## Abstract

The image recorded by a camera depends on three factors: The physical content of the scene, the illumination incident on the scene, and the characteristics of the camera. This leads to a problem for many applications where the main interest is in the physical content of the scene. Consider, for example, a computer vision application which identifies objects by colour. If the colours of the objects in a database are specified for tungsten illumination (reddish), then object recognition can fail when the system is used under the very blue illumination of a clear sky. This is because the change in the illumination affects object colours far beyond the tolerance required for reasonable object recognition. Thus the illumination must be controlled, determined, or otherwise taken into account.

The ability of a vision system to diminish, or in the ideal case, remove, the effect of the illumination, and therefore "see" the physical scene more precisely, is called colour constancy. There is ample evidence that the human vision system exhibits some degree of colour constancy. Interest in human vision, as well as robotics and image reproduction applications, has led to much research into computational methods to achieve colour constancy. Much progress has been made, but most work has addressed the problem in the context of synthetic data and quite simple physical conditions. However, in order to serve the needs of the proposed applications, it is necessary to develop and test computational colour constancy algorithms for real image data. This practical development of computational colour constancy is the focus of this work.

In order to study and to use computational models for real image data, it is necessary to develop a model of the physical characteristics of the vision system of interest. Specifically, we want to predict the camera response based on spectral input, and as part of this work I propose a new method for doing this. In addition, the spatial variation of the optical system, and its noise characteristics are considered.

The next part of this study is the comprehensive testing of current colour constancy algorithms. I present results for a number of algorithms both on synthetic data, and on a large database of real images. The image database consists of images of 33 scenes under 11 illuminants. The illuminants were chosen to approximate a uniform coverage of the span of common natural and man made illuminants. The results from this study, as well as the testing paradigm that was developed, provide a foundation for the rest of the work, which specifically sets out to improve computational colour constancy on image data.

The first area studied to improve computational colour constancy is the sensor sharpening method of Finlayson et al. Sensor sharpening has been proposed as a method of improving colour constancy, but it has not yet been tested in conjunction with real colour constancy algorithms. In this work, I set out to test the degree to which sensor sharpening can help current colour constancy algorithms. I find that the current sharpening methods do not address the needs of this domain, and thus as part of this work, I propose a pragmatic new sharpening method.

I then propose several improvements to variants of Forsyth's CRULE algorithm. The first is a new method of choosing a solution from the feasible set. The second reduces the reliance of the method on the diagonal model. This method enables CRULE to be extended to work with fluorescent surfaces, and is the first algorithm to deal with such input. In order to consider colour constancy in the face of fluorescence, I propose a simple, but effective model for characterizing such surfaces. In addition to the CRULE extension, I also use this characterization to implement a fluorescent capable version of Finlayson et al.'s Colour by Correlation method. These algorithms are tested on generated data, the 321 images taken for the comparison work, and, most importantly, on a set of 59 images with fluorescent surfaces taken for this purpose.

A third modification to gamut mapping algorithms proposed in this work allows them to use specularities to an advantage. Specularities have long been put to use by colour constancy algorithms, but existing algorithms that use specular information are limited in that they require such information to be present. The method presented here combines the use of specular and non-specular information. Thus the algorithm can provide good performance when good specularities are present, or when there is a reasonable diversity of matte surfaces, or some combination of these conditions. Furthermore, it should be noted that the method uses the fact that specular reflection tends to be relatively bright. Most current methods which use specularities do not consider this. Finally, the method is applicable to specular reflection from coloured metallic surfaces, and is, in fact, the first colour constancy algorithm which can use such input gainfully.

Another algorithm chosen for close study is Finlayson et al.'s chromaticity based Colour by Correlation method. This method is attractive because, unlike the CRULE derivatives, it can take advantage of statistical information about the world. However, the comparison work done as part of this thesis indicates that the pixel brightness is also a very important source of information—information that cannot be used by chromaticity based methods. Therefore, I developed a three-dimensional version of Colour by Correlation. This algorithm performs better than all algorithms considered in this thesis when tested on synthetic data. Notably, it performs significantly better than the chromaticity version on both generated and image data. Unfortunately, the performance of the new algorithm on image data still lags slightly behind the best of the CRULE derivatives.

In summary, the many pragmatic difficulties encountered when computational colour constancy meets the real world demand an approach that embraces modeling the physical nature of the world, analysis of camera characteristics, and the use of real images for comprehensive testing and development of algorithms. These requirements for successfully applying colour constancy to the real world have driven the present work.

## Acknowledgments

It would have been impossible to complete this work without the help of others. Certainly I thank my supervisor, Brian Funt, for encouraging me to study color computer vision several years ago, and for allowing me to follow my own interests once I started doing so. Many thanks also go to Lindsay Martin, who helped greatly with the large amount of data collection demanded by this work. I also acknowledge the help of Manfred Trummer in formulating an approach to the sensor sharpening work.

Special thanks go to Emily Butler for being supportive of my work despite too many late night programming sessions, as well as for help with the proof-reading.

Finally, I acknowledge the financial support of NSERC, the Hewlett-Packard Company, ASI, the EBCO/ECPICH foundation, and SFU.

# Contents

A	pproval			ii
A	bstract			iii
A	cknowle	edgmer	ıts	vii
C	ontents			viii
Li	ist of Tab	oles		xii
Li	ist of Fig	ures		xv
1	Introd	uction		1
	1.1	Thesi	s Overview	4
2	Model Reproc	ing Sce luction	ene Illumination Colour for Computer Vision and Image A survey of computational approaches	e 7
	2.1	Image	e Formation and Capture	7
	2.2	Mode	els of Illumination Change	15
	2.3	Comp	outational Colour Constancy	19
		2.3.1	Grey World Algorithms	21
		2.3.2	Retinex Methods	23
		2.3.3	The Maloney-Wandell Algorithm	24
		2.3.4	Gamut Mapping Algorithms	25
		2.3.5	Bayesian Colour Constancy and Colour by Correlation	29
		2.3.6	Neural Network Colour Constancy	31
		2.3.7	Methods Based on Specularities	31
		2.3.8	Methods using Time Varying Illumination (multiple views)	33
		2.3.9	Methods using Spatially Varying Illumination	34

		2.3.10 Methods using Mutual Illumination	36
		2.3.11 Methods for Object Recognition and Image Indexing	36
	2.4	Conclusion	38
3	Camera	a calibration for colour constancy research	39
	3.1	The Camera Model	40
	3.2	Motivation for Camera Calibration	41
	3.3	Previous Work	42
	3.4	The Fitting Approach	44
	3.5	Experimental Method	50
	3.6	Results	51
	3.7	Conclusions	56
4	A Com	parison of Computational Colour Constancy Algorithms	58
	4.1	Approaches to Colour Constancy	59
	4.2	Error Measures	62
	4.3	Assumptions about the physical world: The Mondrian worl beyond	d and 64
	4.4	The World for Algorithm Calibration and Generating Synth Test Images	netic 64
	4.5	Computational Colour Constancy Algorithms	69
		4.5.1 The Grey World method	70
		4.5.2 The Retinex Algorithm	70
		4.5.3 Gamut Mapping Methods	71
		4.5.4 Colour by Correlation	74
		4.5.5 Neural Net Methods	78
	4.6	Computational Colour Constancy Meets the Real World	79
	4.7	Experiments with Generated Data	82
	4.8	Experiments with Image Data	100
	4.9	Discussion	108
	4.10	Conclusions	110
5	Sensor	Sharpening for Computational Colour Constancy	112

	5.1	Sensor Sharpening Theory	113
	5.2	Sensor Sharpening in Practice	115
	5.3	The New Approach to Sharpening	117
	5.4	Error Measures	120
	5.5	Colour Constancy Methods	121
	5.6	Colour Constancy with Sharpening	122
	5.7	Results	125
	5.8	Conclusion	134
6	Improv	vements to Gamut Mapping Colour Constancy Algorithms	135
	6.1	Diminishing Diagonal Model Error	136
	6.2	Improving Solution Choice	138
	6.3	Results	141
	6.5	Conclusion	149
7	Colour	Constancy with Fluorescent Surfaces	151
	7.1	Characterizing Fluorescent Surfaces	152
	7.2	Colour Constancy with Fluorescent Surfaces	153
	7.3	Experimental Results	153
	7.4	Conclusions	163
8	Colour	Constancy with Specular and Non-Specular Surfaces	164
	8.1	Introduction	165
	8.2	Extending CRULE for specularities	167
	8.3	Experiments with Generated Data	169
	8.4	Experiments with Image Data	176
	8.5	Conclusions	177
9	Gamut	Mapping for the Real World	182
	9.1	Combining the Methods	182
	9.2	Results	183
	9.3	Conclusion	184
10	Colour	by Correlation in a Three-Dimensional Colour Space	187

10.1	The Extension to Three Dimensions	188	
10.2	Algorithm Summary	191	
10.3	Experiments	194	
10.4	Conclusion	199	
11 Is Machine Colour Constancy Good Enough?201			
11.1	Colour Indexing and Colour Constancy	202	
11.2	The Test Images	203	
11.3	Results	206	
11.4	Conclusion	210	
12 Conclusions and Contributions 212			
Bibliograp	Bibliography 219		

# List of Tables

Table 3.1	Error obtained on 612 data points using various fitting methods
Table 4.1	A summary of the gamut-mapping algorithms investigated in this paper73
Table 4.2	Algorithm performance with respect to a number of error measures
Table 4.3	Algorithm performance with respect to a number of error measures for generated scenes with specularities
Table 4.4	Algorithm chromaticity performance with respect to two error measures for generated scenes with specularities with three levels of simulated clipping
Table 4.5	Algorithm performance for 321 real images using a generic pre-processing method105
Table 4.6	Algorithm chromaticity performance for 321 real images using the best pre-processing method for that algorithm106
Table 5.1	Sharpening results for the Sony DXC-930 digital camera128
Table 5.2	Sharpening results for the Kodak DCS-200 digital camera129
Table 5.3	Sharpening results for the Kodak DCS-420 digital camera130

Table 6.1	Algorithm chromaticity performance for some of the algorithms developed here, together with the original methods, for generated scenes with 4, 8, and 16 surfaces	147
Table 6.2	The image data results of the new algorithms compared to related algorithms.	149
Table 7.1	Key to the algorithms studied in this chapter	155
Table 7.2	Algorithm chromaticity performance on generated scenes with 4 and 8 surfaces, with and without fluorescent surfaces	158
Table 7.3	Algorithm chromaticity performance on image data for two sets of scenes	162
Table 8.1	Key to the algorithms studied in this chapter	171
Table 8.2	RMS error in (r,g) chromaticity estimates for 1000 generated scenes with 8 surfaces	173
Table 8.3	RMS error in illuminant R+G+B estimates for 1000 generated scenes with 8 surfaces	174
Table 8.4	Chromaticity performance on a number of algorithms on three image data sets with and without clipping	181
Table 9.1	Chromaticity performance of selected comprehensive algorithms on image data.	186

Table 10.1	Algorithm chromaticity performance under three
	different conditions of variants of the new methods
	designed for the various conditions, as well as that for a
	number of comparison algorithms
Table 10.2	Algorithm chromaticity performance in the Mondrian
	world of the new method (MAP and MMSE), as well as
	that for a number of comparison algorithms
Table 10.3	Algorithm chromaticity performance on 321 images of the
	new method with two estimators (MAP and MMSE), as
	well as that for a number of comparison algorithms199
Table 11.1	Colour constancy error and indexing performance by
	algorithm208

# List of Figures

Figure 2.1	The geometry used to define and apply the BDRF	10
Figure 2.2	The same scene taken under two different illuminants	16
Figure 2.3	Visualization of the first part of the gamut mapping procedure	27
Figure 2.4	Visualization of the second part of the gamut mapping procedure	28
Figure 3.1	The variance of red channel measurements verses their values for data similar to that ones used for calibration	46
Figure 3.2	The non-linearity of the red channel response for the Sony DXC-930 camera used for the sensor fitting experiments.	48
Figure 3.3	The sensor response found using the fitting method introduced in this paper without the range constraints in equation (3.8).	53
Figure 3.4	The sensor response found using the pseudo inverse method applied to linearized data	53

Figure 3.5	The sensor response found using the pseudo inverse method with positivity constraints applied to linearized
	data
Figure 3.6	RMS relative fitting error versus sample size (log scales)55
Figure 4.1	The chromaticity distributions of the various sets of illuminants used in this study67
Figure 4.2	The camera sensor response functions used in this study68
Figure 4.3	Illustration of the basic idea of gamut-mapping color constancy72
Figure 4.4	An example posterior distribution, showing the probabilities that the illuminant in the training set with chromaticity (r,g) explains the observed data produced from a randomly selected illuminant and 8 randomly selected surface reflectances
Figure 4.5	Distribution of error in r=R/(R+G+B) for selected algorithms
Figure 4.6	Error in (r,g) chromaticity as a function of the number of surfaces for selected algorithms
Figure 4.7	Error in (r,g) chromaticity as a function of the number of surfaces for some of the better performing algorithms
Figure 4.8	The change in (r,g) error due to the addition of simulated specularities

Figure 4.9	The change in (r,g) error in the case with specularities due to simulated clipping
Figure 4.10	The images of the scenes used for real data under the canonical illuminant103
Figure 4.11	The ball scene under each of the eleven illuminants used for image data104
Figure 4.12	Algorithm performance for 321 real images using the best pre-processing method for each algorithm compared with synthetic results for a comparably difficult number of surfaces (8)
Figure 5.1	The RMS RGB mapping error between the corrected image and the target image, by algorithm-sharpening method combination for the Sony DXC-930 camera
Figure 5.2	The RMS RGB mapping error between the corrected image and the target image, by algorithm-sharpening method combination for the Kodak DCS-200 camera
Figure 5.3	The RMS RGB mapping error between the corrected image and the target image, by algorithm-sharpening method combination for the Kodak DCS-420 camera
Figure 6.1	Illustration of fundamentals of gamut mapping colour constancy136
Figure 6.2	Illustration of the modification to the gamut mapping method to reduce diagonal model failure

Figure 6.3	Algorithm chromaticity performance versus the number of surfaces in generated scenes, showing the main gamut mapping algorithms and their non-diagonal counterparts
Figure 6.4	Algorithm chromaticity performance versus the number of surfaces in generated scenes, showing the main gamut mapping algorithms and their non-diagonal counterparts
Figure 6.5	Algorithm chromaticity performance versus the number of surfaces in generated scenes, showing the selected gamut mapping algorithms, including ones with the new solution selection method
Figure 7.1	The performance of the algorithms developed for fluorescence surfaces and their standard counterparts with generated scenes with 30% fluorescence surfaces
Figure 7.2	The six scenes with fluorescent surfaces used for image data
Figure 7.3	One of the scenes with fluorescent surfaces under all 11 illuminants
Figure 8.1.	Illustration of gamut extension used for specularities. The gamuts are actually polytopes in 3 dimensional RGB space
Figure 8.2	Chromaticity performance of some of the algorithms developed here under 5 different test conditions. More detailed results are available in Table 8.2

Figure 8.3	Illuminant magnitude performance of some of the algorithms developed here under 5 different test
	conditions
Figure 8.4	The 14 scenes with metallic specularities used for image data
Figure 8.5	One of the scenes with metallic surfaces under all 11 illuminants
Figure 8.6	Chromaticity performance of some of selected algorithms on three image data sets with and without clipping
Figure 9.1	Chromaticity performance of selected algorithms over 529 images
Figure 10.1	The chromaticity performance of the new method as compared to the two-dimensional version of the algorithm and two gamut mapping methods195
Figure 11.1.	The 20 objects in the image database as seen under a single illuminant204
Figure 11.2.	The cruncheroos object as seen under the 11 test illuminants205
Figure 11.3.	Object recognition performance as a function of colour constancy error

# **Chapter One**

## Introduction

The image recorded by a camera depends on three factors: the physical content of the scene, the illumination incident on the scene, and the characteristics of the camera. This leads to a problem for many applications where the main interest is in the physical content of the scene. Consider, for example, a computer vision application which identifies objects by colour. If the colours of the objects in a database are specified for tungsten illumination (reddish), then object recognition can fail when the system is used under the very blue illumination of blue sky. This is because the change in the illumination affects object colours far beyond the tolerance required for reasonable object recognition. Thus the illumination must be controlled, determined, or otherwise taken into account.

The ability of a vision system to diminish, or in the ideal case, remove, the effect of the illumination, and therefore "see" the physical scene more precisely, is called colour constancy. There is ample evidence that the human vision system exhibits some degree of colour constancy (see, for example, [1-4]). One consequence of our own color constancy processing is that we are less aware of colour constancy problems which face machine vision systems. These problems become more obvious when dealing with image

reproduction. For example, if one uses indoor film (balanced for tungsten illumination) for outdoor photography, one will get a poor result. The colour change is much larger than we would expect, based on our experience of looking at familiar objects, such as a friend's face, both indoors and out.

This leads us to the relationship between colour constancy and image reproduction. The main thesis here is that illumination modeling is also beneficial for image reproduction and image enhancement. In the above example, taking a good picture required selecting the film based on the illumination. However, choosing among a limited number of film types provides only a rough solution, and has the obvious limitation that human intervention is required. Digital image processing yields opportunities for improved accuracy and automation, and as digital imaging becomes more prevalent, the demand for image manipulation methods also increase. Often modeling the scene illumination is a necessary first step for further image enhancements, as well as being important for standard image reproduction.

To complete the argument that modeling scene illumination is necessary for image reproduction, we must consider the interaction of the viewer with the reproduction. For example, one may ask why the viewer does not remove the blue cast in a reproduction, much as they would remove a blue cast due to blue light in the original scene. First, note that failing to remove the cast from the reproduction is consistent with the claim that humans exhibit colour constancy. This is because colour constancy is by definition the reduction of the effect of the scene illumination, which is the illumination present when the reproduction is viewed, not the illumination present when the picture was taken. Thus, the empirical result is that the viewing experience is sufficiently different in the two cases that human colour constancy functions according to the definition. The most obvious difference is that scenes occupy the entire visual field whereas reproductions do not. However, even if a reproduction occupies the entire visual field, the viewer will still not remove a blue cast due to incorrect film type. It is possible to identify many other ways that the two viewing experiences differ, and the characterization of the relevant differences is a subject of ongoing research. To summarize, since the human viewer compensates for the viewing

illumination, but not the illumination present when the image was taken, image reproduction must compensate for this scene illumination.

Naturally, this is only the beginning of the story. For example, a completely illumination invariant photographic system would not be able to "see" mountains painted red by a setting sun. Here the effect of the illumination is very much a part of the photograph. Nonetheless we expect a perfect image capture system to be cognizant of the overall illumination, because it is relevant to us whether the alpenglow is especially red, or alternatively, white, with the rest of the scene being especially blue, as would be suggested if we were to use indoor film to capture the scene. Thus for automated high quality reproduction, illumination modeling is still an obvious starting point. Similarly, for computer vision applications the goal is not to ignore illumination effects, but to separate them from the overall image signal. For example, a shadow contains information about the world which we want to use, but we also want to recognize that the shadow boundary is not a change in scene surface.

To emphasize the connections between image reproduction and computer vision, imagine a vision system which is able to determine the physical characteristics of the scene, and thus implicitly the illumination. Using this information, we can now reproduce the scene as it would be appear under any illumination, including the original illumination. This is suggestive of image enhancement, which can be defined as image processing which leads to an image which is, in some sense, more appropriate for human viewing. An example of image enhancement which may be approached through illumination modeling is dynamic range compression. Here the problem is that the range of intensities in natural images far exceeds that which can be reproduced linearly with inexpensive technologies. This wide range of intensities is largely due to the wide range of illumination strengths. For example, printed media cannot linearly represent the intensities in a bright outdoor scene and a dark shadow therein. A vision system which can recognize the shadow as such can be used to create an enhanced reproduction where the shadow is reproduced as less dark.

Illumination modeling is required here because mistakenly applying the same processing to a dark surface is undesirable.

It may be argued that the image enhancement example above is actually an example of image reproduction, because the human experience of the scene may involve a less dark shadow—certainly it involves seeing the detail in the shadow. Regardless of the best categorization of the application, it should be clear that proceeding effectively requires an adequate model of human vision, which itself is intimately linked with our research area. One may argue that adequate models of human vision might be obtainable by mere measurement, but one popular point of view, which I think is valuable to pursue, is that a complete understanding of the human vision system requires an understanding of what *computational* problems are being solved [5]. This point of view brings us back to computer vision, which is largely inspired by human abilities, and the philosophical stance that those abilities can be viewed as the result of computation.

In summary, I claim that modeling scene illumination is central to the recovery of facts about the world from image data, which inevitably has the scene illumination intertwined with the information of interest. Furthermore, progress in modeling the scene illumination will result in progress in computer vision, image enhancement, and image reproduction.

### 1.1 Thesis Overview

This thesis deals with the computational colour constancy problem with an emphasis on methods which work well on image data. There has been much progress in the development of computational methods for colour constancy applied to synthetically generated data, but relatively little is known about how these methods behave when they meet the real world. In this thesis I begin by studying the issues and problems encountered in the practical application of computational colour constancy methods. This leads to contributions in the areas of camera calibration and pre-processing for computational colour constancy. I then initiate a testing paradigm to study

the performance of these methods under a variety of conditions, including image data with a variety of pre-processing strategies. For these experiments I used 321 carefully calibrated images, which goes significantly beyond any previous study with image data. One of the findings is that, in the case of several promising algorithms, there are significant discrepancies between results on generated data and results on image data. This finding pointed the way to further progress by identifying some of the problems contributing to this discrepancy.

