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision (ICCV)*. Published by the IEEE Computer Society, 1999, p. 1150.

[12] A. Oliva and A. Torralba, "Building the gist of a scene: The role of global image features in recognition," *Progress in Brain Research*, vol. 155, pp. 23–36, 2006.

[13] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, 2003.

[14] M. Crouse, R. Nowak, and R. Baraniuk, "Wavelet-based statistical signal processing using hidden markov models," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 886–902, 1998.

[15] B. Gunturk, J. Glotzbach, Y. Altunbasak, R. Schafer, and R. Mersereau, "Demosaicking: color filter array interpolation," *IEEE Signal Processing Magazine*, vol. 22, no. 1, pp. 44–54, 2005.

[16] M. Tappen, W. Freeman, and E. Adelson, "Recovering intrinsic images from a single image," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, pp. 1459–1472, 2005.

[17] W. Freeman, T. Jones, and E. Pasztor, "Example-based super-resolution," *IEEE Computer Graphics and Applications*, pp. 56–65, 2002.

[18] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman, "Discovering objects and their location in images," in *Proceedings of the International Conference on Computer Vision (ICCV)*, vol. 1. IEEE, 2005, pp. 370–377.

[19] Z. Stone, T. Zickler, and T. Darrell, "Toward large-scale face recognition using social network context," *Proceedings of the IEEE*, vol. 98, no. 8, pp. 1408–1415, 2010.

[20] R. Fergus, B. Singh, A. Hertzmann, S. Roweis, and W. Freeman, "Removing camera shake from a single photograph," in *ACM Transactions on Graphics Proceedings of SIGGRAPH*, 2006.

[21] A. Levin, Y. Weiss, F. Durand, and W. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[22] A. Chakrabarti, T. Zickler, and W. Freeman, "Spatially-varying blur analysis database and code," available at http://www.eecs.harvard.edu/~ayanc/svblur/.

[23] A. Chakrabarti, K. Hirakawa, and T. Zickler, "Color constancy with spatio-spectral statistics matlab implementation," available at http://www.eecs.harvard.edu/~ayanc/color-constancy/.

[24] A. Chakrabarti and T. Zickler, "Database of real-world hyperspectral images," available at http://vision.seas.harvard.edu/hyperspec/.

[25] J. Cai, H. Ji, C. Liu, and Z. Shen, "Blind motion deblurring from a single image using sparse approximation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[26] H. Ji and C. Liu, "Motion blur identification from image gradients," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[27] N. Joshi, R. Szeliski, and D. Kriegman, "PSF estimation using sharp edge prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[28] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," in *ACM Transactions on Graphics Proceedings of SIGGRAPH*, 2008.

[29] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce, "Non-uniform deblurring for shaken images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 491–498.

[30] A. Gupta, N. Joshi, C. Lawrence Zitnick, M. Cohen, and B. Curless, "Single image deblurring using motion density functions," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010, pp. 171–184.

[31] L. Yuan, J. Sun, L. Quan, and H. Shum, "Image deblurring with blurred/noisy image pairs," in *ACM Transactions on Graphics Proceedings of SIGGRAPH*, 2007.

[32] L. Bar, B. Berkels, M. Rumpf, and G. Sapiro, "A variational framework for simultaneous motion estimation and restoration of motion-blurred video," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.

[33] B. Bascle, A. Blake, and A. Zisserman, "Motion deblurring and super-resolution from an image sequence," in *Proceedings of the European Conference on Computer Vision*, 1996.

[34] W. Chen, N. Nandhakumar, and W. Martin, "Image motion estimation from motion smear-a new computational model," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 18, no. 4, pp. 412–425, 1996.

[35] S. Cho, Y. Matsushita, and S. Lee, "Removing non-uniform motion blur from images," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.

[36] P. Favaro, M. Burger, and S. Soatto, "Scene and motion reconstruction from defocused and motion-blurred images via anisotropic diffusion," in *Proceedings of the European Conference on Computer Vision*, 2004.

[37] P. Favaro and S. Soatto, "A variational approach to scene reconstruction and image segmentation from motion-blur cues," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[38] S. Dai and Y. Wu, "Estimating space-variant motion blur without deblurring," in *Proceedings of the IEEE Conference on Image Processing (ICIP)*, 2008.