The next part of the thesis uses the knowledge gained in the algorithm testing work to improve computational colour constancy. The first improvement is a new method for sensor sharpening which is more appropriate than previously existing methods for use with current colour constancy algorithms. As part of that work I provide results for the application of sensor sharpening to computational colour constancy. Previous to this study, such results were not available.

I then propose several extensions to the three-dimensional gamutmapping methods. This is inspired, in part, by the observation that these methods tend to be very effective on image data. I propose three general classes of extensions. The first shows how the three-dimensional gamutmapping methods can be made more resilient to diagonal model failure. The second is a new method for selecting the solution from the set of feasible solutions. The third method extends the gamut-mapping methods to deal with and use specularities. This method is applicable to metallic specularities, and is the first colour constancy algorithm which makes use of such information.

In this thesis I also consider computational colour constancy with fluorescent surfaces. I begin by developing a simple but accurate method for characterizing fluorescent surfaces for colour constancy processing. I then use this characterization to add fluorescent surface capability to several algorithms. In the case of the gamut-mapping method, this relies on the second extension mentioned above.

One of the main points of this thesis is that the magnitude of the pixels carry important information for colour constancy, even if one is only

interested in correcting for chromaticity. This consideration leads to the modification of the promising Colour by Correlation method so that it can use the brightness information. The realization of this extension is not completely trivial due to the asymmetric roles of brightness and chromaticity in colour constancy. However, once the modifications are made, the performance is substantially better than the chromaticity version.

The final part of this thesis relates computational colour constancy performance to a real world task, specifically colour sensitive object recognition. The main finding here is that colour constancy does indeed help such a task, but that the existing methods still come up short for the requirements of this application. Therefore, as always, there are challenges for the future.

# **Chapter Two**

# Modeling Scene Illumination Colour for Computer Vision and Image Reproduction: A survey of computational approaches

In this chapter I will provide a starting point for the study of computational methods for colour constancy. I will begin with the physics of image formation and capture, and then consider models for illumination change. Having provided this foundation, I will summarize the available approaches for the computational colour constancy problem.

## 2.1 Image Formation and Capture

Modeling illumination on the basis of an image (or a sequence of images), can be viewed as inverting the image formation process. Thus it is essential to look at the relationship between the world and the images in a forward direction. The main conclusion that we will draw is that determining the illumination from an image is inherently very under-constrained, and thus making progress in our quest requires making intelligent assumptions about the world.

We begin with a digital image, which is a sampling of a light signal traditionally modeled by a continuous function of wavelength and geometric variables. In the case of a colour image, we have three samples which are ostensibly centered over the same location<sup>1</sup>. For our purposes, the nature of the spatial sampling is not critical, and I generally will ignore the associated issues. On the other hand, we are quite interested in the sampling of the input with respect to wavelength. In general, the response of image capture systems to a light signal,  $L(\lambda)$ , associated with a given pixel can be modeled by:

$$\rho^{(k)} = \mathbf{F}^{(k)}(\boldsymbol{v}^{(k)}) = \int \mathbf{L}(\boldsymbol{\lambda}) \mathbf{R}^{(k)}(\boldsymbol{\lambda}) d\boldsymbol{\lambda}$$
(2.1)

where  $R^{(k)}(\lambda)$  is the sensor response function for the k<sup>th</sup> channel,  $v^{(k)}$  is the k<sup>th</sup> channel response, and  $\rho^{(k)}$  is the k<sup>th</sup> channel response linearized by the wavelength independent function  $F^{(k)}$ . In this formulation,  $R^{(k)}(\lambda)$  absorbs the contributions due to the aperture, focal length, sensor position in the focal plane. This model has been verified as being adequate for computer vision over a wide variety of systems (see, for example, [6-10] and the references therein). This model is also assumed for the human visual system (see for example [11]), and forms the basis for the CIE colorimetry standard. Here,  $R^{(k)}(\lambda)$  are linear transformations of the colour matching functions,  $\rho^{(k)}$  are the X, Y and Z colour coordinates, and  $F^{(k)}$  is taken to be the identity function.

In the common case of three camera channels,  $\rho^{(1)}$  is the linearized red channel, hereafter designated by R,  $\rho^{(2)}$  is the green channel, designated by G, and  $\rho^{(3)}$  is the blue channel designated by B. Often we wish to ignore the

<sup>&</sup>lt;sup>1</sup>In 3-CCD cameras the sample location is the same within manufacturing tolerances, but in the increasingly common case of mosaic cameras, the samples are interpolated from adjacent sensors in the mosaics, and unfortunately, the exact nature of the sampling is invariably proprietary.

brightness information in the sensor response. In the usual case of three sensors, this is done by mapping the three dimensional RGB responses into a two dimensional chromaticity space. There are numerous ways to do this. The most common is the mapping r=R/(R+G+B) and g=G/(R+G+B). This will be referred to as the rg chromaticity space. Another mapping, used in the two dimensional gamut mapping algorithms described below is given by (R/B, G/B).

The continuous functions in (2.1) are normally approximated by a sequence of measurements at successive wavelengths. For example, the commonly used PR-650 spectraradiometer samples spectra at 101 points from 380nm to 780nm in 4nm steps with each sampling function being approximately 8nm wide. Thus it is natural and very convenient to represent them as vectors, with each component being a sample. Using this representation, (2.1) becomes:

$$\boldsymbol{\rho}^{(k)} = \mathbf{L} \bullet \mathbf{R}^{(k)} \tag{2.2}$$

This notation emphasizes that image capture projects vectors in a high dimension space into a N-space, where N is 3 for standard colour images. This means that image capture loses a large amount of information, and recovery of the spectra from the vision system's response is not possible. Put differently, many different spectra have exactly the same camera response. For human vision in reasonably bright conditions, N is also three, and again, many different spectra will be seen as the same colour. This forms the basis of colour reproduction. Rather than attempt to reproduce the spectra of the scene's colour, it is sufficient to create a spectra which has the same response, or, equivalently, has the same projection into the three dimensional sensor space.

I will now discuss the formation of the input signal, designated by  $L(\lambda)$  above, along the lines in [12] and [13].  $L(\lambda)$  is the result of some illuminant signal  $E(\lambda)$  interacting with the surface being viewed. Since the interaction is

linear it is natural to define the reflectance of the surface as the ratio of the reflected light to the incident light. This ratio is a function of the direction of the illumination, the direction of the camera, and the input and output polarization which I will ignore for the moment. This gives us the bidirectional reflectance function (BDRF), defined as the ratio of the image radiance  $\delta L(\lambda, \vartheta_e, \phi_e)$  in the direction of the solid angle  $\delta \Omega_e$  due to the surface irradiance  $\delta E(\lambda, \vartheta_i, \phi_i)$  from  $\delta \Omega_i$  (see Figure 2.1):

$$f(\lambda, \vartheta_{i}, \phi_{i}, \vartheta_{e}, \phi_{e}) = \frac{\delta L(\lambda, \vartheta_{e}, \phi_{e})}{\delta E(\lambda, \vartheta_{i}, \phi_{i})}$$
(2.3)



Figure 2.1: The geometry used to define and apply the BDRF.

Given the BDRF, we can express the signal from a surface in the more realistic case of multiple extended light sources by<sup>2</sup>:

$$L(\lambda, \vartheta_{e}, \phi_{e}) = \int_{-\pi}^{\pi} \int_{0}^{\pi/2} f(\lambda, \vartheta_{i}, \phi_{i}, \vartheta_{e}, \phi_{e}) E(\lambda, \vartheta_{i}, \phi_{i}) \cos \vartheta_{i} \sin \vartheta_{i} d\vartheta_{i} d\phi_{i}$$
(2.4)

The reflectance of most surfaces does not change significantly if the surface is rotated about the surface normal. Such surfaces are referred to as isotropic. In this case the BDRF can be simplified to  $f = f(\lambda, \phi_i, \phi_e, \vartheta_e - \vartheta_i)$  or, more commonly,  $f = f(\lambda, i, e, g)$ , where the third variable is now the angle between the viewing and illuminant directions.

One important limitation of the BDRF is that is inappropriate for fluorescent surfaces. In the case of fluorescence, a surface absorbs energy at one wavelength, and emits some of that energy at a different wavelength. Since the interaction is linear for any pair of input and output wavelengths, the BDRF now becomes  $f = f(\lambda_{in}, \lambda_{out}, \vartheta_i, \varphi_i, \vartheta_e, \phi_e)$ . So far, fluorescence has been largely ignored in computer vision, likely because of the difficulties it presents. In the case of human vision, psychophysical experiments suggests that a sufficiently fluorescent surface is perceived as self-luminous [14] Finally, if we wish to extent the BDRF to include polarization, then we need to add an input polarization multi-parameter, and an output polarization multi-parameter. This complete model of reflection is referred to as the light transfer function in [15].

Since the BDRF is a function of three (isotropic case) or four geometric parameters, measuring the BDRF for even one surface is very tedious. Nonetheless, some such data has become available for a variety of surfaces [16]. However, it is clear that we need simpler models, and that the main importance of the measured data is for testing our models, rather than being used directly. I will now discuss some of the models that have been developed.

<sup>&</sup>lt;sup>2</sup>The BDRF is expressed in terms of the light reaching a specific region due to the radiance in the direction of the solid angle. When we integrate over the light itself, we must include the cosine factor for the foreshortening of the surface as seen by the illuminant, or perhaps more intuitively, due to the light falling at an oblique angle. The sine factor is due to the form of the differential of the solid angle in polar coordinates.

The simplest possible form of the BDRF is a constant. This corresponds to perfectly diffuse reflection, also referred to as Lambertian reflection. A Lambertian reflector appears equally bright, regardless of the viewing direction. If the Lambertian reflector reflects all energy incident on it without loss, then it can be shown that  $f = \frac{1}{\pi}$  [12].

In computer vision it is common to forgo the BDRF in favour of the reflectance factor function [13, 17], which expresses the reflectance of a surface with respect to that of a perfect diffuser in the same configuration. This is closer to the usual method of measuring reflectance which is to record the reflected spectrum of both the sample and a standard reflectance known to be close to a perfect diffuser. The reflectance factor function is then the ratio of these two. In order to keep the two expressions of reflectance distinct and to maintain consistency with the literature, I will denote the reflectance factor function by  $S(\lambda)$ . This leads to the most common form of the imaging equations:

$$\rho^{(k)} = \int \mathbf{R}^{(k)}(\lambda) \mathbf{S}(\lambda) \mathbf{E}(\lambda) \mathrm{d}\lambda$$
(2.5)

The simplicity of Lambertian reflectance makes it an attractive approximation for modeling reflectance, but unfortunately, it is a poor model in many cases. Investigating the physics of reflectance leads to better models. One very useful idea is the dichromatic model proposed for computer vision in [18]. This model has two terms corresponding to two reflection processes. Specifically, the light reflected from a surface is a combination of the light reflected at the interface, and light which enters the substrate and is subsequently reflected back as the result of scattering in the substrate. These two reflection components are referred to as the interface reflection and the body reflection. Furthermore, for most non-metallic materials, the interface reflection is only minimally wavelength dependent, and thus light reflected in this manner has the same spectra as the illuminant. On the other hand, the scattering processes that lead to the body reflection are normally wavelength dependent. Formally, then, the dichromatic model for a surface reflectance  $S(\lambda)$  is given by:

$$S(\lambda) = m_i(i, e, g)S_i(\lambda) + m_b(i, e, g)S_b(\lambda)$$
(2.6)

where  $S_i(\lambda)$  is the interface reflectance (usually assumed to be a constant),  $S_b(\lambda)$  is the body reflection, and  $m_i(i, e, g)$  and  $m_b(i, e, g)$  are attenuation factors which depend on the geometry developed above (see Figure 2.1). A key simplification offered is the separation of the spectral and geometric effects. Several researchers have carried out experiments testing the efficacy of this model in the context of computer vision [13, 19-21].

The body reflection is often assumed to be Lambertian. In the case of smooth dielectrics, a detailed analysis indicates that this is a good approximation, provided that the angles *e* and *i* in Figure 2.1 are less than 50<sup>o</sup> [22]. In the case of rough surfaces, Lambert's law breaks down, even if the material itself obeys Lambert's law. The effect of surface roughness on the body reflection is modeled in [23].

Surface roughness also affects specular reflection. Two approaches to modeling this effect are surveyed in [24]. The first is based on physical optics (Beckmann-Spizzichino) and the second uses geometric optics (Torrance-Sparrow). Physical optics is exact, but requires approximations and simplifications due to the nature of the equations. Geometric optics is simpler, but requires that the roughness is large compared to the wavelength of light under consideration. Both methods require some specification of the statistical nature of the roughness. The analysis in [24] leads to the proposal of three contributions to reflection: The body reflection, the specular lobe, and the specular spike, which is normally only present for very smooth surfaces. Thus this analysis extends the dichromatic idea by splitting one of the reflection processes into two.

A similar model can be developed in the case of metals [19]. Metals have no body reflection, and the interface reflection is often quite wavelength dependent, explaining the colour of metals such as gold and copper. The proposed model again separates the spectral and geometric effects. The efficacy of such a monochromatic model is tested in [19], and is found to be reasonable.

I will now discuss models for the wavelength dependence of surface reflection, as well as illuminant spectral distribution. Although many of the physical process involved are known, physics based models appropriate for computer vision have yet to be developed. However, statistical models have been studied extensively and have proven to be very useful. The general method is to express a data set as a linear combination of a small number of basis functions. In the case of a surface reflectances we have:

$$S(\lambda) \approx \sum_{i=0}^{N} \sigma_i S_i(\lambda)$$
 (2.7)

Here  $S_i(\lambda)$  are the basis functions and  $\sigma_i$  are the projections. Similarly, for illuminants we have:

$$E(\lambda) \approx \sum_{i=0}^{N} \varepsilon_i E_i(\lambda)$$
(2.8)

If a set of spectra is well approximated by N basis functions, then that set will be referred to as N-dimensional. Such models work well when the spectra of interest are smooth, and thus quite band limited. This seems to be good assumption for surface reflectances, as several large data sets of surface reflectances have been successfully modeled using such models [25-28]. For example, in [27] the spectra of 1257 Munsell color chips were fit to 90% with 4 basis functions, and to 98% with 8 basis functions. The number of basis functions required to fit daylight is even smaller [29, 30]. Dixon [30] found that for a daylight data set taken at one location, three basis functions accounted for 99% of the variance, and for another data set, four functions accounted for 92% of the variance. It should be noted that the spectra of a number of artificial lights, including fluorescent lights, are not smooth, and when such lights need to be included, the approximation in (2.8) is less useful.

The basis functions are normally determined from data sets of spectra using either singular value decomposition, or occasionally by principal component analysis, where the mean is first subtracted from the data set. The singular value decomposition is usually applied to the spectra directly, but in [31] it is argued that the basis functions should be found relative to the vision system sensors. In short, the standard method is sub-optimal because it will reduce errors fitting spectra to which the vision system has little sensitivity at the expense of spectra which need to be well approximated. Thus [31] proposes using the responses directly to find basis functions for surface reflectances or illuminants (one-mode analysis). In the usual case that the responses are produced by both reflectance and illuminant spectra, two-mode analysis is used, which requires iteratively applying one-mode analysis to obtain estimates of the surface reflectance bases and the illuminant bases (convergence is guaranteed).

Finite dimensional models allow image formation to be modeled compactly using matrices. For example, assuming three dimensional surface reflectance functions, we can define a lighting matrix for a given illuminant  $E(\lambda)$  by:

$$\Lambda = \begin{vmatrix} \int E(\lambda)S_{1}(\lambda)R_{1}(\lambda) & \int E(\lambda)S_{2}(\lambda)R_{1}(\lambda) & \int E(\lambda)S_{3}(\lambda)R_{1}(\lambda) \\ \int E(\lambda)S_{1}(\lambda)R_{2}(\lambda) & \int E(\lambda)S_{2}(\lambda)R_{2}(\lambda) & \int E(\lambda)S_{3}(\lambda)R_{2}(\lambda) \\ \int E(\lambda)S_{1}(\lambda)R_{3}(\lambda) & \int E(\lambda)S_{2}(\lambda)R_{3}(\lambda) & \int E(\lambda)S_{3}(\lambda)R_{3}(\lambda) \end{vmatrix}$$
(2.9)

Then for a surface  $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \sigma_3)'$ , the response  $\boldsymbol{\rho} = (\rho_1, \rho_2, \rho_3)'$  is given simply as:

$$\boldsymbol{\rho} = \boldsymbol{\Lambda}\boldsymbol{\sigma} \tag{2.10}$$

### 2.2 Models of Illumination Change

Consider two images of the same scene under two different illuminants. For example, Figure 2.2 shows a ball in front of a green background taken under two illuminants, a tungsten illuminant for which the camera is well balanced, and simulated deep blue sky. Now, a priori based on (2.5), each pixel RGB is affected differently by the illumination change. However, there is clearly a systematic response as well—under the bluer light, all pixels seem to tend towards blue. In this section I will discuss models for the systematic response, as it is this response is the key to progress.



**Figure 2.2**: The same scene taken under two different illuminants. The image on the left was taken under tungsten illumination, which is an appropriate illuminant for the camera settings used. The image on the right is the same scene taken with an illuminant which is similar in colour temperature to deep blue sky.

To aid in the presentation, I will now introduce some notation. In order to be consistent with the gamut mapping approaches described below, I will always describe mappings from the image of a scene taken under a unknown illuminant, to that taken under a known illuminant. Following Forsyth [32], the known illuminant will also be referred to as the canonical illuminant. Quantities specific to the unknown illuminant will be superscripted with U, and quantities specific to the canonical illuminant will be super-scripted with C.

One common simple model of illumination change is a single linear transformation. Thus each pixel of the image taken under the unknown illuminant,  $\mathbf{p}^{U} = (\rho_{1}^{U}, \rho_{2}^{U}, \rho_{3}^{U})'$ , is mapped to the corresponding pixel of the image taken under the canonical illuminant,  $\rho^{C} = (\rho_{1}^{C}, \rho_{2}^{C}, \rho_{3}^{C})'$ , by  $\mathbf{p}^{C} = \mathbf{M}\mathbf{p}^{U}$ , where M is single 3 by 3 matrix used for all pixels. Such a model can be justified using the finite (specifically, 3) dimensional models discussed above. From (2.10) we can estimate  $\mathbf{p}^{U} = \Lambda^{U} \mathbf{\sigma}$  and  $\mathbf{p}^{C} = \Lambda^{C} \mathbf{\sigma}$  which gives the estimate  $\mathbf{p}^{C} = \Lambda^{C} (\Lambda^{U})^{-1} \mathbf{p}^{U}$ , and thus M above is given explicitly by:  $\mathbf{M} = \Lambda^{C} (\Lambda^{U})^{-1}$  It should be noted that due to a number of factors, the linear transformation model of illumination change can easily be more accurate than the finite dimensional models used to justify it. More to the point, the transformation  $\mathbf{M} = \Lambda^{C} (\Lambda^{U})^{-1}$  does not need to be the best possible M for our particular scene, illuminant pair, and camera sensors.

If we restrict M above to be a diagonal matrix, we get an even simpler model of illumination change. Such a model will be referred to as the diagonal model. The diagonal model maps the image taken under one illuminant to another by simply scaling each channel independently. For concreteness, consider a white patch in the scene with response under an unknown illuminant  $\mathbf{p}^{U} = (\rho_{1}^{U}, \rho_{2}^{U}, \rho_{3}^{U})'$  and response under a known canonical illuminant  $\mathbf{p}^{C} = (\rho_{1}^{C}, \rho_{2}^{C}, \rho_{3}^{C})'$ . Then the response of the white patch can be mapped from the test case to the canonical case by scaling the i<sup>th</sup> channel by  $\frac{\rho_{i}^{C}}{\rho_{i}^{U}}$ . To the extent that this same scaling works for the other, non-white patches, we say that the diagonal model holds.

The diagonal model has a long history in colour constancy research. It was proposed by von Kries as a model for human adaptation [33], and is thus often referred to as the von Kries coefficient rule, or coefficient rule for short. This model has been used for most colour constancy algorithms. The limitations of the model itself have been explored in [34-37]. In [36], Worthey and Brill discuss how the efficacy of the diagonal model is largely a function of the vision system sensors, specifically whether or not they are narrow band, and whether or not they overlap. The relationship is intuitively understood by observing that if the sensors are delta functions, the diagonal model holds exactly. In [35] it is pointed out that the use of narrow band illumination, which has a similar effect to narrow band sensors, aids the colour constancy observed and modeled in the well known Retinex work [1].

In [38], Finlayson et al propose the idea of using a linear combination of the vision system's sensors to improve the diagonal model. If the vision system sensors are represented by the columns of a matrix, then the new sensors are obtained by post multiplying that matrix by the appropriate transform T. An important observation is that if camera responses are represented by the rows of a matrix R, then the camera response to the new, modified sensors, is also obtained by post multiplication by T. The main technical result in sensor sharpening is finding the transformation T. Three methods for finding T are proposed: "sensor based sharpening", "database sharpening", and "perfect sharpening". Sensor based sharpening is a mathematical formulation of the intuitive idea that narrower band (sharper) sensors are better. Database sharpening (discussed further below) insists that the diagonal model holds as well as possible in the least squares sense for a specific illumination change. Finally, perfect sharpening does the same for any illumination change among a set of two dimensional illuminants in a world of three dimensional reflectances.