[39] ——, "Motion from blur," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[40] J. Jia, "Single image motion deblurring using transparency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[41] A. Levin, R. Fergus, F. Durand, and W. Freeman, "Image and depth from a conventional camera with a coded aperture," in *ACM Transactions on Graphics Proceedings of SIGGRAPH*, 2007.

[42] A. Levin, "Blind motion deblurring using image statistics," in *Advances in Neural Information Processing Systems (NIPS)*, 2006.

[43] M. Wainwright and E. Simoncelli, "Scale mixtures of gaussians and the statistics of natural images," in *Advances in Neural Information Processing Systems (NIPS)*, 2000.

[44] Y. Zhang and N. Kingsbury, "Image deconvolution using a gaussian scale mixtures model to approximate the wavelet sparseness constraint," in *Proceedings of the IEEE International Conference on Audio, Speech and Signal Processing (ICASSP)*, 2009.

[45] H. Jeffreys, "An invariant form for the prior probability in estimation problems," *Proceedings of Royal Society of London. Series A, Mathematical and Physical Sciences*, pp. 453–461, 1946.

[46] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," in *ACM Transactions on Graphics Proceedings of SIGGRAPH*, 2004.

[47] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient n-d image segmentation," *International Journal of Computer Vision (IJCV)*, vol. 70, no. 2, pp. 109–131, 2006.

[48] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society B*, vol. 39, no. 1, pp. 1–38, 1977.

[49] S. Beauchemin and J. Barron, "The computation of optical flow," *ACM Computing Surveys (CSUR)*, vol. 27, no. 3, pp. 433–466, 1995.

[50] G. Buchsbaum, "A spatial processor model for object colour perception," *Journal of the Franklin Institute*, vol. 310, no. 1, pp. 1–26, 1980.

[51] V. Cardei and B. Funt, "Committee-based color constancy," in *Proceedings of the IS&T/SID Color Imaging Conference*, 1999, pp. 311–313.

[52] D. Forsyth, "A novel algorithm for color constancy," *International Journal of Computer Vision (IJCV)*, vol. 5, no. 1, 1990.

[53] D. Brainard and W. Freeman, "Bayesian color constancy," *Journal of the Optical Society of America (JOSA) A*, vol. 14, no. 7, pp. 1393–1411, 1993.

[54] C. Rosenberg, T. Minka, and A. Ladsariya, "Bayesian color constancy with non-gaussian models," in *Advances in Neural Information Processing Systems (NIPS)*, 2003.

[55] P. Gehler, C. Rother, A. Blake, T. Minka, and T. Sharp, "Bayesian color constancy revisited," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[56] J. van de Weijer, T. Gevers, and A. Gijsenij, "Edge-based color constancy," *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2207–2214, 2007.

[57] A. Gijsenij, T. Gevers, and J. van de Weijer, "Generalized gamut mapping using image derivative structures for color constancy," *International Journal of Computer Vision (IJCV)*, vol. 86, no. 2, pp. 127–139, 2010.

[58] K. Hirakawa and T. Parks, "Adaptive homogeneity-directed demosaicing algorithm," in *Proceedings of the IEEE Conference on Image Processing (ICIP)*, 2003.

[59] G. West and M. H. Brill, "Necessary and sufficient conditions for von kries chromatic adaptation to give color constancy," *Journal of Mathematical Biology*, vol. 15, no. 2, pp. 249–258, 1982.

[60] G. Finlayson, M. Drew, and B. Funt, "Diagonal transforms suffice for color constancy," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 1993.

[61] H. Chong, S. Gortler, and T. Zickler, "The von Kries hypothesis and a basis for color constancy," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.

[62] G. Finlayson and G. Schaefer, "Convex and non-convex illuminant constraints for dichromatic colour constancy," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001.

[63] A. Hurlbert and T. Poggio, "Synthesizing a color algorithm from examples," *Science*, vol. 239, no. 4839, pp. 482–485, 1988.

[64] V. Cardei, B. Funt, and K. Barnard, "Estimating the scene illumination chromaticity using a neural network," *Journal of the Optical Society of America (JOSA) A*, vol. 19, no. 12, pp. 2374–2386, 2002.

[65] R. Gershon, A. Jepson, and J. Tsotsos, "From [r, g, b] to surface reflectance: computing color constant descriptors in images," *Perception*, vol. 17, pp. 755–758, 1988.

[66] B. Singh, W. Freeman, and D. Brainard, "Exploiting spatial and spectral image regularities for color constancy," in *Proc. Workshop on Statistical and Computational Theories of Vision*, 2003.

[67] I. Selesnick, "The estimation of laplace random vectors in additive white gaussian noise," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3482–3496, 2008.

[68] F. Sinz, S. Gerwinn, and M. Bethge, "Characterization of the p-generalized normal distribution," *Journal of Multivariate Analysis*, vol. 100, no. 5, pp. 817–820, 2009.

[69] L. Shi and B. Funt, "Re-processed version of the gehler color constancy dataset of 568 images," accessed from http://www.cs.sfu.ca/~colour/data/.

[70] A. Gijsenij, T. Gevers, and J. van de Weijer, "Computational color constancy: Survey and experiments," *IEEE Transactions on Image Processing*, 2011.

[71] E. Land and J. McCann, "Lightness and retinex theory," *Journal of the Optical Society of America (JOSA) A*, vol. 61, no. 1, p. 1, 1971.

[72] K. Hirakawa and T. Parks, "Chromatic adaptation and white-balance problem," in *Proceedings of the IEEE Conference on Image Processing (ICIP)*, 2005.

[73] A. Werner, "The spatial tuning of chromatic adaptation," *Vision Research*, vol. 43, no. 15, pp. 1611–1623, 2003.

[74] M. Webster, "Human colour perception and its adaptation," *Network: Computation in Neural Systems*, vol. 7, no. 4, pp. 587–634, 1996.

[75] T. Hansen, M. Olkkonen, S. Walter, and K. Gegenfurtner, "Memory modulates color appearance," *Nature Neuroscience*, vol. 9, no. 11, pp. 1367–1368, 2006.

[76] M. Olkkonen, T. Hansen, and K. Gegenfurtner, "Color appearance of familiar objects: Effects of object shape, texture, and illumination changes," *Journal of Vision*, vol. 8, no. 5, p. 13, 2008.

[77] T. Owens, K. Saenko, A. Chakrabarti, Y. Xiong, T. Zickler, and T. Darrell, "Learning object color models from multi-view constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[78] A. Chakrabarti, D. Scharstein, and T. Zickler, "An empirical camera model for internet color vision," in *Proceedings of BMVC*, 2009.

[79] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proceedings of the European Conference on Computer Vision*, 2010.

[80] F. Ciurea and B. Funt, "A large image database for color constancy research," in *Proceedings of the IS&T/SID Color Imaging Conference*, 2003.

[81] E. Belluco, M. Camuffo, S. Ferrari, L. Modenese, S. Silvestri, A. Marani, and M. Marani, "Mapping salt-marsh vegetation by multispectral and hyperspectral remote sensing," *Remote Sensing of Environment*, vol. 105, no. 1, pp. 54–67, 2006.

[82] M. Borengasser, W. Hungate, and R. Watkins, *Hyperspectral remote sensing: principles and applications*. CRC, 2008.

[83] A. Castrodad, Z. Xing, J. Greer, E. Bosch, L. Carin, and G. Sapiro, "Discriminative sparse representations in hyperspectral imagery," in *Proceedings of the IEEE Conference on Image Processing (ICIP)*, 2010.

[84] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 8, pp. 1778–1790, 2004.

[85] E. Underwood, S. Ustin, and D. DiPietro, "Mapping nonnative plants using hyperspectral imagery," *Remote Sensing of Environment*, vol. 86, no. 2, pp. 150–161, 2003.

[86] D. Dicker, J. Lerner, P. Van Belle, S. Barth, D. Guerry *et al.*, "Differentiation of normal skin and melanoma using high resolution hyperspectral imaging." *Cancer Biology & Therapy*, vol. 5, no. 8, p. 1033, 2006.

[87] L. Randeberg, I. Baarstad, T. Løke, P. Kaspersen, and L. Svaasand, "Hyperspectral imaging of bruised skin," in *Proc. SPIE*, 2006.

[88] G. Stamatas, C. Balas, and N. Kollias, "Hyperspectral image acquisition and analysis of skin," in *Proc. SPIE*, 2003.

[89] R. Rowe, K. Nixon, and S. Corcoran, "Multispectral fingerprint biometrics," in *Proc. Info. Assurance Workshop*, 2005.