In database sharpening, RGB are generated using a database of reflectance spectra, together with an illuminant spectrum and the sensors. This is done for two separate illuminants. Let A be the matrix of RGB for the first illuminant and B be the matrix for the second, with the RGB's placed row-wise. In the sharpening paradigm we map from B to A with a sharpening transform, followed by a diagonal map, followed by the inverse transform. If we express each transform by post multiplication by a matrix we get:  $A \approx BTDT^{-1}$ . In database sharpening the matrix T (and implicitly D) is found that minimizes the RMS error,  $||A - BTDT^{-1}||_2$ . T is found by diagonalizing M, where M minimizes  $||A - BM||_2$ . Thus the sharpening transform gives exactly the same error as the best linear transform M, and therefore, for a specific illumination change, the diagonal model is equivalent to the a priori more powerful full matrix model. This notion is explored in detail in [39].

In summary, the diagonal model is the simplest model of illumination change that gives reasonable results. As will become clear below, its simplicity supports many algorithms by keeping the number of parameters to be estimated small. It should be noted that since overall brightness is often arbitrary in colour constancy, the number of parameters is often one less than the number of diagonal elements. In general, the error incurred in colour constancy is a combination of parameter estimation error, and the error due to the model of illumination change. Intuitively, the error due to parameter estimation increases with the number of parameters. With current colour constancy methods, the error in parameter estimation in the case of diagonal model algorithms is still large compared to the error due to diagonal model itself, especially when the camera sensors are sufficiently sharp, or when sharpening can be used (see [8] for some results). Thus it would seem that there is little to recommend using models with more parameters than sensors (less one, if brightness is considered arbitrary).

So far I have been discussing the simple case that the illumination is uniform across the image under consideration. However, the above generalizes easily to the case where the illumination varies, as any given model of illumination change must apply locally. Thus in the case of varying illumination, we have an entire spatially varying field of mappings. This means that the diagonal model is sufficient because we now model the illumination change of each image sample independently. Formally, in the usual case of three sensors, each response  $\mathbf{p}^{U} = (\rho_{1}^{U}, \rho_{2}^{U}, \rho_{3}^{U})'$  is mapped to  $\rho^{C} = (\rho_{1}^{C}, \rho_{2}^{C}, \rho_{3}^{C})'$  by a diagonal matrix specific to that response:  $\mathbf{p}^{C} = \text{diag}\left(\frac{\rho_{1}^{C}}{\rho_{1}^{U}}, \frac{\rho_{2}^{C}}{\rho_{2}^{U}}, \frac{\rho_{3}^{C}}{\rho_{3}^{U}}\right)\mathbf{p}^{U}$ .

### 2.3 Computational Colour Constancy

As discussed in the introduction, the goal of computational colour constancy is to diminish the effect of the illumination to obtain data which more precisely reflects the physical content of the scene. This is commonly characterized as finding illuminant independent descriptors of the scene. However, we must insist that these descriptors carry information about the physical content of the scene. For example, computing a field of zeros for every image is trivially illuminant independent, but it is useless.

One we have an illumination independent description of the scene, it can be used directly for computer vision, or it can be used to compute an image of how the scene would have looked under a different illuminant. For image reproduction applications, this illuminant is typically one for which the vision system is properly calibrated. It has proved fruitful to use such an image itself as the illuminant invariant description [8, 32, 40]. Ignoring degenerate cases, illuminant invariant descriptions can be inter-converted, at least approximately. However, the choice of invariant description is not completely neutral because it is normally more accurate to directly estimate
the descriptors that one is interested in. This often leads us to prefer using the image of the scene under a known, canonical illuminant as the illuminant invariant description. In the case of image reproduction this should be clear, as we are typically interested in how the scene would have appeared under an illuminant appropriate for the vision system. It is equally the case in computer vision, if only because most computer vision algorithms developed so far assume that the there is an illuminant—and typically ignore the problem that it may change. Specifically, computer vision algorithms tend to work on pixel values, and thus implicitly assume both illumination and sensors are involved, as opposed to assuming that some other module delivers some abstract characterization of the scene. This makes sense, because such a characterization will have error, and thus it is preferable to use the raw data. An example is object recognition by colour histograms [41]. Here, a database of colour histograms of a variety of objects is computed from images of these objects. Since we know the illuminant used to create the database, a natural choice of descriptors is how the objects would appear under this known illuminant. Other choices can be made, perhaps with certain advantages, but likely at the expense of some error.

Many algorithms have been developed to find the illuminant invariant descriptions discussed above. The most prominent ones will be discussed below. Since the problem is under-constrained, making progress requires making some additional assumptions. The algorithms can be classified to some degree by which assumptions they make, and the related consideration of where they are applicable.

The most important classification axis is the complexity of the illumination, and the most important division is whether or not the illumination is uniform across the image. A second important classification axis is the whether the algorithm is robust with respect to specular reflection or the lack thereof. Some algorithms require the presence of specular reflections, others are neutral with respect to them, and some are degraded by them. Most algorithms assume that the illumination is uniform, and that there are no specularities. This has been referred to as the Mondrian world, since the collections of matte papers used in the Retinex experiments were

likened to paintings by Mondrian (this likeness is debatable). Finally, some algorithms attempt to recover a description which is only invariant with respect to illuminant chromaticity, ignoring illuminant brightness. It should be clear that any algorithm which also recovers brightness can be used as an algorithm to recover chromaticity by simply projecting the result. Also, any algorithm used to recover chromaticity can be used together with an estimate of brightness to be compared with algorithms which recover both. I will now discuss the most prominent approaches in the context of these classifications.

### 2.3.1 Grey World Algorithms

Perhaps the simplest general approach to colour constancy is to compute a single statistic of the scene, and then use this statistic to estimate the illumination, which is assumed to be uniform in the region of interest. An obvious candidate for such a statistic is the mean, and this leads to the so called grey world assumption. In physical terms, the assumption is that the average of the scene reflectance is relatively stable, and thus is approximately some known reflectance which is referred to as grey. Although this is a very simple approach, there are a number of possible variations. One distinction is the form of the specification of the grey. Possibilities include specifying the spectra, the components of the spectra with respect to some basis, and the RGB response under a known, canonical illuminant. A second, more important, distinction, is the choice of the grey. Given a method for specifying the grey, the best choice would be the actual occurrence of that grey in the world. However, this quantity is not normally available (except with synthetic data), and thus the choice of grey is an important algorithm difference.

One approach is to assume that the grey is in fact grey; specifically, the reflectance spectra is uniform. This means that its RGB response under a given illuminant is a particular fraction of that for a pure white. For example, a reflectance of 50% could be used, although this corresponds to a relatively bright grey as perceived by humans. Using the diagonal model, the algorithm is to normalize the image by the ratio of the RGB response to grey under the canonical illuminant, to that of the average image RGB. A related method is

to use the average spectra of a reflectance database to obtain the RGB of grey, instead of assuming uniform reflectance.

Buchsbaum used a grey world assumption to estimate a quantity analogous to the lighting matrix defined in (2.9) [42]. However, as pointed out by Gershon et al [43], the method is weakened by an ad hoc choice of basis, as well as the choice of grey, which was set to have specific, equal, coefficients in the basis. Gershon et al improved on the method by computing the basis from a database of real reflectances, and using the average of the database as the reflectance of their grey. The output of the algorithm is estimates of the coefficients of the surface reflectances with respect to the chosen basis. As touched upon above, for most applications, using the camera response as descriptors is likely preferable, and if this algorithm were modified in this manner, then it would become the last algorithm described in the previous paragraph.

Gershon et al recognized that exact correspondence between their model and the world requires segmentation of the image so that the average could be computed among surfaces as opposed to pixels. In their model, two surfaces should have equal weight, regardless of their respective sizes. The reliance on segmentation would seem to be problematic because segmentation of real images is difficult, but I will argue that this algorithm should degrade gracefully with respect to inaccurate segmentation. This is because the result from any segmentation corresponds to the result with perfect segmentation for some possible physical scene under the same illuminant (my observation-the paper does not analyze this). To see why this is the case, consider an inappropriate merge of regions. The average of the single resultant region is exactly the same as a mix of the two regions seen from sufficiently far away, and thus sampled differently. For example, we may not be able to segment the green, yellow, and red leaves in an autumn tree, but the average of the incorrectly segmented blob is no different in terms of input to the algorithm than a similar tree seen at a distance. The case of erroneous splitting also corresponds to the proper segmentation of a possible scene, specifically, a scene where the surfaces of the original scene have been split up and reorganized. Of course, as the segmentation improves, the results of the algorithm should also improve, but the results should always be reasonable.

### 2.3.2 Retinex Methods

An important contribution to colour constancy is the Retinex work of Land and his colleagues [1, 44-48] and further analyzed and extended by others [49-54]. The original aim of the theory is a computational model of human vision, but it has also been used and extended for machine vision. In theory, most versions of Retinex are robust with respect to slowly spatially varying illumination, although testing on real images has been limited to scenes where the illumination has been controlled to be quite uniform. Nonetheless, the varying illumination component of this work is both interesting and important. In Retinex based methods, varying illumination is discounted by assuming that small spatial changes in the responses are due to changes in the illumination whereas large changes are due to surfaces changes. The goal of Retinex is to estimate the lightness of a surface in each channel by comparing the quantum catch at each pixel or photoreceptor to the value of some statistic—originally the maximum— found by looking at a large area around the pixel or photoreceptor. The ratios of these quantities (or their logarithms) are the descriptors of interest, and thus the method implicitly assumes the diagonal model. The details vary in the various versions of Retinex.

In [1, 45] the method is to follow random paths from the pixel of interest. As each path is followed, the ratio of the response in each channel for adjacent pixels is computed. If the ratio is sufficiently close to one, then it is assumed that the difference is due to noise, or varying illumination, and the ratio is treated as exactly one. If, on the other hand, if the ratio is sufficiently different from one, then it is used as is. The ratios are then combined to determine the ratio the response of the pixel of interest to the largest response found in the path. Finally, the results for all the paths are averaged.

The above is simplified by using the logarithms of the pixel values. With this representation, the essence of the matter is differentiation (to identify the jumps), followed by thresholding (to separate reflectance from illumination), followed by integration (to recover lightness), and various schemes have been proposed to formulate Retinex as a calculus problem [49-54].

In [46, 47] Land also used differences in logarithms with thresholding, to remove the effect of varying illumination, together with the random path idea. However, the lightness estimate was changed to the average of the differences after thresholding. As before the result for a number of paths was averaged. In [48], the estimate was simplified even further to the logarithm of the ratio of the response of a given pixel to a weighted average of the responses in a moderately large region surrounding the pixel. The weighting function used was the inverse distance from the pixel of interest. In [51], a method to solve Horn's Poisson equation corresponding to Retinex can be approximated by a similar simple estimate, but the weighting function is now a Gaussian which is applied *after* logarithms are taken. Finally, in [55], Moore et al change the Gaussian to  $e^{-/rt/k}$ , as convolution with this kernel can be achieved using a resistive network, and thus is appropriate for their hardware implementation of Retinex.

If the illumination is assumed to be uniform, then the first version of Retinex discussed above amounts to simply scaling each channel by the maximum value found in the image. Similarly, the second method discussed converges to normalizing by the geometric mean [52], and thus it is essentially a grey world algorithm (as is the third method). Thus Retinex can be simply and more powerfully implemented if the illumination is assumed to be uniform.

### 2.3.3 The Maloney-Wandell Algorithm

An especially elegant method for computing surface descriptors from an image was proposed by Maloney and Wandell [56, 57]. This approach is based on the small dimensional linear models discussed above. Assuming that illuminants are N dimensional and surfaces are N-1 dimensional, where N is the number of sensors, the sensor responses under a fixed, unknown light will fall in an N-1 dimensional hyper-plane, anchored at the origin. The orientation of this plane indicates the illumination. Unfortunately, in the usual case of three sensors, this method does not work very well [58, 59] which is not surprising, as the dimensionality of surfaces is more than two, and the dimensionality of illuminants can easily be more than 3 if fluorescent lighting is a possibility. Further analysis of the Maloney-Wandell method, as well as an extension for the case where the same scene is captured under multiple lights is provided by D'Zmura and Iverson [60].

## 2.3.4 Gamut Mapping Algorithms

The gamut mapping approach was introduced by Forsyth [32], and has recently being modified and extended by Finlayson [40]. These approaches explicitly constrain the set of possible mappings from the image of the scene under the unknown illuminant to the image of the scene under the known, canonical, illuminant. Although Forsyth's analysis included both diagonal and linear maps, his most successful algorithm, CRULE, and all subsequent extensions have been restricted to diagonal maps.

One source of constraints is the observed camera responses (image pixels). The set of all possible responses due to all known or expected surface reflectances, as seen under a known, canonical illuminant, is a convex set, referred to as the canonical gamut. Similarly, the set of responses due to a unknown illuminant is also a convex set. Assuming the diagonal model of illumination change, the two gamuts are within a diagonal transformation of each other. The canonical gamut is known, but since the illuminant is unknown, we must use the observed sensor responses in the input image as an estimate of the unknown gamut. Since this estimate is a subset of the whole, there are a number of possible mappings taking it into the canonical gamut. Each such map is a possible solution, and the main technical achievement of the algorithm is calculating the solution set. A second part of the algorithm is to choose a solution from the set of possibilities. Since this algorithm delivers the entire feasible set of solutions, it has the advantage that it provides bounds on the error of the estimate. I will now provide some of the details for the computation of the solution set.

First, it is important that the gamuts are convex. A single pixel sensor may sample light from more than one surface. If we assume that the response is the sum of the responses of the two contributing pieces, and that the response due to each of these is proportional to their area, then it is possible to have any convex combination of the responses. Thus the gamut of all possible sensor responses to a given light must be convex.

Since the gamuts are convex, they will be represented by their convex hulls. Now consider the RGB's in the image taken under an unknown light. The convex hull of these RGB's will be referred to as the measured gamut. The measured gamut must be a subset of the unknown gamut, and since we are modeling illumination changes by diagonal transforms, each of these measured RGB's must be mapped into the canonical gamut by the specific diagonal transform corresponding to the actual illumination change. It can be shown that a diagonal transform which maps all measured gamut hull vertices into the canonical gamut will also map the non-vertex points into the canonical gamut. Thus only the measured gamut vertices need to be considered to find plausible illumination changes.

Figure 2.3 illustrates the situation using two-dimensional triangular sets for explanatory purposes. Here triangle "abc" represents the convex hull of the measured RGB's. A proposed solution must map it into the canonical gamut represented by triangle "ABC". Reiterating the above, a proposed solution must map "a" into the canonical gamut (and similarly "b" and "c").

Now the set of maps which take a given point (e.g. "a") into some point in the canonical gamut is determined by the maps that take that point into the hull points of the canonical gamut. If we use vectors to represent the mappings from the given point to the various canonical hull points, then we seek the convex hull of these vectors. It is critical to realize that we have introduced a level of abstraction here. We are now dealing with geometric properties of the mappings, not the gamuts. It is easy to verify that it is sufficient to consider the mappings to the hull points (as opposed to the entire set), by showing that any convex combination of the *maps* takes a given point into a similar convex combination of the canonical hull *points*.

The final piece of the logical structure is straightforward. Based on a given point ("a" in our example), we know that the mapping we seek is in a specific convex set. The other points lead to similar constraints. Thus we intersect the sets to obtain a final constraint set for the mappings. Figure 2.4 illustrates the process.

Recently Finlayson proposed using the gamut mapping approach in chromaticity space, reducing the dimensional complexity of the problem from three to two in the case of trichromats [40]. Not all chromaticity spaces will



an approximation of the entire gamut under the unknown illuminant.

convex combination of the maps "aA", "aB", and "aC" will work, and any map outside the consequence of the observation of colour "a", the set of possible maps from the unknown gamut to the canonical gamut is constrained to lie within the convex hull of maps "aA", "aB", and "aC".

Figure 2.3: Visualization of the first part of the gamut mapping procedure.

work. However, Finlayson showed that if the chromaticity space was obtained by dividing each of two sensor responses by a third, as in the case of (R/B, G/B), then convexity is maintained where required. One advantage to working in a chromaticity space is that the algorithm is immediately robust with respect to illumination intensity variation. Such variation is present in almost every image, as it originates from the ubiquitous effects of shading and extended light sources. Furthermore, specular reflections do not present problems because the resultant chromaticity is the same as that of the same surface with some added white.

In addition to using chromaticity space, Finlayson added an important new constraint. Not all theoretically possible lights are commonly encountered. From this observation, Finlayson introduced a constraint on the illumination. The convex hull of the chromaticities of the expected lights



The set of possible mappings from the gamut under the unknown illuminant to the canonical gamut is constrained to lie in this region.

Figure 2.4: Visualization of the second part of the gamut mapping procedure.

makes up an illumination gamut. Unfortunately, the corresponding set of allowable mappings from the unknown gamut to the canonical gamut is not convex (it is obtained from taking the component-wise reciprocals of the points in the above convex set). Nonetheless, Finlayson was able to apply the constraints in the two dimensional case. In [8] the convex hull of the nonconvex set was found to be a satisfactory approximation for an extensive set of real illuminants.

Unless the image has colours near the gamut boundaries, the set of possible diagonal transforms can be large enough that choosing a particular solution is an important second stage of the gamut mapping approach. In [32], the mapping which lead to the largest mapped volume was used. In [40], this method of choosing the solution was maintained in the case of two dimensional mappings used in the chromaticity version. In [8], the centroid of the solution set was used, both in the chromaticity case and in the RGB case. The centroid is optimal if the solutions are uniformly distributed and a least squares error measure is used. However, in the two dimensional case, a uniform distribution of the solutions is not a good assumption because of the distorted nature of the specific chromaticity space. This lead Finlayson and Hordley to propose finding the constraint sets in two dimensions, and perform the average in three dimensions [61]. They justify this method by showing that under reasonable conditions, the constraint set delivered by the two and three dimensional versions is the same.

### 2.3.5 Bayesian Colour Constancy and Colour by Correlation

Bayesian statistics has been applied to the colour constancy problem [59]. In Bayesian colour constancy, one assumes knowledge about the probability of occurrence of illuminants and surface reflectances. Furthermore, each illuminant and surface combination leads to an observed sensor response, and an illuminant together with a scene leads to a conjunction of observed sensor response. If we let **y** be the observed sensor responses, and let **x** contain parameters describing proposed illuminant and scene reflectances, then Bayes's method estimates P(x) by:

$$P(\mathbf{x}/\mathbf{y}) = \frac{P(\mathbf{y}/\mathbf{x})P(\mathbf{x})}{P(\mathbf{y})}$$
(2.11)

Since we are only interested in choosing **x**, and not the actual value of  $P(\mathbf{x}/\mathbf{y})$ , the denominator  $P(\mathbf{y})$  can be ignored. Once the estimates for  $P(\mathbf{x}/\mathbf{y})$  have been computed, a value for **x** must be chosen. One natural choice is the **x** corresponding to the maximum of  $P(\mathbf{x}/\mathbf{y})$ . However, if this maximum is an isolated spike, and a second slightly lower value is amidst other similar values, then intuitively, we would prefer the second value, because choosing it makes it more likely that we have a value that is close to the actual value in the face of measurement error. A common method to overcome this problem is to use a loss function which gives a penalty as a function of estimation error. Such a function may be convolved with  $P(\mathbf{x}/\mathbf{y})$  to yield the loss as a function of estimate, which is then minimized. Loss functions are discussed in detail in [59] which also includes the introduction of the new local mass loss function which is felt appropriate for the colour constancy application.

Bayesian colour constancy as described in [59] has a number of problems. First, the number of parameters is a function of the number of surfaces, and so the method is very computationally expensive. Second, the calculation of P(x) from illuminant and surface distributions assumes that the surfaces are independent, which implies that the image is properly segmented. If the image pixels are used instead, then the surfaces are not independent, as neighbours tend to be alike. Finally, the required statistical distributions of the world are not well known, and thus there is likely to be large discrepancies between simulation and real applications. In [59] the authors only test on synthetic scenes, but it should be noted that some effort was made to investigate the algorithm performance when the test statistics differed from the model statistics.

Some of these problems are elegantly addressed with colour by correlation [40], although an estimate of prior probability distributions is still required. Colour by correlation is a discrete implementation of the Bayesian concept. More importantly, the method is free from the complexities of implicitly estimating surface parameters. In colour by correlation, the probability of seeing a particular chromaticity, given each expected possible illuminant, is calculated. Then this array of probabilities is used, together with Bayes's method, to estimate the probability that each of the potential illuminants is the actual illuminant. Finally, the best estimate of the specific illuminant is chosen using a loss function.

The Colour by Correlation method is related to Finlayson's chromaticity version of gamut mapping ("Colour in Perspective") [40]. First, since the algorithm chooses an illuminant among the expected ones, Finlayson's illumination constraint is built in. Second, a specific version of colour by correlation can be seen as quite close to the colour in perspective algorithm [62].

# 2.3.6 Neural Network Colour Constancy

Recently good results have been achieved using a neural net to estimate the chromaticity of the illuminant [63-67]. Here a neural net is trained on synthetic images randomly generated from a database of illuminants and reflectances. The scenes so generated may include synthetically introduced specularities [64]. In the work reported so far, rg chromaticity space is divided into discrete cells and the presence or absence of any image chromaticity within each of the cells is determined. This binary form of a chromaticity histogram of an image is used as the input to the neural network. During training the input corresponding to the generated scenes is presented to the network together with the correct answer. Back-propagation is used to adjust the internal weights in the network so that it thus learns to estimate the illuminant based on the input.

# 2.3.7 Methods Based on Specularities

If a surface obeys the dichromatic model discussed above, then the observed RGB responses to that surface under a fixed illumination will fall in a plane. This is because the possible colours are a combination of the colour due to the body reflection, and the colour due to the interface reflection, with the amounts of each being a function of the geometry. Mathematically, the k<sup>th</sup> sensor response,  $\rho^k$ , can be expressed as:

$$\rho^{k} = \int (m_{i}(i, e, g)S_{i}(\lambda)E(\lambda)R_{k}(\lambda) + m_{b}(i, e, g)S_{b}(\lambda)E(\lambda)R_{k}(\lambda))d\lambda$$
(2.12)

which becomes:

$$m_{i}(i, e, g)\rho_{i}^{k} + m_{i}(i, e, g)\rho_{b}^{k}$$
(2.13)

and using vector notation becomes:

$$\boldsymbol{\rho} = \mathbf{m}_{i}(\mathbf{i}, \mathbf{e}, \mathbf{g})\boldsymbol{\rho}_{i} + \mathbf{m}_{i}(\mathbf{i}, \mathbf{e}, \mathbf{g})\boldsymbol{\rho}_{b}$$
(2,14)

Thus the possible RGB responses,  $\rho$ , are a linear combination of the interface RGB,  $\rho_i$ , and the body RGB,  $\rho_b$ , and thus lie in a plane through the origin.

In the case of dielectrics, the interface function,  $S_i(\lambda)$ , is a constant, and thus the colour due to the interface reflection is the same as the illuminant,  $\varepsilon$ . If two or more such surfaces can be identified with different body reflections, then the RGB of each will fall into a different planes, and those planes will intersect in the illuminant direction  $\varepsilon$ . A number of authors have proposed colour constancy algorithms based on this idea [20, 68-73]. An obvious difficulty is recognizing the surfaces as such. If the observed RGB are projected onto an appropriate two-dimensional chromaticity space such as rg chromaticity, then the projected points for the surfaces present become line segments which intersect at a common point, specifically the chromaticity of the illuminant. Starting from each colour edge point found by conventional means, Lee [69] collects pixels in the direction of the greatest gradient in the green channel, until another edge point is reached. Each such collection of pixels gives an estimate of a line segment, and an estimate of the intersection points of the line segments is used as the final illuminant chromaticity estimate. A slightly different approach is to look directly for the structure of lines convergent on a point in chromaticity space [73].

The colour histograms due to dichromatic reflection have additional structure which may be exploited to identify such surfaces or highlights. Given a specific viewing geometry, highlights occur at a narrow range of surface normals, and thus combine with a specific amount of body reflection. Therefore the histograms consist of a line through the origin for the body reflection, together with a branch for the specular reflection departing from the colour of the body reflection at the particular angle where specular reflection occurs—the so called "dog-leg" [74, 75]. Further analysis reveals that the specular part of the histogram spreads out where it meets the body part, the degree of spreading, accompanied by a shortening of the specular segment, being a function of the surface smoothness. Finally, the location of the merging of the two parts is a function of the viewing geometry [76-78]. In [76], Klinker et al use these finer points of the histogram structure for the combined segmentation and illumination determination of images of dielectrics.

Nayar et al [79] manage to dodge the inherent segmentation problem by using polarization together with analysis of the observed colour along the lines discussed above. Polarization is an effective tool because specular reflection from dielectrics has different polarization than the body reflection.

Another method which is less dependent on segmentation, since it can work on a single region segmented very conservatively, is provided in [80]. Here, the difference in the nature of the spatial variation of the specular and diffuse illumination is exploited. Specifically, specular illumination is expected to vary much more rapidly, and Lee fits a one parameter model, derived from the dichromatic model, which maximizes the smoothness of the diffuse illumination. The method can combine the results from multiple regions, again, with conservative segmentation. It should be noted, however, that this promising method has only been tested on synthetic data. A related approach is to fit the observed RGB of a surface to a Lambertian model using robust statistics [81].

Finally, one general difficulty with methods based on specularities should be mentioned. Specularities tend best to reveal the colour of the illuminant where they strongly reflect that illuminant. This means that such specular regions tend to be very bright, often exceeding the dynamic range of a camera, and are thus unusable.

### 2.3.8 Methods using Time Varying Illumination (multiple views)

If we have access to images of the same scene under two or more illuminants, then we have more information about the scene and the illuminants. To see this, suppose we are trying to recover 3 parameters for both the surfaces, and the illuminants, that there are M surfaces in the scene, and that we have 3 camera sensors. Then, one image presents us with 3+3M unknowns, and 3M measurements. However, two images presents us with 6+3M unknowns, but 6M measurements. Assuming that the unknowns are not overly correlated, this is clearly a more favorable situation.

As already mentioned above, D'Zmura and Iverson [60] have extended the Maloney-Wandell algorithm for this circumstance. In addition, Tsukada and Ohta worked with the equations implied in the preceding paragraph in the case of two surfaces [82]. This yields 12 measurements to estimate 12 parameters, which become 10 parameters if brightness is normalized. Unfortunately, 3 of the measurements are quite correlated with the others, so the method is not particularly stable. The stability of the method can be improved by restricting the illuminant to CIE daylight [83].

### 2.3.9 Methods using Spatially Varying Illumination

The illumination falling on scenes often varies spatially due to the interaction of different illumination sources with the three dimensional world. For example, consider a white ball lying on a sunlit lawn. Part of the ball faces the sun, and receives mostly the yellow illumination of the sun, with some contribution from the blue sky. As we move around the ball, the contribution from direct sun becomes less, and the distinctly blue contribution from the sky becomes more extreme. In the self-shadowed part of the ball, the illumination is purely that from the sky. As a further example, near the lawn, the ball is also illuminated by light reflected from the lawn which is green in colour.

If we can identify a surface which is illuminated by varying illumination, then we have a situation similar to the time varying illumination case discussed in the preceding section. Specifically we have the response of that surface under more than one light. Thus we potentially have more data available to solve for the illumination. It should be clear that any algorithm based on multiple views can be modified to exploit the varying illumination. However, despite the fact that varying illumination is common, there are very few algorithms designed to exploit the extra information available.

As mentioned earlier, Retinex based methods discard slowly spatially varying illumination, thus achieving some robustness in this case, but they do not exploit the varying illumination. In [58], Finlayson et al provide an algorithm along the lines of [32, 40], but for the varying illumination chromaticity case. Using the observation that the chromaticities of illuminants are restricted, the authors show that the magnitude of the illuminant chromaticity changes can be used to constrain the actual illuminant chromaticity. For example, suppose common illuminants are less blue than some maximal blue, denoted by B. Now suppose that going from point X to point Y, the amount of blue doubles. Then the amount of blue at X can be at most one half B. If it were to exceed one half B, then the amount of blue at Y would exceed B, and this would break the assumption that the scene is illuminated by common illuminants.

In [58], a limited set of illuminants was used, and the gamut of the reciprocals of their chromaticities was approximated by a straight line. Furthermore, no attempt was made to identify the varying illumination. In [8, 84] a more comprehensive set of illuminants was used. In addition, the algorithm was modified so that it could be used in conjunction with the gamut mapping algorithms developed for the uniform illumination case [32, 40]. The idea here is that once the varying illumination has been identified, the image can be mapped to one which has uniform illumination, and thus provides constraints on the illumination due to the surfaces. These constraints are used in conjunction with the constraints found due to the varying illumination.

Also in [8, 84] a method was introduced to identify the varying illumination in the case of slowly varying illumination. The method is based on the assumption that small spatial changes are due to illumination changes (or noise), and that large changes are due to changes in surfaces. Using this assumption, a conservative segmentation is produced. A perfect segmentation is not needed. Specifically, it does not matter if regions of the same surface colour are combined, or if some regions are split, although too many spurious segments will degrade the recovery of the illumination. Given the segmentation, the varying illumination within a segment is easily determined, and a method is provided to robustly combine these variations into an estimate of the varying illumination field for the entire image.

### 2.3.10 Methods using Mutual Illumination

A special case of varying illumination is mutual illumination. Mutual illumination occurs when two surfaces are near each other, and each reflect light towards the other. For example, consider an inside corner which is the meeting of a red surface and a blue surface, illuminated by a white light. Then the red surface near the corner will be somewhat blue near the junction due to the reflection of the white light from the nearby blue surface. Similarly, the blue surface will also have some added red near the junction.

If mutual illumination can be recognized, then it can be exploited for colour constancy. For example, Funt et al [85] showed that if the mutual illumination between two surfaces could be identified as such, then this effectively added a sensor to the Maloney-Wandell algorithm, potentially increasing its efficacy. And in [86] the authors exploit the observation that the colours of a surface exhibiting mutual illumination are a linear combination of the two-bounce colour and the one-bounce colour. Two such planes due to a pair of mutually reflecting surfaces will intersect along the two-bounce colour, and using this information it is possible to solve for the one-bounce colour of the illuminant).

### 2.3.11 Methods for Object Recognition and Image Indexing

An important application of colour constancy processing is for illumination invariant object recognition, and its weaker cousin, image indexing. Image indexing treats images as the objects to recognized, with the canonical task being finding a test image in a database of images. As discussed in the introduction, both these problems are sensitive to the illumination, and the performance of corresponding algorithms increases with effective removal of illumination effects. To remove the illumination, any of the methods discussed above can be used. However, algorithms have also been developed which take advantage of the nature of the task. Specifically, these algorithms look for known objects, and thus they exploit knowledge about what they are looking for. I will now discuss some of these algorithms.

In [87], Matas et al model each of the objects in their test database under the range of expected illuminations. Modeling known objects in the presence of a variety of expected illumination conditions is also used in [88]. In [87] each surface on a specific object is represented by a convex set of the possible chromaticities under the range of possible illuminations. The occurrence of a chromaticity in this range is a vote for the presence of the object. In this manner, the likelihood of the presence of each object can be estimated. In [89] the authors integrate colour edge adjacency information into their object recognition scheme, and use Nayar and Bolle's [90] intensity reflectance ratio as an illumination invariant quantity in each of the three channels (see also [91]). This invariant is based on the assumption that the illumination is usually roughly constant across a boundary, and under the diagonal model the RGB ratios will be a constant across the junction of a given surface pair. To avoid problems with small denominators, Nayar and Bolle defined their reflectance ratio as (a-b)/(a+b) instead of (a/b).

Image indexing is simpler than these general object recognition approaches because it avoids the difficult problem of segmenting objects from the background. Image indexing can be used for object recognition and localization by exhaustively matching image regions. This clearly requires indexing to be fast and robust with respect to the inclusion of background as well as pose and scale. Nonetheless, the original work [41] was proposed as an object recognition strategy based on overcoming these difficulties. This method matched images on the basis of colour histograms. As the colour histogram of an image is dependent on the illumination, Funt and Finlayson [92, 93] proposed an illumination invariant version based on matching histograms of the ratios of RGB across surface boundaries. The histograms are computed directly (without segmentation) from the derivative of the logarithm of the image, after values close to zero have been discarded. Another illumination invariant approach is to simply "normalize" both the images in the database, and the test image [94-96]. Under the diagonal model, the image is scaled by the RGB of the illuminant. Any normalization of the RGB which coincides with the scaling due to the illuminant will be illumination invariant. For example, the image may be normalized by the average RGB. This is like using the grey world algorithm, but now, because of the image indexing context, the "world" is precisely known—it is the image.

# 2.4 Conclusion

Modeling scene illumination is an important problem in computer vision. This claim is supported by the existence of a large body of work addressing this problem. This work has lead to improvements in image understanding, object recognition, image indexing, image reproduction, and image enhancement. Nonetheless, much more work is required. One main problem is the development of algorithms for real image data. Most of the algorithms discussed above have quite specific requirements for good results, and those requirements are not met in most real images. Furthermore, even if the requirements are met, they are not verifiable. Preliminary work suggests that the key to progress is better overall models which include more of the physical processes which impact the images. For example, by modeling varying illumination, algorithms have been developed which are not only robust with respect to varying illumination, but can use the varying illumination for better performance. The same applies to specular reflection. Models for real images must be comprehensive, because we cannot always rely on the existence of certain clues such as varying illumination or specularities. Furthermore, both these cases have connections to other computer vision problems such as segmentation and determining scene geometry from image data. Invariably, progress in these areas both aids modeling the scene illumination, and is aided by modeling the scene illumination. Thus there are great opportunities for progress using more sophisticated and comprehensive physics bases models of the interaction of scene with illumination.

# **Chapter Three**

# Camera calibration for colour constancy research

The image recorded by a camera depends on three factors: The physical content of the scene, the illumination incident on the scene, and the characteristics of the camera. Since the camera is an integral part of the resulting image, research into image understanding normally requires a camera model. Given such a model we need to verify that it is in fact adequate for a particular vision system. Furthermore, once we have the particular parameters of the model for a given camera, we can use the model to predict what the camera will see, given an input spectral distribution. This has applications in the development and practical realization of colour-related image processing algorithms, such as computational colour constancy algorithms.

We first introduce the standard camera model used in colour-oriented computer vision. We then discuss previous methods for fitting the parameters of that model, and then introduce a new method for obtaining these parameters. We then provide the results of our camera calibration experiments.

# 3.1 The Camera Model

The goal of this work is to develop a model which predicts image pixel values from input spectral power distributions. In this section we discuss the general form of the model. For the moment we assume that all camera controls such as aperture are fixed. Let  $v^{(k)}$  be the value for the k'th channel of a specific image pixel and let  $L(\lambda)$  be the spectral power distribution of the signal imaged at that pixel. Then we model image formation by:

$$\rho^{(k)} = F^{(k)}(\upsilon^{(k)}) = \int L(\lambda) R^{(k)}(\lambda) d\lambda$$
(3.1)

where R<sup>(k)</sup> is a sensor sensitivity function for the k'th channel, and F<sup>(k)</sup> is a wavelength independent linearization function. This model has been verified as being adequate for computer vision over a wide variety of systems [6-10]. This model is also assumed for the human visual system, and forms the basis for the CIE colorimetry standard.

As we move around the image plane the signal is attenuated due to geometric effects, notably vignetting [12, page 26], and a fall-off proportional to the fourth power of the cosine of the off axis angle [12, page 208]. These effects can be considered to be absorbed into either  $R^{(k)}$  or  $F^{(k)}$ . In our calibration experiments we deferred these considerations by using only the central portion of the image.

Similarly, effects on the overall magnitude of the response can also be considered to be absorbed into either  $R^{(k)}$  or  $F^{(k)}$ . The two most pertinent effects on overall magnitude are the size of the aperture and the focal length. In fact, for much work in colour, absolute brightness is somewhat arbitrary, being under aperture control, and usually adjusted by the user or the camera system to give a reasonable image. For this reason, work in colour often uses a chromaticity space which factors out brightness. The most common such space is (r,g) defined by (R/(R+G+B), G/(R+G+B)). In chromaticity space geometric attenuation effects can be ignored. On the other hand, if absolute brightness is important, then these effects have to be either controlled or calibrated for.

Successful use of the above model requires sufficient consideration of the function  $F^{(k)}$ .  $F^{(k)}$  reverses any added gamma correction, and subtracts any camera black, as well as correcting for other more subtle non-linearities which may exist. Even if  $R^{(k)}$  is not required for an application,  $F^{(k)}$  must usually be taken into consideration. For example, reliably mapping into a chromaticity space such as (r,g), requires either an estimate of  $F^{(k)}$ , or confidence that it is the identity function and thus can be ignored.

For the practical application of the above model, the continuous function of the wavelength,  $\lambda$ , are replaced by samplings of those functions. For example, our data is collected with a PhotoResearch PR-650 spectraradiometer, which measures data from 380nm to 780nm in 4nm steps. The function  $L(\lambda)$  then becomes the vector **L**,  $R^{(k)}(\lambda)$  becomes the vector **R**<sup>(k)</sup>, and equation (3.1) becomes:

$$\rho^{(\mathbf{k})} = \mathbf{F}^{(\mathbf{k})}(\upsilon^{(\mathbf{k})}) = \mathbf{L} \bullet \mathbf{R}^{(\mathbf{k})}$$
(3.2)

Using this notation, camera calibration can be defined as finding  $F^{(k)}$  and  $\mathbf{R}^{(k)}$ .

# 3.2 Motivation for Camera Calibration

We have become interested in colour camera calibration as part of our research into computational colour constancy. Practically all algorithms for colour constancy assume that the image pixels are proportional to the input spectral power. This is equivalent to assuming either that  $F^{(k)}$  is the identity function, or that it is known and has been applied to the data. In other words, colour constancy algorithms require  $\rho^{(k)}$  as input, as opposed to the more readily available  $v^{(k)}$ .

Determining the function  $\mathbf{R}^{(k)}$  is also important for computational colour constancy. Most algorithms, including all the ones we currently think are the most promising, require an estimate of what the camera may see when it is used in the real world with its many different surfaces and illuminations. Although it is conceivable to obtain camera responses for a large number of surfaces under a given illuminant, it is impractical to obtain this data for each camera. Some algorithms would further require this data

for each possible illumination, including combinations of several sources. It is thus far more effective to first measure or obtain reflectance functions and illuminant spectra, and then to use a camera model to predict the wide range of camera responses required for these algorithms. If the camera model can be easily updated, then databases for reflectance and illuminant spectra can be continuously improved, shared, and re-used for colour constancy or similar problems on many different cameras.

# 3.3 Previous Work

Since  $F^{(k)}$  is independent of wavelength, it can be determined by stimulating the camera with varying intensities of a single light source, obtained with neutral density filters, or by simply moving the source. An appropriate function can then be fitted to the data, or alternatively a smoothed version of the data can be used to generate a look up table. Vora et all [9] used this method to verify that a Kodak DCS-200 digital camera was linear over most of its operating range, and also to develop a linearization curve for a Kodak DCS-420 digital camera. They then determined  $\mathbf{R}^{(k)}$  for those cameras by stimulating them with very narrow band illumination produced by a monochrometer [10]. This method is conceptually very simple, and if done carefully, it can be very accurate. However, the equipment required to produce sufficiently intense narrow band illumination at uniformly spaced wavelengths is expensive and not readily available. Hence various researchers have investigated methods for calibration which do not use such equipment.

The general approach of these methods is to first measure  $F^{(k)}$ , and then to measure a number of input spectra and the corresponding camera responses. Let  $\mathbf{r}^{(k)}$  be a vector whose elements are the linearized camera responses  $\rho^{(k)}$  and let L be a matrix whose rows are the corresponding sampled spectra. Then (3.2) becomes:

$$\mathbf{r}^{(k)} = \mathbf{L}\mathbf{R}^{(k)} \tag{3.3}$$

Equation (3.3) can be solved by multiplying both sides by the pseudo-inverse of L. However, this does not work very well because L is invariably rankdeficient. L is rank-deficient because we are now trying to determine R using easily obtainable input spectra, and these tend to be of relatively low dimensionality. If L was of full rank, then we would have a method analogous to the monochrometer method. Given that L is rank deficient, the results using this method are very sensitive to noise (since it is mainly the noise that is being fitted), and the resulting sensor responses tend to have numerous large spikes, and have an abundance of non-negligible negative values (see Figure 3.4).

Sharma and Trussell [6] improved the prospects for a reasonable solution by introducing various constraints on  $\mathbf{R}^{(k)}$ . First, instead of solving (3.3) exactly, they constrained the maximum allowable error as well as the RMS error. In addition, they constrained a discrete approximation of the second derivative to promote a smooth solution. Finally, they constrained the response functions to be positive. They then observed that the constraint sets are all convex, and so they computed a resulting constraint set using the method of projection onto convex sets.

Hubel et al [97] also recognized that some form of smoothness was necessary for a good solution, and they investigated the Wiener estimation method, as described by Pratt and Mancill [98], as a method for finding a smooth fit. They found that the method gave generally good results. They note, however, that the method produced negative lobes in the response functions, and mention briefly using the projection onto convex sets method to remedy this problem.

Sharma and Trussell's contribution was the starting point for some of our own work on this problem [8]. Rather than constrain the absolute RMS error, we chose instead to minimize the relative RMS error. We then rewrote Sharma and Trussell's other constraints so that the entire problem became a least squares fit with linear constraints for which there are standard numerical methods readily available. Once we had a fit for our camera sensors, we noted that they were essentially uni-modal, and that once the sensors dropped to a small value they remained small. On these grounds we also constrained the sensors to be zero outside a certain range on subsequent runs. In this particular case this forced the sensors curves to be uni-modal. This last step needs to be applied with care, as it is possible that the sensors are in fact non-zero beyond the points where the main peaks drop to zero.

Recently Finlayson et al also used a similar approach [99]. They constrained smoothness by constraining the sensors to be linear combinations of the first 9-15 Fourier basis functions. They also introduced a modality constraint expressed in terms of the peak location. They then found the best fit for the proposed modality by stepping through all possible peak locations. This method also requires care, as the modality is often unknown.

# 3.4 The Fitting Approach

We now describe our proposed fitting method in two stages. First, we will describe the basic method which estimates the response vector for each channel  $\mathbf{R}^{(k)}$ , on the assumption that the linearization function  $F^{(k)}$  has been found and applied. Second, we incorporate the estimation of  $F^{(k)}$  into the fitting procedure. This has the advantage that the error in the two fits can be traded off against each other, and data collected to find  $\mathbf{R}^{(k)}$  can also help in the effort to find  $F^{(k)}$ .