[90] S. Lansel, M. Parmar, and B. A. Wandell, "Dictionaries for sparse representation and recovery of reflectances," in *Proceedings of SPIE— Computational Imaging VII*, 2009.

[91] D. Marimont and B. Wandell, "Linear models of surface and illuminant spectra," *Journal of the Optical Society of America (JOSA) A*, vol. 9, no. 11, pp. 1905–1913, 1992.

[92] J. Parkkinen, J. Hallikainen, and T. Jaaskelainen, "Characteristic spectra of munsell colors," *Journal of the Optical Society of America (JOSA) A*, vol. 6, no. 2, pp. 318–322, 1989.

[93] R. Baddeley, P. Hancock, and L. Smith, "Principal components of natural images," *Network*, vol. 3, pp. 61–70, 1992.

[94] Y. Weiss and W. Freeman, "What makes a good model of natural images?" in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[95] G. Heidemann, "The principal components of natural images revisited," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 28, no. 5, pp. 822–826, 2006.

[96] T. Lee, T. Wachtler, and T. Sejnowski, "The spectral independent components of natural scenes," in *Biologically Motivated Computer Vision*, 2000.

[97] C. Párraga, T. Troscianko, and D. Tolhurst, "Spatiochromatic properties of natural images and human vision," *Current Biology*, vol. 12, no. 6, pp. 483–487, 2002.

[98] D. Ruderman, T. Cronin, and C. Chiao, "Statistics of cone responses to natural images: implications for visual coding," *Journal of the Optical Society of America (JOSA) A*, vol. 15, pp. 2036–2045, 1998.

[99] T. Wachtler, T. Lee, and T. Sejnowski, "Chromatic structure of natural scenes," *Journal of the Optical Society of America (JOSA) A*, vol. 18, no. 1, pp. 65–77, 2001.

[100] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[101] J. Cai, H. Ji, C. Liu, and Z. Shen, "Blind motion deblurring from a single image using sparse approximation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[102] H. Choi and R. Baraniuk, "Multiscale image segmentation using wavelet-domain hidden markov models," *IEEE Transactions on Image Processing*, vol. 10, no. 9, pp. 1309–1321, 2002.

[103] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio, "Pedestrian detection using wavelet templates," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1997.

[104] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Transactions on Image Processing*, vol. 4, no. 11, pp. 1549–1560, 2002.

[105] F. Yasuma, T. Mitsunaga, D. Iso, and S. Nayar, "Generalized assorted pixel camera: Post-capture control of resolution, dynamic range and spectrum," Columbia University, Tech. Rep. CUCS-061-08, 2008.

[106] H. Du, X. Tong, X. Cao, and S. Lin, "A prism-based system for multispectral video acquisition," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009.

[107] D. Foster, S. Nascimento, and K. Amano, "Information limits on neural identification of colored surfaces in natural scenes," *Visual Neuroscience*, vol. 21, no. 03, pp. 331–336, 2004.

[108] S. Hordley, G. Finalyson, and P. Morovic, "A multi-spectral image database and its application to image rendering across illumination," in *Proc. Int. Conf. on Image and Graphics*, 2004.

[109] J. Park, M. Lee, M. Grossberg, and S. Nayar, "Multispectral imaging using multiplexed illumination," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.

[110] M. Parmar, S. Lansel, and B. A. Wandell, "Spatio-spectral reconstruction of the multispectral datacube using sparse recovery," in *Proceedings of the IEEE Conference on Image Processing (ICIP)*, Oct. 2008, pp. 473–476.

[111] G. Wyszecki and W. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulas, 2nd Edition*. Wiley-Interscience, 2000.

[112] R. Boie and I. Cox, "An analysis of camera noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, pp. 671–674, 1992.

[113] R. Raskar, A. Agrawal, and J. Tumblin, "Coded exposure photography: motion deblurring using fluttered shutter," *ACM Transactions on Graphics Proceedings of SIGGRAPH*, vol. 25, no. 3, pp. 795–804, 2006.

[114] O. Cossairt, C. Zhou, and S. Nayar, "Diffusion coded photography for extended depth of field," *ACM Transactions on Graphics Proceedings of SIGGRAPH*, vol. 29, no. 4, p. 31, 2010.

[115] A. Levin, P. Sand, T. Cho, F. Durand, and W. Freeman, "Motion-invariant photography," in *ACM Transactions on Graphics Proceedings of SIGGRAPH*, 2008, pp. 1–9.