In initial work [8] we minimized the relative RMS error in equation (3.3) subject to positivity constraints, smoothness constraints, and constraint on the maximum allowable error (and/or relative error). We have since found that it is better to replace the constraint on smoothness with a regularization term added to the objective function. Thus we minimize the relative error and the non-smoothness measure together. This allows deviations from one to be traded against deviations of the other. With the hard constraint used previously, there is no recourse in the case that making the sensor response slightly less smooth at a particular location can substantially reduce the error. Similarly, there is no recourse when a small increase in error beyond the hard limit can substantially increase the smoothness.

We choose to minimize the relative RMS error for two reasons. First, as discussed in more detail below, the standard deviation of the R, G, and B values used for calibration increases with their magnitude. And second, we have found that minimizing relative error better reduces the error in chromaticity, which is difficult to minimize directly, but is often of most interest, as discussed above. However, for some applications minimizing absolute error, or even a weighted combination of both, may make more sense. We have also found that it is not generally necessary to use Sharma and Trussell's constraint on maximum allowable error to get good results, but again, limiting either the absolute error or relative error may be called for in some cases, and is easily added to the method.

To investigate the error in our pixel values, we made 100 measurements at various intensities of the R, G, and B average of 24 different 50 by 50 windows, each exposed to uniform stimulus. We computed the mean and the variance over the 100 measurements for each intensity level. We estimate the actual intensity by the mean of the pixel values, linearized by the method described later in this paper. The results for the red channel are plotted in Figure 3.1. We express the variance as the sum of the variances which is dependent on intensity and the various sources of other variance. We further assume that the intensity dependent variance is due to photoelectron shot noise, and thus is proportional to the mean [100]. Thus we expect that the observed variance is proportional to the intensity. This is more or less consistent with the graph in Figure 3.1.

We now provide the details of the fitting procedure. The preferred formulation is somewhat driven by the software package which will be used to solve the problem. However, a concrete example will likely help to clarify the method. We begin with a formulation which minimizes absolute error. Let N be the number of spectral samples being used. First, we form the N-2 by N second derivative matrix S:

$$\mathbf{S} = \begin{vmatrix} -1 & 2 & -1 & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & & \\ & & & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & & -1 & 2 & -1 \end{vmatrix}$$
(3.4)

Then we solve

$$\left|\frac{\mathbf{L}}{\lambda \mathbf{S}}\right| \mathbf{R}^{(k)} = \left|\frac{\mathbf{r}^{(k)}}{\mathbf{0}}\right| \tag{3.5}$$

in the least squares sense, subject to linear constraints. The upper part forms the error term

$$\sum_{i} \left( \mathbf{L}_{i} \bullet \mathbf{R}_{i}^{(k)} - \boldsymbol{\rho}_{i}^{(k)} \right)^{2}$$
(3.6)

and the lower part forms a term for smoothness. The coefficient  $\lambda$  specifies the relative weight attributed to the two terms. If  $\lambda$  is zero and there are no constraints, then this becomes the pseudo inverse method. A serviceable

### Variance of red channel data versus intensity



**Figure 3.1**: The variance of red channel measurements verses their values for data similar to that ones used for calibration.

value for  $\lambda$  is easily found by trial and error. To ensure positivity, we use the constraint:

$$\mathbf{R}^{(k)} \ge \mathbf{0} \tag{3.7}$$

To specify that the sensor response is zero outside the range [min, max] we can add the constraint:

$$R_i^{(k)} \le 0 \quad \text{for } i < \min, i > \max$$
(3.8)

To specify that the absolute error is no more than a specified positive value,  $\delta$ , we can add the constraint:

$$\rho_i^{(k)} - \delta \le L \mathbf{R}^{(k)} \le \rho_i^{(k)} + \delta \tag{3.9}$$

To minimize the relative error we need to replace (3.6) by:

$$\sum_{i} \left( \frac{\mathbf{L}_{i} \bullet \mathbf{R}_{i}^{(k)} - \rho_{i}^{(k)}}{\rho_{i}^{(k)}} \right)^{2} = \sum_{i} \left( \frac{\mathbf{L}_{i} \bullet \mathbf{R}_{i}^{(k)}}{\rho_{i}^{(k)}} - 1 \right)^{2}$$
(3.10)

One way to express this is to use a modified version of L,  $L_{rel}^{(k)}$ , which is simply the rows of L divided by the corresponding sensor response. Formally,  $L_{rel}^{(k)}$  is given by:

$$L_{\text{rel}}^{(k)} = (\text{diag}(\mathbf{r}^{(k)}))^{-1} \bullet L$$
(3.11)

We then replace (3.5) with:

$$\left|\frac{\mathbf{L}_{\text{rel}}^{(k)}}{\lambda S}\right| \mathbf{R}^{(k)} = \left|\frac{\mathbf{I}}{\mathbf{0}}\right|$$
(3.12)

Finally, if we require a constraint limiting the relative error to less than a positive amount  $\zeta$ , we can use:

$$1 - \zeta < \mathcal{L}_{rel}^{(k)} \mathbf{R}^{(k)} < 1 + \zeta \tag{3.13}$$

We note that minimizing the relative error may need to be modified slightly to deal with very small  $\rho^{(k)}$ . Such data is likely to be inaccurate for other reasons. For example, if there is a camera offset due to camera black, then small values of  $\rho^{(k)}$  include error from  $F^{(k)}$ . Thus we need to either ignore small values of  $\rho^{(k)}$  or give the corresponding data row less weight in the fitting process. Equation (3.12) can be interpreted as a weighted version of equation (3.5), with the weighting being inversely proportional to  $\rho^{(k)}$ . Thus it is natural and easy to put an upper bound on this weighting, and this is how we safeguard against small  $\rho^{(k)}$  when we do not want to exclude them outright.

The method so far assumes that the function  $F^{(k)}$  has been found and applied. As mentioned above,  $F^{(k)}$  can be found by fitting the response as a function of brightness. However, the body of data collected to find  $\mathbf{R}^{(k)}$  also contains information about  $F^{(k)}$ , and since this data set needs to be comprehensive, it makes sense to use it for the final determination of  $F^{(k)}$ . Therefore we propose fitting  $\mathbf{R}^{(k)}$  and  $F^{(k)}$  together. This has the advantage that the errors in  $F^{(k)}$  and  $\mathbf{R}^{(k)}$  can be traded against each other. We first make a rough measurement of  $F^{(k)}$  and use it to develop a parameterized expression for it. We then fit the parameters for  $F^{(k)}$  and  $\mathbf{R}^{(k)}$  simultaneously. We will now provide a specific example of such a strategy.

The Sony DXC-930 camera which we used for our experiments is quite linear for most of its range, provided it is used with gamma disabled. However, it has a non-negligible offset and a slight non-linearity for small pixel values. Due to this non-linearity, the offset is not the camera black, and using the camera black for the offset leads to errors in chromaticity. Therefore the non-linearity must be taken into account, even if it is not explicitly fitted. Figure 3.2 shows the slight non-linearity for the red channel. The other two channels are similar.





### CAMERA CALIBRATION FOR COLOUR CONSTANCY RESEARCH

The linear fit shown in Figure 3.2 was found using a simple linear fit on the calibration data using pixel values greater than 30. This illustrates the point that linearization information is available in the data set which is to be used to find  $\mathbf{R}^{(k)}$ . If we forced a linearization function found using less data onto the fit for  $\mathbf{R}^{(k)}$ , then this information would be disregarded. To proceed with our strategy, we need to parameterize the non-linearity. The particular form of the parameterization is somewhat arbitrary and will vary substantially from case to case. With a little experimentation we found that the non-linearity for our camera could be approximated by:

$$F^{(k)}(x) = x - a_0^{(k)} - a_1^{(k)} e^{-C_k (x - b_k)}$$
(3.14)

where  $b_k$  is the camera black for channel k, and  $C_k$  is a constant which must be found by trial and error, but was found to be quite stable. If we used the simpler form:

$$F^{(k)}(x) = x - a_0^{(k)}$$
(3.15)

then we would simply be fitting a camera offset simultaneously with  $\mathbf{R}^{(k)}$ . This would be a reasonable approach for our camera if we did not wish to use smaller pixel values. In general, the parameters of the approximation function must generate a reasonable collection of curves which roughly fit the non-linearity so that the overall fitting procedure can find a good estimation of  $F^{(k)}(x)$ . In addition, the parameters which are fitted must be linear coefficients. For example, we can only directly fit for  $a_0^{(k)}$  and  $a_1^{(k)}$ ;  $C_k$ must be found by trial and error.

To find the parameters for the approximation of  $F^{(k)}(x)$  simultaneously with **R**<sup>(k)</sup> when fitting for absolute error, we replace equation (3.5) with:

$$\left| \frac{\mathbf{L}}{\lambda S} \right| \frac{1}{0} \frac{e^{-C_{k}(\mathbf{r}^{(k)} - \mathbf{b}_{k})}}{0} - \left| \mathbf{e}^{\mathbf{R}_{k}^{(k)}} \right| = \left| \frac{\mathbf{r}_{0}^{(k)}}{\mathbf{a}_{1}^{(k)}} \right| = \left| \frac{\mathbf{r}_{0}^{(k)}}{\mathbf{0}} \right|$$
(3.16)

where the arithmetic in the upper right block of the matrix is done elementwise as needed. Similarly, in the case of fitting for relative error, we replace (3.12) with:

where again, the arithmetic in the upper right block of the matrix is done element-wise as needed. Note that the response vectors  $\mathbf{r}^{(k)}$  now correspond to the observed camera responses  $v^{(k)}$  in equation (3.2) in contrast to the earlier formulation where  $\mathbf{r}^{(k)}$  corresponded to the linearized camera responses,  $\rho^{(k)}$ .

In all cases the entire fitting procedure is a least squares minimization problem with linear constraints, or equivalently, it can be viewed as a quadratic programming problem. Such problems can be solved with standard numerical techniques for which software is readily available. We use the freely available SLATEC fortran library routine DBOCLS. The routine DLSEI in that library may also be used. A third option is the Matlab routine "qp".

# 3.5 Experimental Method

In order to investigate and ensure the robustness of the calibration we endeavored to obtain a large set of input spectra and their corresponding camera responses. Thus we automated the data collection. Our target was a Macbeth colour checker which has 24 different coloured patches which we illuminated with a number of illuminant/filter combinations. The main criteria of the setup is to ensure that the camera and the spectraradiometer measure the same signal. Furthermore, we wanted the camera data for each patch to be from the center of the image. Therefore we mounted the colour checker on an XY table which moved it under computer control. The camera and the spectraradiometer were mounted on the same tri-pod. Rather than aim them simultaneously at the target, we decided instead to set the optical axes to be parallel. This meant that the tri-pod had to be raised and lowered between capturing camera data and spectraradiometer data. Thus we captured an entire chart worth of camera data before capturing an entire chart worth of spectra. A total of 26 illuminant/filter combinations were used. A few measurements were eliminated due to pixel clipping, yielding a total of 612 measurements.

We took additional steps to reduce the error. As indicated above, it is important that the camera and the spectrometer are exposed to the same signal. In order to minimize the effect of misalignment, we made the illumination as uniform as possible. We extracted a window from the image which corresponded as closely as possible to the area used by the spectrometer. The pixels in this window were averaged. The camera measurements were further averaged over 50 frames. The spectrometer measurements were averaged over 20 capture cycles. Finally, in order to reduce the effect of flare, the entire setup was viewed through a hole in a black piece of cardboard, exposing the region of interest, but as little else as was practical. Some of the periphery was exposed to aid alignment.

# 3.6 Results

We experimented both with minimizing the absolute error and the relative error. As mentioned above, minimizing the relative error is arguably more suitable for our data and our purposes, and the results support this. For example, minimizing relative error was substantially better for reducing the absolute error in (r,g) chromaticity, which is difficult to minimize directly. Furthermore, minimizing relative error gave better distributed errors with fewer outliers and less indications of systematic problems, especially in the case of (r,g). In terms of straight RGB error, however, the methods are essentially symmetric. Fitting for relative error naturally gives the lowest relative error, about 20% less than fitting for absolute error. Similarly, fitting for absolute error gives the lowest absolute error, also about 20% less than that obtained by fitting for relative error.

The sensors computed using the new fitting method are shown in Figure 3.3. To compute the sensor curves shown, the range constraint in equation (3.8) was not used. If we use the range constraint to force the curves to be zero once they drop to zero, then we get sensor curves which are visually very close to the ones shown, except of course, they are exactly zero outside the main peak. Adding these range constraints increase the RMS relative error from 1.02% to 1.24% which is more than we expected. Thus either the camera really does have small responses outside the main peaks, or, more likely, there is some source of error which we have not yet isolated. Two possibilities to consider are flare and fluorescence in some of the surface reflectances.

The fitting process produces both the sensor curves, and the linearization function. We can use this linearization function to linearize the data, and further compare fitting methods. Figure 3.4 shows the sensor curves obtained using the pseudo inverse method on the linearized data (lambda=0 and no constraints). As expected, the result has a large number of extreme spikes. Figure 3.5 shows the sensor curves obtained using the pseudo inverse method with positivity constraints. Table 3.1 reports the RMS relative errors of the various methods.



**Figure 3.3**: The sensor response found using the fitting method introduced in this paper without the range constraints in equation (3.8). Two of the sensors have small responses outside the main peaks that can be removed by including the range constraints set to the obvious boundaries of the main peaks. The resulting sensors are otherwise very similar.



**Figure 3.4**: The sensor response found using the pseudo inverse method applied to linearized data. This method minimizes the error, but takes nothing else into account. Since the matrix is rank-deficient, the method mainly fits the noise in the data. The resulting sensors are clearly incorrect, and if they are used on different data, the error will be very large.



Pseudo-Inverse with Positivity Constraints

**Figure 3.5**: The sensor response found using the pseudo inverse method with positivity constraints applied to linearized data. The spikes conform to the rough outline of the sensors, and the result is surprisingly robust (see Figure 3.6). Nonetheless, there is no reason to use these sensor responses, as they are clearly incorrect, and the results plotted in Figure 3.6 indicate that the error on different data will be larger than the other methods.

Fitting Method	RMS relative error (percent)	RMS absolute error	RMS absolute error in r=R/(R+G+B)	RMS absolute error in g=G/(R+G+B)
New fitting method with linearity fitting	1.02	1.09	0.0020	0.0017
New method with linearity fitting and range	1.24	1.11	0.0029	0.0023
Pseudo inverse method on linearized data	0.93	0.98	0.0071	0.0014
Pseudo inverse method positivity constraints	1.21	1.02	0.0023	0.0015
New method on linearized data without positivity	1.28	1.05	0.0025	0.0018
New method on linearized data	1.39	1.03	0.0024	0.0018
New method on linearized data with range	1.55	1.13	0.0027	0.0023

**Table 3.1**: Error obtained on 612 data points using various fitting methods. The first two methods fit for both linearity and the sensor curves. The last five methods are applied to linearized data.



#### Logarithm of relative RMS error versus logarithm of sample size

**Figure 3.6**: RMS relative fitting error versus sample size (log scales). The methods which are more constrained are more robust. The unconstrained pseudo inverse method (first curve), degrades rapidly with decreasing sample size. A fit with smoothness, but without positivity (second curve) also degrades quite rapidly. This indicates that promoting smoothness alone cannot ensure a robust result. Interestingly enough, positivity without smoothness gives a fairly robust result (third curve). This graph also shows the improved performance of the fitting method which includes fitting the linearization function (the two darker curves).
#### CAMERA CALIBRATION FOR COLOUR CONSTANCY RESEARCH

As expected, for the five results on the same (linearized) data, as constraints are added, the error increases. In the extreme case, the pseudo inverse method has the lowest error, but the sensor curves are obviously not correct. As mentioned above, because the matrix is rank-deficient, this method fits the noise of the specific data set. This becomes evident when we look at how well the fitting methods perform when run on a subset of the data. To test this, we ran each of the methods on sub-sets of the data of sizes 400, 200, 100, 50, 25, and 15. The RMS error was computed using all 612 measurements. The result for each sample size was the average of the results for several hundred randomly selected samples. The results are shown in Figure 3.6. The pseudo inverse method degrades very rapidly with decreasing sample size, verifying that the fits are largely an artifact of the chosen data set. If positivity constraints are added to the pseudo inverse method, then the results are much better and surprisingly robust. However, the error quickly goes from slightly less than that for the more reasonable smooth fit, to somewhat more, and thus there is no reason to use such a method. The other methods tested are quite robust, degrading slowly and reasonably with decreasing sample size.

Figure 3.6 also shows that the addition of the range constraint also aids robustness. This result is somewhat artificial, as the range constraints used were those determined from the full data fits, but nonetheless the graph illustrates how constraints become more critical as the amount of available data decreases.

### 3.7 Conclusions

We have developed and tested a new method for fitting the standard camera model used in colour research. By promoting smoothness, and using constraints on the sensor response functions such as positivity, we obtain a result which is both reasonable and robust. We have found that it is best to promote smoothness by adding a regularization term to the minimization expression rather than constraining it, as has been done in earlier work by others and ourselves. We have also investigated fitting a small non-linearity in the camera response simultaneously with the sensor response curves. This was found to be effective because the errors in the two components of camera fitting could be traded against each other for a better fit. This also allowed the linearization data inherent in the calibration data set to be used to an advantage. If we forced a linearization function obtained with less data onto the sensors fit, then this information would be lost. Finally, we claim that it is often preferable to minimize the relative error, especially if chromaticity accuracy is more important than overall accuracy.

# **Chapter Four**

# A Comparison of Computational Colour Constancy Algorithms

In this work we test the leading computational colour constancy algorithms on both synthetically generated data and on a large collection of calibrated images. The algorithms chosen for study include several grey world methods, a version of the Retinex method, a number of variants of Forsyth's gamutmapping method, Cardei et al's neural net method, and Finlayson et al's Colour by Correlation method. We investigate the ability of these algorithms to make three different estimates of colour constancy related quantities from image input. These are the chromaticity of the scene illuminant, the overall magnitude of that illuminant, and a corrected, illumination invariant, image. We investigate this performance under the influence of several parameters. In the case of synthetic data, we first look at the performance of the algorithms as a function of the number of surfaces in scenes generated from reflectance spectra. We then study the effect of adding specular reflections to the synthetic scenes. Finally we investigate the effect of clipping the brightest specularities, as specular data is often clipped due to limited dynamic range.

For testing on real image data we use a collection of 321 images of a variety of scenes under 11 carefully chosen illuminants. We extend the

dynamic range of the images with frame averaging so that we can investigate the possibilities for computational colour constancy with a high dynamic range vision system. We also test the algorithms under the conditions of more limited dynamic range by artificially clipping the data. Finally, we test the algorithms under a variety of pre-processing methods. We then analyze the results of testing on real image data in the context of the synthetic data experiments.

Our synthetic results indicate that the methods which emphasize the use of input data statistics, specifically Colour by Correlation and the neural net algorithm, are potentially the most effective at estimating the chromaticity of the scene illuminant. Unfortunately, we were unable to realize comparable performance on real images. Here we found that the best performance was provided by three-dimensional gamut-mapping algorithms. Hence we conclude that in order to reap the potential benefits of the statistically based algorithms, we need a better understanding of the statistics of real image data. Unless such an understanding is available for a particular problem domain, this work indicates that the three-dimensional gamutmapping algorithms are likely to be the best choice.

### 4.1 Approaches to Colour Constancy

Computational colour constancy has been most commonly expressed as the task of finding illuminant invariant descriptors of the scene surfaces. This is often broken into two steps. The first step is to estimate illuminant parameters, and then a second step uses those parameters to compute illumination independent surface descriptors [42, 56]. These illuminant independent descriptors can then be used for a wide range of computer vision tasks, such as colour sensitive object recognition. In such applications the descriptors can be quite abstract, but if one is using colour constancy for image reproduction, then eventually an image of interest to humans must be produced. In this application the input is an image which may have a colour cast, due to a mismatch between the scene illumination and the illuminant for which the camera is balanced. The output is the corrected image, or if the

goal is automatic camera balancing, then the output is the appropriate camera balance. This more restricted form of colour constancy, where the illuminant invariant descriptors are tied to the output of the vision system, was formally introduced in [32]. Here the goal of colour constancy is reformulated as finding a mapping from the input image, which is of some scene under an unknown illuminant, to a second image, which is the camera response to the same scene under a standard, known illuminant. This standard illuminant is referred to as the canonical illuminant. The choice of the canonical illuminant is somewhat arbitrary. For image reproduction applications it makes most sense to use an illuminant for which the camera is balanced, and this is the choice we have settled on in general.

The two approaches can be made roughly comparable once a model of illumination change has been specified. By far the most common model of illumination change used for colour constancy is the diagonal model, and it is used by all of the algorithms for which we present results. The diagonal model maps the image taken under one illuminant, to the image taken under another illuminant, by simply scaling each channel independently. For concreteness, consider a scene with a white patch. Suppose that the camera response to the white patch under the unknown illuminant is  $\mathbf{\rho}^{\mathrm{U}} = (\rho_1^{\mathrm{U}}, \rho_2^{\mathrm{U}}, \rho_3^{\mathrm{U}})'$ , and that the response under the known, canonical, illuminant is  $\mathbf{\rho}^{C} = (\rho_{1}^{C}, \rho_{2}^{C}, \rho_{3}^{C})'$ . Then the response of the white patch can be mapped from the unknown case to the canonical case simply by scaling the i<sup>th</sup> channel by  $\rho_i^{\rm C}/\rho_i^{\rm U}$ . To the extent that this same scaling works for the other, non-white patches, we say that the diagonal model holds. The efficacy of the diagonal model is largely a function of the vision system sensors, specifically whether or not they are narrow band, and whether or not they overlap [34-37]. In the case of the camera used for the present work, the diagonal model is a good approximation. If the diagonal model leads to large errors, then it may be possible to get better results using sensor sharpening [101].

All of the algorithms studied here which use the first approach, namely illuminant parameter estimation, specifically estimate the camera response to a pure white, or a projection thereof (chromaticity). Using the diagonal model, the result of such an algorithm can be used to correct an image, or, using the terminology above, the appropriate diagonal map taking the image under the unknown illuminant to one under the canonical illuminant can be determined. Similarly, algorithms using the second approach, namely directly estimating transformations to the image under the canonical illuminant, imply an illuminant estimate. This illuminant estimate can be obtained by applying the inverse of the transformation to the camera response to white under the canonical, or a projection thereof. All the algorithms studied here which use the second approach are variants of Forsyth's method, and all restrict the sought transformation to diagonal maps, and thus inverting the transformation as described above is easy. Thus we see that we can convert the results from both approaches to either framework. In general, if an algorithm does well in one context, it does well in the other, but the correspondence is not exact. It is therefore prudent to look at the performance of the algorithms using several error measures.

We will now discuss a second classification of computational colour constancy algorithms, that being the number of parameters estimated. All of the algorithms tested in this work attempt to recover either 2 or 3 parameters<sup>1</sup>. The algorithms which recover 2 parameters will be referred to as chromaticity based, and the algorithms which recover 3 will be referred to as RGB based. This classification applies to both approaches discussed above.

If we consider the approach of estimating the camera response to the illuminant, it is arguably most natural to estimate the three parameters inherent in the (R,G,B) of a white patch under that illuminant. However, it is often the case that we are most interested in the chromaticity of the illuminant, and an estimate of that chromaticity will suffice. We remind the reader that chromaticity is colour normalized by overall magnitude—one chromaticity space is (r,g)=(R/(R+G+B), G/(R+G+B)). A number of color constancy algorithms have been developed which work entirely in some chromaticity space [40, 61-63, 65, 66], and much progress has been made by taking advantage of the simplifications afforded by this strategy. However, since these algorithms ignore the magnitude of the image pixels, they are potentially less powerful than algorithms which attempt to use information

<sup>&</sup>lt;sup>1</sup>The general case is N-1 or N parameters, where N is the number of camera sensors.

that may be implicit in those values. For example, it is commonly recognized that specular highlights carry information about the illuminant chromaticity [18, 20, 69, 70, 72, 73], and the fact that they are relatively bright may be of use to some algorithms.

For the purpose of comparing algorithms, we note that any algorithm which estimates the full (R,G,B) of the illuminant, also provides an estimate of the chromaticity of the illuminant. On the other hand, a chromaticity estimate cannot be converted to a full (R,G,B) estimate, unless an independent estimate of the illuminant magnitude is available. A chromaticity estimate is often sufficient because an illuminant magnitude is often implicitly present. For example, when a picture is taken, either a human operator or some mechanism has set the aperture to a reasonable value. Thus a correction for chromaticity, which leaves the overall brightness the same, is often sufficient for image reproduction applications. However, for machine vision applications, especially as cameras with higher dynamic range become available (see [102] for information about one high dynamic range camera), developing and using algorithms which go beyond the capabilities of current automatic apertures may well be worthwhile.

### 4.2 Error Measures

In order to compare fully algorithms with regard to the multiple issues of illuminant chromaticity estimation, illuminant brightness estimation, and accuracy of corrected images, more than one error measure is needed. We will use five in this paper. To measure illuminant chromaticity estimation we use two methods. The first considers the illuminant (R,G,B), and the corresponding estimate thereof, as vectors in RGB space, and computes the angle between these two vectors in degrees. The second is the vector distance in (r,g) space of the illuminant chromaticity and the estimate thereof. These two measures are roughly interchangeable, but the first is arguable more natural for the (R,G,B) algorithms, whereas the second is more natural for several of the chromaticity algorithms which work solely in (r,g) space. Furthermore, the error in "r" and "g" considered alone is used for looking at chromaticity error distributions.

To measure illuminant (R,G,B) error it is perhaps most natural to consider the vector distance in (R,G,B) between the illuminant and the estimate thereof, and thus we include some results using this error measurement. However, given the application dependent asymmetry between illuminant chromaticity and illuminant brightness, we find it more useful to look at the conjunction of a chromaticity error measure and the error in brightness. For the latter we use the difference in R+G+B between the illuminant and the estimate thereof.

Finally, it is of interest to consider the error in the final colour constancy result, which is the difference between the corrected image, and the exact target image taken under the canonical illuminant. These results are difficult to obtain with image data, because they require registered images with exactly the same geometry for each illuminant. This is only feasible if all illuminants are produced from a single source in conjunction with filters, which precludes the use of a general illuminant set like ours. The problems are diminished in the restricted case of chromaticity mappings, but even here, variations in geometry cause problems. Thus for this work we only supply mapping results for generated data. There are a number of ways to characterize the differences between the target data and the estimate thereof. We report the RMS error in (R,G,B) and the RMS error in (r,g). The reader may be wondering why we do not compute the difference between the result image and the target image in a perceptual space such as L\*a\*b. The main reason is that none of the algorithms attempt to minimize error in a perceptual space, and thus differences in performance in this space compared with performance in RGB space would be somewhat random.

# 4.3 Assumptions about the physical world: The Mondrian world and beyond

The bulk of colour constancy research has assumed that the world consists of perfectly diffuse reflecting surfaces. For example, in experiments investigating human colour constancy, the norm is to use matte cardboard. In the Retinex work of Land and McCann [44, 45], random collages of coloured cardboard were used, and referred to as Mondrians due to their likeness to Mondrian's paintings. Colour constancy has also been attempted by making use of specularities [18, 20, 69, 70, 72, 73]. In this paper we do not test any algorithms which specifically require specularities to be present. However, we will consider the performance of the algorithms with regard to the existence of specularities, as they are very common. In fact, it is difficult to take a comprehensive set of test images without including some specular reflection. Thus we feel that an appropriate context for testing available colour constancy algorithms is a Mondrian world with specularities from non-homogeneous materials such as paints and plastics. We further assume that the illumination chromaticity is spatially uniform. Readers interested in the case of slowly varying illumination are referred to [49, 50, 103, 104].

# 4.4 The World for Algorithm Calibration and Generating Synthetic Test Images

In addition to the assumptions discussed above, most colour constancy algorithms make assumptions about the diversity, and possibly the statistics, of the surfaces and the illuminants that may be encountered. Typically the surfaces and illuminants are supplied as collections of surface reflectances and illuminant energy spectra. The required data sets are then computed using an appropriate camera model. A collection of M surfaces and N illuminant spectra can be used to synthesize the camera responses to M\*N surfaceilluminant combinations. It is not normally practical to gather the required data directly, and doing so would have to be re-done every time the camera changed (but see [105] for work on colour constancy without calibration).

For surface reflectances we used a set of 1995 spectra compiled from several sources. These surfaces included the 24 Macbeth colour checker patches, 1269 Munsell chips, 120 Dupont paint chips, 170 natural objects, the 350 surfaces in Krinov data set [106], and 57 additional surfaces measured by ourselves. This set was used both for modeling the world for algorithm calibration (training) and testing the algorithms. We believe that the range of colour in this set encompasses most of the surfaces encountered in the image database, and thus it is also a serviceable training set for the image data experiments.

The choice of illuminant spectra must be made with more care. Several of the algorithms are sensitive to the statistics of the occurrence of the illuminants in the algorithm calibration (training) set. Thus we feel that it is best to have the training and testing sets both at least roughly uniformly distributed in (r,g) space. To obtain the appropriate illuminant sets, we first selected 11 sources to be used for the image data. These were selected to span the range of chromaticities of common natural and man made illuminants as best as possible, while bearing in mind the other considerations of stability over time, spectral nearness to common illuminants, and physical suitability. Our 11 sources include three fluorescent lights (Sylvania warm white, Sylvania cool white, and Philips Ultralume), four different 12 volt incandescent lights, and those four used in conjunction with a blue filter. The spectra of one of the incandescent lights (Sylvania 50MR16Q) is very similar to a regular incandescent light bulb. The other three are bulbs developed to provide spectra similar to daylight of three different colour temperatures (Solux 3500K, Solux 4100K, Solux 4700K). When used in conjunction with the blue filter (Roscolux 3202) these bulbs provide a reasonable coverage of the range of outdoor illumination. The chromaticities of all 11 illuminants are shown in Figure 4.1(a).

To create the illuminant set used for training, we divided (r,g) space into cells 0.02 units wide, and placed the 11 illuminants described above into the appropriate cells. We then added illumination spectra from a second set of 97, provided that their chromaticity bins were not yet occupied. This second set consisted of additional sources, including a number of illumination spectra measured in and around our university campus. The chromaticities of this additional illuminant set are shown in Figure 4.1(b). Finally, to obtain the desired density of coverage, we used random linear combinations of spectra from the two sets. This is justified because illumination is often the blending of light from two or more sources. In addition, to the extent that the diagonal model holds, these constructed illumination spectra will behave like physical sources with the same chromaticities as the constructed ones. Figure 4.1(c) shows the chromaticities of the training set obtained using this method. Finally, to produce the illuminant set for testing, we followed the same procedure, but filled the space 4 times more densely. The resultant chromaticities are shown in Figure 4.1(d).

In order to use the illuminant and reflectance spectra to generate data, we need a camera model. We calibrated our Sony DXC-930 CCD camera as described in Chapter 3. In general, camera calibration involves two steps. First we need a mapping from the actual camera output to ideal, linearized, sensor responses. This is discussed more fully below. The second step is to estimate the idealized sensor responses. Given these responses, the ideal camera response for channel k,  $\rho^{(k)}$ , is computed from a surface spectra  $S(\lambda)$  and illuminant spectra  $E(\lambda)$  and sensor function  $R^{(k)}$  by:

$$\rho^{(k)} = \int \mathbf{R}^{(k)}(\lambda) \mathbf{S}(\lambda) \mathbf{E}(\lambda) d\lambda$$
(4.1)

Of course, in practice, all the functions are replaced by vectors. In our case we use 101 samples from 380 nm. to 780 nm. in steps of 4 nm. which is the sampling provided by our PhotoResearch PR-650 spectrometer. The camera sensors are shown in Figure 4.2.





**Figure 4.1**: The chromaticity distributions of the various sets of illuminants used in this study. The 11 illuminants used for creating test images are shown in (a). In (b) we plot the chromaticities of an additional set composed of more sources, including a number illuminations measured in and around our university campus. The training set constructed from these sources is shown in (c). A similar set used for testing with the chromaticity space more densely populated is shown in (d).



Figure 4.2: The camera sensor response functions used in this study.

Finally, for parts of this study, we need to specify a canonical illuminant to be used as a standard, or target, illuminant. We chose the Sylvania 50MR16Q, as this is the illuminant for which the camera is best balanced. Specifically, under this illuminant, the camera response to perfect white is roughly the same across the three channels. The spectra of this illuminant is also smooth, unlike that of the three fluorescent illuminants, which we feel are less appropriate for standard illuminants.

# 4.5 Computational Colour Constancy Algorithms

We now turn our attention to the colour constancy algorithms themselves. We endeavored to include the most promising algorithms, as well as several simple, yet effective, commonly used algorithms. Several algorithms that were not implemented bear mentioning. First, we did not test the innovative Maloney-Wandell algorithm. Despite being an important contribution to the development of many ideas, this algorithm simply does not work well in the general context in which we test colour constancy. The reason for this is that, for a three sensor vision system, this algorithm requires that the surface reflectances of the world can be well approximated by two basis functions. This is not true in general. Several authors have noted that this problem leads to poor performance [58, 59].

A second important algorithm not tested is Brainard and Freeman's Bayesian method [59]. We feel that this algorithm has been superseded by the Colour by Correlation method which is investigated here in detail. We also exclude Buchsbaum's grey world variant [42], as well as that of Gershon et al [43]. Both of these methods use linear models in conjunction with assumptions similar to those of grey world approaches. The grey world algorithms that were implemented for this study use camera sensor responses instead, and therefore do not incur the error inherent in the linear models. We note that if linear models are to be used in conjunction with a specific camera, then the method for finding the basis functions provided by Marimont and Wandell [31] is likely a better a choice than the standard principal component analysis or SVD methods used in the four preceding algorithms. An additional part of Gershon et al's algorithm is the idea that grey world averaging should be done over segmented components of an image, rather than the image pixels. We note that evaluation of this idea is implicit in the pre-processing results presented below.

Finally, the extensive body of work on the Retinex theory of human vision has yielded several algorithms. The emphasis of Retinex theory is on the human vision system, and goes beyond simple illuminant estimation. Hence, computational colour constancy algorithms emerge from Retinex more as a process of analogy than through specification by the original researchers. Nonetheless, at least three algorithms for simple illuminant estimation in our context can be identified. We investigate the one closest to [1, 45, 54]. We do not test the method in [46, 47], which is analyzed in [52], nor the method in [48], as these methods are essentially grey world algorithms. We now consider the details of the algorithms for which we provide results.

#### 4.5.1 The Grey World method

The grey world method assumes that the average of the surface reflectances of a typical scene is some pre-specified value, which is referred to as "grey". The exact definition of "grey" requires some clarification. One possibility is simply true grey; specifically, a 50% uniform reflectance. This leads to the algorithm labeled GW in the results. A second choice is to use the average of the reflectance database. This method has an unnatural advantage in synthetic testing, and the result is guaranteed to be excellent if a large number of surfaces are used. In the case of real images, however, the actual average surface reflectance is not known, and thus this method is expected to fair relatively less well. We denote this algorithm by DB-GW in the results. We note that since these algorithms work on camera sensor responses, the actual assumption about scene averages is weaker than stated above. The algorithms simply assume that the scene average is identical to the camera response to the chosen "grey" under the scene illuminant. Then, under the diagonal assumption, the colour of white can be estimated from that average. In the case of GW, the average is simply multiplied by two. In the case of DB-GW, we scale the result by the ratio of the camera response to white under the canonical illuminant, to the camera response to grey, again under the canonical illuminant.

#### 4.5.2 The Retinex Algorithm

The Retinex based algorithm used for this study simply estimates the illuminant (R,G,B) by the maximum response in each channel [1, 45, 49]. This method is sensitive to the dynamic range of the vision system. If the dynamic

range is too small, then the (R,G,B) responses will be clipped at the largest possible value (typically 255), and this degrades the performance of this algorithm relative to most others [107]. We also note that in the Mondrian world, the estimate of the illuminant magnitude provided will be biased, as the maximum reflectance in the scene will always be less than that of a pure white. However, it is difficult to compensate for this bias, as it changes with the number of surfaces in a scene. Also, if specularities are present, then the maximum reflectance can easily be greater than that of pure white. On the positive side, we note that if specularities are present, and the vision system has sufficient dynamic range to prevent significant specularities from being clipped, then this method provides an excellent estimate of illuminant chromaticity. We note that in doing so, the algorithm is implicitly making use of pixel brightness information and thus could easily out-perform algorithms which use only chromaticity input.

#### 4.5.3 Gamut Mapping Methods

We present the results of a number of algorithms based on Forsyth's gamutmapping idea [32]. We will very briefly outline the general approach. First we form the set of all possible RGB due to surfaces in the world under the known, "canonical" illuminant. This set is convex and is represented by its convex hull. The set of all possible RGB under the unknown illuminant is similarly represented by its convex hull. Under the diagonal assumption of illumination change, these two hulls are a unique diagonal mapping (a simple 3D stretch) of each other.

Figure 4.3 illustrates the situation using triangles to represent the gamuts. In the full RGB version of the algorithm, the gamuts are actually three-dimensional polytopes. The upper thicker triangle (blue) represents the unknown gamut of the possible sensor responses under the unknown illuminant, and the lower thicker triangle (red) represents the known gamut of sensor responses under the canonical illuminant. We seek the mapping between the sets, but since the one set is not known, we estimate it by the observed sensor responses. These responses form a subset, the convex hull of which is illustrated by the thinner triangle (green). Because the observed set is



normally a proper subset, the mapping to the canonical is not unique, and Forsyth provides a method for effectively computing the set of possible diagonal maps. This set turns out to be a convex set in the space of mapping coefficients. (See §2.3.4 or [8, 32, 37, 40, 61] for more details on gamut-mapping algorithms).

Finlayson's Colour in Perspective algorithm adds two additional ideas [40]. First, Finlayson showed that the gamut-mapping method could be applied in two dimensions using the appropriate chromaticity space, specifically (R/B, G/B). Second, he observed that some of the possible diagonal maps do not correspond to common illuminants. By making the assumption that the scene illuminant is inside a set of common, natural and man made illuminants, Finlayson was further able to constrain the set of diagonal maps. This constraint is non-convex in the space of diagonal maps. In [40] the combined solution set was considered to be the intersection of the convex constraint set due to the original surface constraints, and the non-convex illuminant constraint set. In [8] the illuminant constraint set was approximated by its convex hull. In addition, in that work the illumination constraint was added to the full (R,G,B) case.

Once the set of possible maps has been computed, an important second stage of the algorithm is to choose a solution from the feasible set. Several different methods for doing this have been proposed. The original method chose the solution which maximized the volume of the mapped set [32]. The Colour in Perspective method uses the same heuristic in chromaticity space. However, this solution method is quite biased, and in [8] the average of the constraint set was investigated, both in the chromaticity based algorithm and the (R,G,B) algorithm. This method for choosing the solution is still biased in the chromaticity case, and in [61] the averaging was done in three dimensions. Specifically, the constraints on the mappings in perspective space correspond to cones in the space of mappings between (R,G,B) gamuts. In order to average over the non-convex illumination constraint, Monte Carlo integration was used. In this work, we instead approximate this average using numerical integration, both in the chromaticity and in the (R,G,B) case.

To summarize, we investigate three methods of forming the solution set. These are the original method, designated by CRULE (for "coefficientrule", the name of the original algorithm), the Colour in Perspective method, designated by CIP, and the illumination constraint set applied to CRULE designated by ECRULE (for "extended-CRULE"). We do not deal with the

Label	Reference	Notes
CRULE-MV	[32]	The original algorithm
CRULE-AVE	[8]	Solution is average of constraint set
ECRULE-MV	[8]	Finlayson's illuminant constraint added to the original algorithm
ECRULE-AVE	[8]	Solution is average of constraint set; the illuminant constraint set contribution is approximated by its convex hull.
ECRULE-ICA	This work	Solution is average of constraint set, computed by numerical integration
CIP-MV	[40]	Colour in perspective
CIP-AVE	[8]	Solution is average of constraint set; the illuminant constraint set contribution is approximated by convex hull.
CIP-ICA	[61]	Solution is average of constraint set computed in three- dimensional map space using numerical integration (originally Monte Carlo).

Table 4.1: A summary of the gamut-mapping algorithms investigated in this paper.

chromaticity case without the illumination constraint. These solution sets are paired with methods of selecting a solution from them. We use MV to denote the original maximum volume heuristic, AVE to specify that the constraint set is averaged, using a convex approximation to the illumination constraint if necessary, and ICA to specify that the constraint set is numerically integrated to deal with the fact that it is non-convex ("illumination constrained average"). Table 4.1 lists the eight algorithms investigated.

#### 4.5.4 Colour by Correlation

Recently, Finlayson et el. introduced Colour by Correlation [62] as an improvement on the Colour in Perspective method. The basic idea of Colour by Correlation is to pre-compute a correlation matrix which describes how compatible proposed illuminants are with the occurrence of image chromaticities. Each row in the matrix corresponds to a different training illuminant. The matrix columns correspond to possible chromaticity ranges resulting from a discretization of (r,g) space, ordered in any convenient manner. Two versions of Colour by Correlation are described in [62]. In the first version, the elements of the correlation matrix corresponding to a given illuminant are computed as follows: First, the (r,g) chromaticities of the reflectances in the training set under that illuminant are computed using the camera sensors. Then the convex hull of these chromaticities is found, and all chromaticity bins within the hull are identified as being compatible with the given illuminant. Finally, all entries in the row for the given illuminant corresponding to compatible chromaticities are set to one, and all other elements in that row are set to zero.

To estimate the illuminant chromaticity, the correlation matrix is multiplied by a vector whose elements correspond to the ordering of (r,g) used in the correlation matrix. The elements of this vector are set to one if the corresponding chromaticity occurred in the image, and zero otherwise. The i'th element of the resulting vector is then the number of chromaticities which are consistent with the illuminant. Under ideal circumstances, all chromaticities in the image will be consistent with the actual illuminant, and that illuminant will therefore have maximal correlation. As is the case with gamut-mapping methods, it is possible to have more than one plausible illuminant, and in our implementation we use the average of all candidates close to the maximum. We label this algorithm "C-by-C-01".

In the second version of Colour by Correlation, the correlation matrix is set up to compute the probability that the observed chromaticities are due to each of the training illuminants. The best illuminant can then be chosen using a maximum likelihood estimate, or using some other estimate as discussed below. To compute the correlation matrix, the set of (r,g) for each illuminant using our database of surface reflectances is again found. The frequency of occurrence of each discrete (r,g) is then recorded. If additional information about the frequency of occurrence of these reflectances is available, then the frequency counts are weighted accordingly. However, since such a distribution is not readily available for the real world, in our implementation we simply use uniform statistics. The same applies for the illuminant data set. The counts are proportional to the probability that a given (r,g) would be observed, given the specific illuminant. The logarithms of these probabilities for a given illuminant are stored in a corresponding row of the correlation matrix. The application of the correlation matrix, which is done exactly as described above, now computes the logarithm of the posterior distribution.

This computation of the posterior distribution is a simple application of Bayes's rule. Specifically, the probability that the scene illuminant is I, given a collection of observed chromaticities C, is given by:

$$P(I/C) = \frac{P(C/I)P(I)}{P(C)}$$

$$(4.2)$$

Since we are assuming uniform priors for I, and since P(C) is a normalization which is not of interest, this reduces to:

$$P(I/C) \propto P(C/I) \tag{4.3}$$

Assuming that the observed chromaticities are independent, P(C | I) itself is the product of the probabilities of observing the individual chromaticities *c*, given the illuminant I:

$$P(C/I) = \prod_{c \in C} P(c/I)$$
(4.4)

Taking logarithms gives:

$$log(P(C/I)) = \sum_{c \in C} log(P(c/I))$$
(4.5)

This final quantity is exactly what is computed by the application of the correlation matrix to the vector of chromaticity occurrences. Specifically, the i'th element of the resulting vector is the logarithm of the posterior probability for the i'th illuminant.

The method described so far will work fine on synthetic data, provided that the test illuminant is among the training illuminants. However, once we apply the method to the real world, there are several potential problems. First, due to noise, and other sources of mismatches between the model and the real world, an observed set of chromaticities can yield zero probability for all illuminants, even if the illuminant, or a similar one, is in the training set. Second, the illumination may be a combination of two illuminants, such as an arbitrary mix of direct sunlight and blue sky, and ideally we would like the method to give an intermediate answer. We deal with these problems as follows. First, as described above, we ensure that our illuminant set covers (r,g) space, so that there is always a possible illuminant not too far from the actual. Second, as we build the correlation matrices, we smooth the frequency distribution of observed (r,g) with a Gaussian filter. This ensures that there are no holes in the distribution, and compensates for noise.

The final step is to choose an answer, given the posterior probability distribution, an example of which is shown in Figure 4.4. The original work [62] mentions three choices: The maximum likelihood, mean likelihood, or local area mean, introduced in [59]. That work discusses these methods in detail with respect to a related colour constancy algorithm, where they are referred to as the MAP, MMSE, and MLM estimators, respectively. We will adopt this notation here. The MAP estimate is simply the illuminant which has the maximum posterior probability. To compute the MMSE estimate of the chromaticity estimate we take the average (r,g) weighted by the posterior distribution. The MLM estimator is computed by convolving the posterior distribution with a Gaussian mask, and then finding the maximum. For our

purposes, we would like to choose the particular Gaussian mask which minimizes the error of some specific task. Unfortunately, the bulk of our results are not of much help here, as they are based on RMS error, and thus we already know that the MMSE method will work better. Brainard and Freeman argue that the MLM estimate should be considered in favour of the MMSE estimate because the latter penalizes large errors too much—they would prefer that once errors are beyond a certain size, they are considered equally bad. Without a specific task, it is difficult to quantify such an error measure, but we can take a small step in that direction by considering the average absolute error, as opposed to the RMS error. If we do this, then we can find a mask size where the MLM estimate is slightly better than the MMSE estimate for Colour by Correlation in simulation. We choose this mask size for the MLM estimates reported. However, we remind the reader that demonstrating the virtues of the MLM method would require error measures which are different than the ones used for this study.



**Figure 4.4**: An example posterior distribution, showing the probabilities that the illuminant in the training set with chromaticity (r,g) explains the observed data produced from a randomly selected illuminant and 8 randomly selected surface reflectances.

#### 4.5.5 Neural Net Methods

We also provide the results from a neural network trained to estimate the colour of the illuminant [63-67]. The neural net is a multi-layer Perceptron with two hidden layers. As is common, the general structure is pyramidal. The input layer consists of 2500 nodes, the first hidden layer has 400 nodes, the second hidden layer has 30 nodes, and the output layer has 2 nodes. We divide (r,g) chromaticity space into discrete bins, with each input neuron corresponding to one of the discrete bins. The input to each neuron is a binary value representing the presence or absence of a scene chromaticity falling in the corresponding (r,g) bin. Thus we form a (r,g) histogram of the image, and then binarize that histogram.

The output signal from the two output neurons are real valued, and correspond to an estimate of the chromaticity of the scene illuminant. The network is trained to compute this estimate by being presented with many synthesized images, generated from the training sets described above, together with the chromaticity of the illuminant used to generate each image. The training of the neural net occurs by re-adjustment of neuron weights using back-propagation without momentum [108], based on the discrepancy between predicted and actual scene illuminant chromaticity. Since we know in advance that some of the input (r,g) space will not be used, we use an adaptive network whereby links to neurons dormant for an entire training epoch are deleted and replaced by ones connected to randomly selected input nodes. This modification substantially reduces training time.

The training of the neural network can be structured to match more closely the real world. For example, we added a small amount of normally distributed, random noise to the synthetically generated data used for training. In addition, we trained a second network with data that modeled the occurrence of specularities as described in [64]. We label results obtained with the standard neural net with NEURAL-NET, and those obtained from the one trained with specularities as SP-NEURAL-NET.

### 4.6 Computational Colour Constancy Meets the Real World

Most work in computational colour constancy has been on generated data. Algorithms are sometimes tested on real data, but comprehensive tests are rare. Specifically, there is a dearth of work towards the goal of making algorithms work well for real images. In this section we will discuss some of the issues that need to be addressed.

The first issue is camera calibration. Given the dependence of the field on using illuminant and reflectance spectra, we must be able to predict camera responses to colour signals. This goes beyond knowing the sensor response curves. Specifically, before such curves can be used, we must map the camera responses into a space where those responses are proportional to input energy. In our case, our camera has a small non-linearity, and has a substantial dark current offset. Properly correcting for these characteristics yields better colour constancy performance. Our method for doing so is described in Chapter 3; camera calibration is also discussed in [7, 9, 10, 97, 98, 109]. Once the camera was calibrated, we did not change any camera parameters. Specifically, focus, aperture, zoom, balance, channel gains, and gamma setting were all held fixed.

We also perform spatial corrections to the image data. We measured the fixed pattern noise by averaging 5000 frames taken with the lens cap on, and subtracted this spatial variation from all our images. This is a small correction, which is only relevant due to the frame averaging discussed below. A bigger problem with our camera is spatial variation due to the optics, which is in addition to any variation due to non-uniformity among the CCD elements. Healey describes a method for measuring spatial variation [7], but performing this calibration is quite difficult, and requires a more spatially uniform illumination than we have access to. However, differences in pixel magnitude simply mimics shading, and for our purposes, we are most concerned with chromaticity variation. Since we can create illuminants which are quite spatially uniform in chromaticity, we were able to correct for this variation. Using such an illuminant, the chromaticity variation of a spatially uniform surface is due to variations in the imaging process. We used the ratio of the (R/B, G/B) chromaticity at the center of such a calibration image to that of each pixel as the spatial correction factors which modify the R and G values, but leave the B values unchanged. Before this correction, the images had a noticeable reddish halo towards the edges; our correction removed the bulk of this defect.

Another property of non-ideal cameras is limited dynamic range. This does not present a problem in the case of the Mondrian world, but when we move to images which contain specularities, we find that they are often clipped at the maximum possible response (typically 255). While we are interested in studying colour constancy under these conditions, we are also interested in the possibilities afforded by higher dynamic range cameras, such as the one described in [102]. In order to investigate these issues more fully we extend the dynamic range of our image data by averaging multiple frames to reduce the noise in the small pixel values. The effect of any aperture setting, and consequent pixel clipping, could then be simulated by artificially scaling and clipping the images. We note that rather than using the clipped pixels, we discard them, as using them degrades the performance of most algorithms. We also discard pixels with overly small pixel values, as some algorithms, notably the Colour in Perspective ones, are sensitive to noise in such pixels.

The second issue we face when we take colour constancy to the real world is image pre-processing. All the algorithms discussed are cast in terms of one input item per identifiable image surface. This is a little different than a real image, which is a collection of multiple samples per surface, and includes samples which straddle surface boundaries. Many algorithms are indifferent to the statistics of the sampling; only the presence or absence of a colour makes a difference to the outcome. Thus it is common to simply use the image pixels themselves as input. However, initial experiments indicated that it is better to first average the images, with a block size of 5 by 5 being roughly optimal for our camera. Nonetheless, blindly averaging the image in this way reduces the information available to the algorithms, and such a step should be less important when frame averaging is used. Interestingly, we found that this is not case. We believe that the reason for this anomaly is as follows: Using each pixel as a datum makes the implicit assumption that the (R,G,B) of pixels which straddle two surfaces is a convex combination of the (R,G,B)'s of pixels on one side or the other. Careful examination of images reveals that, for our camera, this assumption does not hold, possibly due to

chromatic aberration or misregistration of the CCD elements (our camera has 3 CCD arrays—the incoming signal goes through beam splitters and filters on its way to the CCD arrays). Specifically, we found that (R,G,B)'s of a non-negligible number of boundary pixels were not the convex combination of surrounding pixels, and thus should be considered erroneous data. Furthermore, cameras with mosaic'ed sensors, which are more common than 3 CCD cameras like ours, could also be susceptible to similar anomalies.

This problem, together with the above observation that the algorithms are expressed in terms of surfaces, leads to the consideration of image segmentation as a form of pre-processing, and we investigated this idea in detail. We were able to find segmentation parameters which improved most algorithms, although the effect was quite algorithm dependent, and in fact, more volatile than we expected. We present results using these somewhat arbitrarily chosen, general purpose parameters. However, since it is reasonable for a proponent of a given algorithm to optimize the preprocessing for that algorithm, we also present results where the optimal among all pre-processing methods was chosen on an algorithm-by-algorithm basis.

To segment the images we used region-growing, subject to two constraints. First, we ensured that all chromaticities in a region were within a certain absolute tolerance of each other. Second, we ensured that the pixel brightness, quantified by R+G+B, of all pixels in a region were within a certain relative tolerance of each other. In addition, we insisted that the region was larger than a certain number of pixels. Thus there were 3 segmentation parameters. We used 4 different values of the first (0.0025, 0.005, 0.01, 0.02), 4 for the second (10%, 20%, 30%, and 40%), and 3 for the third (5, 10, 20). The collection of the (R,G,B) averages of each region was then used as input to the algorithms. We also pre-processed by averaging blocks with lengths of sides (1, 2, 3, 5, and 7), and also by then additionally grouping the block averaged result into bins in RGB space (100 divisions per channel), and then using the average of the RGB in each of the bins, thereby removing duplicate colours from the input. A final possible pre-processing step that we have experimented with is to use the (R,G,B) vertices of the convex hull of the data instead of the data itself. This is motivated by the knowledge that for the gamut-mapping algorithms, the hull boundary points are the ones that

matter. Of course, using the convex hulls has no effect on gamut-mapping algorithms, as they already use the convex hull, but preliminary results with the neural net and Colour by Correlation are promising. Therefore we also tested using this additional step in the case of block averaging (5 block sizes) and two selected segmentation methods. Thus a total of 65 pre-processing methods were tested.

### 4.7 Experiments with Generated Data

Although the ultimate goal is to develop and study colour constancy algorithms for image data, certain aspects of the algorithms are best studied using the controlled environment of generated data. However, comparisons based on synthetic data have to be made with care, as some algorithms are more sensitive than others to the benefits of a testing situation that has similar statistical properties to that of the training situation. For example, the DB-GREY-WORLD algorithm has access to the exact average of the reflectance data set, and can benefit accordingly, but this information is not known the case of real images.

We present the results for four experiments. In the first, we look at the distribution of (r,g) chromaticity errors for a variety of algorithms. Next we look at the chromaticity performance of the algorithms as the number of surfaces in the synthetic scenes increases, and we also look at their performance using 8 surfaces with variety of error measures. For our third experiment we added simulated specularities to the generated data, and in the forth experiment we studied the effect of clipping the overly bright specular pixels.

We begin by looking at the distribution of (r,g) chromaticity errors. Ideally, the errors will be approximately normally distributed around (0,0). If the mean is much different than (0,0), the algorithm is biased, and is not performing optimally—it could be improved simply by adding the appropriate bias to the answer. Figure 4.5 shows the distribution of errors in r=R/(R+G+B) for 12 algorithms. The plots are the histograms of the errors from 5000 runs, using 8 surfaces. As expected, the CIP-MV algorithm is biased. Furthermore, using the average of the perspective diagonal maps, as is done with the CIP-HA algorithm, offers only a modest improvement in this regard. These results demonstrate that choosing a solution in (R/B, G/B) space requires care. The CIP-ICA method, which averages the results after back-projecting them to three-dimensional space, has no obvious bias, nor does the C-by-C-01 algorithm, which can be viewed as an implementation of the Colour in Perspective method. A second observation is that all the Colour in Perspective algorithms have quite wide and rounded error distributions, in contrast to the other algorithms whose error distributions are more suggestive of the normal distribution.

We also observe that the CRULE-HA algorithm has a small bias, which is substantially decreased when Finlayson's illumination constraint is added (ECRULE-ICA). We interpret this result as indicating that the solutions excluded by the illumination constraint are themselves biased. Finally we note that the three algorithms which are sensitive to the testing statistics (DB-GREY-WORLD, C-BY-C-MAP, and NEURAL-NET), all have near normal error distributions when tested with statistics similar to those used for training.

We now consider the performance of the algorithms with respect to the error measures introduced above. We measured the performance of the algorithms for synthetic scenes with 4, 8, 16, 32, 65, 128, 256, 512, and 1024 surfaces. For each number of surfaces, we generated 1000 scenes with the surfaces randomly selected from the reflectance database and a randomly selected illuminant from the test illuminant database. For each algorithm and number of scenes we computed the RMS of the 1000 results. We choose RMS over the average because, on the assumption of roughly normally distributed errors with mean zero, as we observed for most algorithms in the preceding experiment, the RMS gives us an estimate of the standard deviation of the algorithm estimates around the target. This is preferable to using the average of the magnitude of the errors, as those values are not normally distributed. Finally, given normal statistics, we can estimate the relative error in the RMS estimate by  $1/\sqrt{2N}$  [110, p. 269] For N=1000, this is roughly 2%.



**Figure 4.5**: Distribution of error in r=R/(R+G+B) for selected algorithms.



**Figure 4.5 (continued)**: Distribution of error in r=R/(R+G+B) for selected algorithms.



**Figure 4.5 (continued)**: Distribution of error in r=R/(R+G+B) for selected algorithms.

#### A COMPARISON OF COLOUR CONSTANCY ALGORITHMS

In Figures 4.6 and 4.7 we plot the error in (r,g) for selected algorithms as a function of the number of surfaces. As the number of surfaces in the synthetically generated scenes increases, there is more information available for the algorithms, and performance generally increases. In Table 4.2 we provide the results of all algorithms using several error measures for the exemplary case of 8 surfaces. We note that the number of surfaces cannot easily be translated into the number of colours in a real image, as the reflectance data set includes a much higher population of extreme colours than the real world, as it is designed to provide the range of possible colour.

We include the results of two minimal colour constancy methods for comparison purposes. The first is to do nothing, which implicitly assumes that the vision system is already properly calibrated for the actual illuminant, and in our context, is equivalent to guessing that the actual illuminant is in fact the canonical (target) illuminant. The method is denoted by NOTHING. The second method is similar, but instead, the actual illuminant is assumed to be the average of the normalized illuminants in our database. The method is denoted by AVE-ILLUM. Both NOTHING and AVE-ILLUM are independent of the scene, and thus their error is constant with respect to the number of surfaces. Since the test illuminants are distributed throughout the data set, and since the canonical illuminant is towards the periphery of the set (it is redder than average), AVE-ILLUM is a more effective minimal algorithm than NOTHING.



**Figure 4.6**: Error in (r,g) chromaticity as a function of the number of surfaces for selected algorithms. The results shown are the RMS value of the error for 1000 synthetically generated scenes. With the scale used above, the C-by-C-01 (not shown) is similar to CIP-ICA, and C-by-C-MAP and C-by-C-MMSE are similar to the neural net. These algorithms and other good performers are plotted on a larger scale in Figure 4.7.



**Figure 4.7**: Error in (r,g) chromaticity as a function of the number of surfaces for some of the better performing algorithms. The results of NOTHING, AVE-ILLUM, CIP-MV, and CIP-HA are largely off the scale, and thus are omitted. Numeric results for the case of 8 surfaces (corresponding to x=3 in this plot) for all algorithms are available in Table 4.2.

Algorithm	Illuminant Estimate Angular Error	Illuminant Estimate rg Error	Illuminant Estimate RGB Error	Illuminant Estimate R+G+B Error	Scene Mapping RMS rg error	Scene Mapping RMS RGB error	CPU Time (seconds)
NOTHING	16.45	0.114	*	*	0.113	*	*
AVE-ILLUM	11.79	0.086	*	*	0.089	*	*
GW	8.00	0.058	190.2	310.1	0.062	137.4	*
DB-GW	6.51	0.048	95.9	144.6	0.054	33.1	*
RETINEX	9.03	0.067	165.4	267.0	0.072	105.6	*
CIP-MV	26.27	0.200	*	*	0.240	*	0.030
CIP-AVE	18.12	0.130	*	*	0.141	*	0.030
CIP-ICA	10.51	0.077	*	*	0.081	*	0.074
NEURAL-NET	5.23	0.038	*	*	0.045	*	0.015
SP-NEURAL-NET	5.47	0.040	*	*	0.046	*	0.014
C-by-C-01	10.79	0.078	*	*	0.082	*	0.026
C-by-C-MAP	5.63	0.042	*	*	0.048	*	0.028
C-by-C-MLM	5.25	0.039	*	*	0.045	*	4.275
C-by-C-MMSE	4.66	0.034	*	*	0.041	*	0.036
CRULE-MV	6.75	0.052	111.3	178.3	0.058	54.3	0.071
CRULE-AVE	8.39	0.061	207.5	302.7	0.066	44.8	0.057
ECRULE-MV	6.04	0.046	107.3	172.2	0.052	51.1	0.186
ECRULE-AVE	7.22	0.051	147.4	223.3	0.054	37.3	0.151
ECRULE-ICA	7.15	0.051	143.7	217.6	0.053	36.9	1.835

**Table 4.2**: Algorithm performance with respect to a number of error measures. The values listed are the RMS over 1000 synthetically generated scenes, each having 8 surfaces randomly selected from the reflectance data set, as viewed under an illuminant randomly selected from the test illuminant data set. The uncertainty in these numbers is roughly 2%. An asterisk is used for missing or non-applicable values. For example, the algorithms which only estimate chromaticity have no entry for error measures which involve illuminant brightness. Brightness oriented measures are also not applied NOTHING and AVE-ILLUM because the illuminant data set is normalized, and thus, in our implementation, these algorithms have access to the brightness. The rightmost column is not an error measure, but simply the CPU time required to calculate the results. The numbers reported are necessarily tied to our implementation and our computers, and hence they can only give the reader a general idea as to the resources required by each algorithm.

#### A COMPARISON OF COLOUR CONSTANCY ALGORITHMS

A similar consideration explains the poor performance of the CIP-MV algorithm. As noted above, this algorithm is biased. Specifically, in the (R/B, G/B) chromaticity space, the maximum volume constraint chooses essentially the bluest illuminant consistent with the observed chromaticities. With our comprehensive illuminant data set, many surfaces are required before the algorithm CIP-MV performs better than the two minimal algorithms. Similarly, many surfaces are also required to obtain a good result using the only slightly less biased CIP-HA algorithm. Finally the CIP-ICA algorithm was consistently better than both NOTHING and AVE-ILLUM, and performed much better than the other two Colour in Perspective methods. Also, as expected, the C-by-C-01 algorithm behaved similarly to the CIP-ICA method.

A concise description of the results is difficult, as the rank order of algorithm performance is a function of the number of surfaces. For example, the CIP-ICA and C-by-C-01 methods performed better than RETINEX for a small number of surfaces, but the error with RETINEX dropped rapidly as the number of surfaces increased, becoming lower than that for the Colour in Perspective methods around 8 surfaces. The methods based on threedimensional gamut-mapping generally did better than RETINEX, although the variants which average their constraint set to obtain the final solution (CRULE-AVE, ECRULE-AVE, and ECRULE-ICA) were exceeded by RETINEX at around 16 surfaces. The maximum volume constraint was better than averaging for estimating illuminant chromaticity, except in the important case of a small number of surfaces, where the ECRULE-ICA method was the best gamut-mapping algorithm.

In general, the methods which take advantage of the statistical nature of the input performed the best. As expected, the C-by-C-MMSE algorithm performed better than the C-by-C-MAP algorithm, as it is known to be optimal for RMS error. The neural net methods were between C-by-C-MAP and C-by-C-MMSE, as was C-by-C-MLM.
### A COMPARISON OF COLOUR CONSTANCY ALGORITHMS

We note that for a large number of surfaces (much better input than normally available in practice) the errors of most algorithms did not tend to zero. In the case of the Colour by Correlation methods, this is simply due to the discretization of the input. We chose to break the (r,g) space into 50 units by 50 units, which corresponds to bins which are 0.02 units square. Thus we cannot expect the algorithm to do much better than an error of 0.01 in each of r and g, which corresponds to a vector distance error of  $0.01^*\sqrt{2}$ , which is consistent with our findings. We note that we chose the resolution to be consistent with that of the neural network, and we assume that the limiting error for the neural net is also largely due to the same problem. This error could be reduced by using a more accurate discretization, but doing so is not particularly important, as we rarely have this kind of input outside of simulation.

The gamut-mapping methods also did not converge to zero error. Here the problem is the failure of the diagonal model. The error for RETINEX, on the other hand, did get close to zero. It does not go exactly to zero because we did not include a perfect reflectance in the data set. The GW algorithm converges to a set error, which represents the difference between the actual database average, and a perfect grey. Finally, when we used the database average for the grey with the DB-GW algorithm, then the error converged to zero as expected.

We turn briefly to brightness measures. The DB-GW algorithm is the overall best algorithm in the case of 8 surfaces, but since it uses information which is not normally available with real data, we are more interested in the other algorithms. We note that among the more readily realizable algorithms, the best choice is invariably one of the gamut-mapping algorithm. However, we make the interesting observation that the preferred gamut mapping algorithm depends on the error measure. If the goal is to estimate the illuminant brightness then we prefer the maximum volume heuristic. On the other hand, if we wish to accurately map the image to a similar one of the same scene under the canonical illuminant, then we prefer choosing the solution from the constraint set by averaging. This is consistent with the easily demonstrated claim that the average minimizes the expected RMS mapping error [8], and in fact this was the reason for introducing this solution selection method. Given that the two quantities being estimated are related through the diagonal model, the observation that a different estimation method works better for illuminant (R,G,B) estimation, needs explanation. The reason is that the space of diagonal maps is approximately proportional to the element-wise *inverse* of the illuminant (R,G,B). This is most easily visualized by noting that as we approach the origin in the mapping space, we are dealing with an increasingly bright illuminant. Thus averaging diagonal maps is not equivalent to averaging illuminants.

In our third experiment, we simulated scenes with a significant number of specularities. We randomly selected 25% of the surfaces in the data set to be specular. To each of these reflectance spectra, we added a random factor times a perfect reflectance. The random factors were uniformly distributed between 0 and 2. Thus the surface reflectances could now be up to three times as bright as in the previous experiment.

We plot the change in the chromaticity results from the non-specular case as a function of the number of surfaces (Figure 4.8), and provide a variety of absolute results for the case of 8 surfaces (Table 4.3). Here we see that when specularities are present, most algorithms estimate chromaticity more accurately, even though they were not designed to take advantage of specularities. For example, with specular reflection, the maximum value in each channel is more likely to be close to the colour of white under the scene illuminant, and thus, in this test, RETINEX does especially well. Chromaticity based algorithms cannot make similar use of the brightness information, but as argued in [40], these algorithms are at least robust with respect to specularities. This is because specularities simply move the observed chromaticities towards white, and the modified ones are valid chromaticities for some physically realizable surface. However, we expect some degradation given our experimental paradigm, because colours which are more saturated tend to lead to stronger constraints on the illuminant. This is consistent with the slight degradation of the Colour in Perspective method clearly shown in Figure 4.8.

We also expected the grey world algorithms to estimate the illuminant chromaticity better when specularities are present because the extra part added to the sum used in the average has the same chromaticity as a perfect grey. We note that it does not have the same chromaticity as the grey used by the DB-GW algorithm, and this explains why that algorithm degrades dramatically in the case of a large number of surfaces. This is more a statement about the unnaturally good performance of the DB-GW algorithm in the standard case, rather than a problem with using it with specularities. Amongst the three-dimensional gamut-mapping algorithms, the largest performance increase due to specularities was when the maximum volume heuristic was used to choose the solution. This was expected, as this heuristic tends to choose the map which takes a bright (and thus in this experiment, specular) pixel to the (R,G,B) for white under the canonical, which yields a good chromaticity result.

Turning to the results provided in Table 4.3, we note that the neural net trained on specular input does slightly less well than the standard net. This is likely due to normal variability, but we also note that the statistics of the specular training set was different than the testing statistics described above. When the statistics match, then the specular training has been shown to help substantially [64].



**Figure 4.8**: The change in (r,g) error due to the addition of simulated specularities. The experiment used to produce Figure 4.7 was rerun with added specularities. This plot is the difference of the errors, as a function of the number of surfaces. The absolute errors for the case of 8 surfaces (corresponding to x=3 in this plot) are available in Table 4.3.

Algorithm	Illuminant Estimate Angular Error	Illuminant Estimate rg Error	Illuminant Estimate RGB Error	Illuminant Estimate R+G+B Error	Scene Mapping RMS rg error	Scene Mapping RMS RGB error
NOTHING	16.4	0.114	*	*	0.114	*
AVE-ILLUM	11.8	0.086	*	*	0.090	*
GW	4.8	0.035	132.2	216.5	0.041	104.1
DB-GW	4.5	0.033	448.0	731.4	0.038	170.1
RETINEX	4.5	0.033	256.7	427.8	0.039	139.7
CIP-MV	26.5	0.202	*	*	0.245	*
CIP-AVE	18.3	0.131	*	*	0.143	*
CIP-ICA	10.6	0.078	*	*	0.082	*
NEURAL-NET	4.1	0.031	*	*	0.037	*
SP-NEURAL-NET	4.4	0.032	*	*	0.038	*
C-by-C-01	10.9	0.079	*	*	0.083	*
C-by-C-MAP	4.2	0.031	*	*	0.037	*
C-by-C-MLM	3.7	0.027	*	*	0.034	*
C-by-C-MMSE	3.5	0.026	*	*	0.032	*
CRULE-MV	3.8	0.029	349.3	577.1	0.035	156.1
CRULE-AVE	7.5	0.053	1027.0	1631.9	0.054	219.1
ECRULE-MV	3.6	0.027	350.3	578.7	0.033	156.1
ECRULE-AVE	6.6	0.046	824.6	1352.5	0.049	210.1
ECRULE-ICA	6.5	0.046	810.8	1330.9	0.048	209.2

**Table 4.3**: Algorithm performance with respect to a number of error measures for generated scenes with specularities (uncertainty is roughly 2%). A synthetic specularity was added to 25% of the surfaces used. This entailed adding a random factor, uniformly distributed between 1 and 2, of a perfect reflectance to the surface reflectances chosen to be made specular. In all other regards, the experiment which produced the numbers here is analogous to that for Table 4.2. We do not report the CPU time in this table, as it is similar to that for Table 4.2.

### A COMPARISON OF COLOUR CONSTANCY ALGORITHMS

The results in Table 4.3 also show that specular reflection has a generally negative impact on the estimation of the overall brightness of the illuminant. This is expected, as none of the algorithms model the effect of specular reflection on pixel brightness. The decrease in error in the GW case is simply due to a brightness bias in that algorithm in the non-specular case, and thus it essentially works better by accident.

As discussed above, when strong specularities are present in real images, they are often clipped. This motivates our fourth experiment where we investigate the performance of the algorithms in the specular case with simulated clipping of the brightest pixels. Three levels of artificial clipping were used. For each level, all pixels with R, G, or B over that level were discarded. The three clipping levels used were 250, 300, and 400. In Figure 4.9 we plot the change in algorithm performance as a consequence of imposing a clipping level of 300. Naturally, clipping degraded most algorithms, but the algorithms differ with respect to the degree of degradation. On the one extreme, clipping under these circumstances has little effect on the Colour in Perspective algorithms. This is understandable because the level of clipping used was such that only specular pixels are clipped, and these pixels, having chromaticities near white, are not of much use to those algorithms.

At the other extreme, clipping essentially disables the ability of RETINEX to use specularities to improve the illumination chromaticity estimate. As the number of surfaces increases, the effect becomes less damaging because it becomes more likely that there is at least one specular pixel which is just below the clipping level, which helps the algorithm. However, as the number of surfaces becomes very large, RETINEX degrades quite rapidly. The reason is that with a large number of surfaces and many random specularities, the non-discarded pixels will tend to have a maximum in each channel that approaches the clipping level. For example, in the case of the clipping level of 300, then the illuminant estimate will converge to (300, 300, 300). This is the same answer as the NOTHING algorithm! It is important to note that even though we do not normally have this diversity of colours in real images, a similar effect does in fact occur in images with fewer colours but a wide variety of strengths of specularities.





	Illuminant Estimate Angular Error			Illuminant Estimate rg Error		
Algorithm	Clipped at 400	Clipped at 300	Clipped at 250	Clipped at 400	Clipped at 300	Clipped at 250
NOTHING	16.4	16.4	16.4	0.1138	0.1138	0.1138
AVE-ILLUM	11.8	11.8	11.8	0.0858	0.0858	0.0858
GW	6.2	7.3	7.8	0.0450	0.0537	0.0578
DB-GW	5.4	6.2	6.6	0.0396	0.0463	0.0498
RETINEX	6.3	7.9	8.7	0.0458	0.0590	0.0658
CIP-MV	26.5	26.5	26.5	0.2021	0.2021	0.2021
CIP-AVE	18.3	18.3	18.3	0.1311	0.1311	0.1311
CIP-ICA	10.6	10.6	10.6	0.0775	0.0775	0.0775
NEURAL-NET	4.6	5.1	5.4	0.0342	0.0376	0.0396
SP-NEURAL-NET	4.9	5.4	5.7	0.0360	0.0396	0.0416
C-by-C-01	10.9	10.9	10.9	0.0789	0.0789	0.0789
C-by-C-MAP	4.7	5.5	5.8	0.0347	0.0402	0.0424
C-by-C-MLM	4.4	5.0	5.2	0.0320	0.0369	0.0385
C-by-C-MMSE	4.0	4.5	4.7	0.0295	0.0329	0.0347
CRULE-MV	5.2	6.3	6.8	0.0387	0.0475	0.0521
CRULE-AVE	8.0	8.3	8.6	0.0565	0.0602	0.0629
ECRULE-MV	4.8	5.6	6.1	0.0354	0.0420	0.0459
ECRULE-AVE	6.9	7.1	7.3	0.0486	0.0503	0.0519
ECRULE-ICA	6.8	7.0	7.2	0.0478	0.0496	0.0511

**Table 4.4**: Algorithm chromaticity performance with respect to two error measures for generated scenes with specularities with three levels of simulated clipping (uncertainty is roughly 2%). All generated (R,G,B) values with any of R, G, or B over the clipping level were discarded. In all other regards, the experiment which produced the number here is analogous to that for Table 4.2.

## 4.8 Experiments with Image Data

To investigate the performance of the various algorithms on more realistic data, we took images of 30 scenes under the 11 illuminants specified above, for a total of 330 images. Some of the images had to be culled due to problems, leaving 321 for our experiments. The images of all the scenes under the canonical illuminant are shown in Figure 4.10, and the image of one of the scenes under all 11 illuminants is shown in Figure 4.11. The images had a varying amount of specularities. For example, the image labeled macbeth has few specularities, whereas books-4 is quite specular. We tried to avoid images with coloured metals or fluorescent surfaces, as none of the algorithms tested deal with this kind of input.

The experimental routine was as follows: First a new scene was constructed. We then placed a reference white standard in the center of the scene, perpendicular to the direction of the illuminant. The position of the illuminant was set so that the number of clipped pixels was small. This meant that if the scene had bright specularities, then the image was purposely under-exposed. We then took a picture of the scene with the reference white in the center. Finally, we removed the reference white, and took 50 successive pictures which were averaged to obtain the final input image. We then repeated the process for the remaining 10 illuminants, and then we moved onto the next scene.

The images with the reference white were used to provide the answer. We extracted the central 30 by 30 pixel window of each of these images, and used the average (R,G,B) over these windows as the estimate of the illuminant for the corresponding input images. We note that both the input image and this target value were first mapped into a more linear space, and received the other corrections discussed more fully above. We believe that this method provided a good estimate of the chromaticity of the illuminant, but that the error in the illuminant magnitude for any given picture could be quite high–easily 10%, because of the difficulties in keeping the white reflectance standard perpendicular to the light source. Furthermore, three of the sources were distended, and here we simply attempted to find the orientation which maximized the brightness of the reflectance standard.

In Table 4.5 we present the results using a generic pre-processing method that works relatively well with most algorithms, but is optimal for none of them. Table 4.5 also includes results of input modified to be closer to the input provided our camera when the automatic aperture is used. These results were obtained by artificially scaling and clipping the image data so that the maximum (R,G,B) of the reference white would be 300, and all pixels with R, G, or B over 255 were discarded. In Table 4.6 we provide some results of our experiments with pre-processing. Here we show the range of results obtained using the 65 pre-processing methods described above. Finally, in Figure 4.12 we compare real results with synthetic results. To make this comparison, we note that the performance of the algorithms on our image data set is roughly comparable to their performance on synthetic scenes with 8 surfaces. The image data results in this figure are the ones using the optimal pre-processing method for each algorithm.

The most significant deviation of the real image results from the generated ones is that the statistically based algorithms lose ground to the gamut-mapping algorithms in their ability to estimate the chromaticity of the illuminant. For example, on the image data, with generic pre-processing, the C-by-C-MMSE method is only 5% better than the CIP-01 method, down from

### A COMPARISON OF COLOUR CONSTANCY ALGORITHMS

having half the error in the 8 surface synthetic case. And the (R,G,B) gamutmapping algorithms perform the best on average, compared with being somewhat worse than both the neural network and Colour by Correlation methods in the synthetic case. We will discuss this discrepancy in more detail below. For now, we note that Figure 4.12 indicates that specularities do not explain it.

Our experiments with pre-processing confirmed that pre-processing can have a significant effect on algorithm performance. The difference between the average method and the best method is usually greater than 10%, and in some cases, such as the two grey world algorithms, it is of the order of 30%. Furthermore, this effect seems to be quite algorithm dependent, which supports our philosophy that a careful comparison of colour constancy algorithms must take this into account. Our current strategy for doing this is to provide comparisons based on the optimal pre-processing chosen on an algorithm by algorithm basis as we do in Table 4.7 and Figure 4.12.

Finally we note that the simulated clipping we applied to this data did not have a major effect on the numbers. This is because overall, our image database did not include an over abundance of extreme specularities, and only about one third of the images had significant specularities. The small effects that did occur are generally consistent with the results with generated data, except that RETINEX was degraded slightly less compared to the threedimensional gamut-mapping algorithms than was predicted. However, overall, the impact of the clipping results on our conclusions is negligible, as it induces practically no change in the rank ordering of the algorithms. Thus our conclusions hold for moderately specular images, even if a significant number of those specularities are clipped.

# **INSERT COLOUR PLATE HERE**

Figure 4.10: The images of the scenes used for real data under the canonical illuminant.

# **INSERT COLOUR PLATE HERE**

**Figure 4.11**: The ball scene under each of the eleven illuminants used for image data.

		Extended dy	Standard dynamic range			
Algorithm	Illuminant Estimate Angular Error	Illuminant Estimate rg Error	Illuminant Estimate RGB Error	Illuminant Estimate R+G+B Error	Illuminant Estimate Angular Error	Illuminant Estimate rg Error
NOTHING	17.9	0.1252	*	*	17.9	0.1252
AVE-ILLUM	12.9	0.0944	*	*	12.9	0.0944
GW	15.0	0.1196	157.8	254.4	15.0	0.1198
DB-GW	12.9	0.1042	116.7	177.6	13.0	0.1044
RETINEX	8.9	0.0603	92.8	142.6	9.1	0.0618
CIP-MV	24.2	0.1794	*	*	23.2	0.1723
CIP-AVE	16.8	0.1172	*	*	16.0	0.1125
CIP-ICA	10.8	0.0785	*	*	10.6	0.0772
NEURAL-NET	9.9	0.0716	*	*	9.9	0.0716
SP-NEURAL-NET	9.9	0.0717	*	*	9.9	0.0717
C-by-C-01	10.9	0.0803	*	*	10.1	0.0744
C-by-C-MAP	10.6	0.0758	*	*	10.6	0.0758
C-by-C-MLM	10.5	0.0753	*	*	10.5	0.0753
C-by-C-MMSE	10.4	0.0749	*	*	10.4	0.0749
CRULE-MV	5.8	0.0432	88.2	135.3	6.2	0.0457
CRULE-AVE	6.7	0.0491	272.1	421.6	7.7	0.0578
ECRULE-MV	5.4	0.0406	89.4	137.5	5.9	0.0437
ECRULE-AVE	6.6	0.0473	226.6	359.1	7.3	0.0527
ECRULE-ICA	6.8	0.0482	229.5	361.1	7.4	0.0529

**Table 4.5**: Algorithm performance for 321 real images using a generic pre-processing method (uncertainty is roughly 4%). Specifically, the images were segmented subject to the constraint that the (r,g) vector distance of any two pixels in the region was not more than 0.05, that the value of R+G+B did not vary by more than 10%, and that the region was at least 5 pixels. Connections between pixels were only over horizontal and vertical boundaries. Once segmented, the average of (R,G,B) over each of the regions was used as the input to the algorithms. The two rightmost columns are the chromaticity results obtained when we scaled and clipped the image to mimic the data from our camera used in a more standard fashion. In this case, the we cannot reduce the exposure to avoid clipping without loosing information in the darker regions of the image. Thus an exposure setting which clips some pixels is a more serviceable compromise, and this is what is emulated.

Algorithm	Minimum illuminant estimate rg error	Illuminant estimate rg error using the generic method	Average illuminant estimate rg error	Maximum illuminant estimate rg error	Pre-processing method for minimum
NOTHING	0.125	0.125	0.125	0.125	All are equal
AVE-ILLUM	0.094	0.094	0.094	0.094	All are equal
GW	0.072	0.120	0.103	0.129	RGB Clustering
DB-GW	0.053	0.104	0.086	0.115	RGB Clustering
RETINEX	0.054	0.060	0.061	0.083	Averaging blocks of 5 pixels
CIP-MV	0.149	0.179	0.186	0.200	Convex hull data reduction
CIP-AVE	0.104	0.117	0.124	0.135	Convex hull data reduction
CIP-ICA	0.078	0.079	0.081	0.086	Segmentation (5,0.005,0.2)
NEURAL-NET	0.064	0.072	0.070	0.081	Segmentation (20,0.005,0.4)
SP-NEURAL-NET	0.061	0.072	0.069	0.083	Convex hull data reduction
C-by-C-01	0.073	0.080	0.079	0.088	Segmentation (20,0.02.0.1)
C-by-C-MAP	0.064	0.076	0.073	0.085	Convex hull after ave(3)
C-by-C-MLM	0.064	0.075	0.072	0.085	Convex hull after ave(3)
C-by-C-MMSE	0.062	0.075	0.072	0.083	Convex hull after ave(3)
CRULE-MV	0.043	0.043	0.047	0.066	Segmentation (10,0.005,0.2)
CRULE-AVE	0.045	0.049	0.052	0.085	Segmentation (20,0.02,0.1)
ECRULE-MV	0.040	0.041	0.043	0.065	Segmentation (5,0.0025,0.1)
ECRULE-AVE	0.046	0.047	0.050	0.079	RGB Clustering after ave(5)
ECRULE-ICA	0.047	0.048	0.051	0.077	RGB Clustering after ave(5)

**Table 4.6**: Algorithm chromaticity performance for 321 real images using the best pre-processing method for that algorithm. Hence each result in this table is obtained using a (potentially) different pre-processing method. We caution the reader from drawing overly strong conclusions from the optimal pre-processing method, as the exact best method is dependent on our data. The main point here is that preprocessing has a significant effect, which is algorithm dependent. For example, the generic method choices for the results in Table 4.6 is not equally flattering to all algorithms, when compared with the minimum possible. The segmentation parameters are (min-segment-size, max-rg-variation, max relative R+G+B variation). We use ave(n) to indicate that N by N blocks were averaged.



#### Comparison of algorithm performance on real and synthetic data

**Figure 4.12**: Algorithm performance for 321 real images using the best pre-processing method for each algorithm compared with synthetic results for a comparably difficult number of surfaces (8)

## 4.9 Discussion

As noted above, the main discrepancy between our synthetic data results and our image data results is the performance of the statistically based methods relative to the three-dimensional gamut-mapping methods. The synthetic results strongly suggest that, given a good match between testing and training statistics, the statistically based algorithms can yield excellent colour constancy. Thus we conclude that the statistics of our training set do not match the statistics in our real image data set, and therefore, they quite likely do not match the statistics of the real world. The success of the threedimensional gamut-mapping algorithms indicate that perhaps a reflectance data set which yielded uniformly distributed (R,G,B) may be more appropriate for the real world than our reflectance data set. This is because gamut mapping is roughly analogous to Colour by Correlation with uniform statistics. For example, we have already mentioned that CIP-01 can be interpreted as an implementation of two-dimensional gamut mapping. Since the three-dimensional algorithms do better, it is plausible that uniform statistics in (R,G,B) space are part of the reason.

To further explore this notion, we re-ran one of the real image experiments with correlation matrices computed using our reflectance data set augmented so that it yielded more uniformly distributed (R,G,B). We found that Colour by Correlation did improve, but only by a small amount. Nonetheless, we observe that since C-by-C-MMSE does better than CIP-01, both our reflectance data set, and the augmented one, are more appropriate for our image data set than the uniform (r,g) statistics implied by CIP-01. It remains an open question as to what extent the statistics of real images can be specified, and how much improvement in Colour by Correlation would result.

The above discussion supports the claim that the three-dimensional gamut-mapping algorithms perform better on real images simply because they can take advantage of brightness information. We reason that since improving the uniformity of the training statistics did not bridge the gap

### A COMPARISON OF COLOUR CONSTANCY ALGORITHMS

between the Colour by Correlation method and the three-dimensional gamut-mapping algorithms, the difference must be elsewhere. Furthermore, the main difference available is the use of brightness information. We also conclude that the good performance of the statistics based algorithms in simulation is due to their ability to take advantage of the specific training statistics. When these statistics are not present, the performance decrement is large. This is a significant problem for real image data given that we are not confident that we know the appropriate statistics.

The use of pixel brightness information by the three-dimensional gamut-mapping algorithms bears further comment. In [111], Finlayson and Hordley prove that, under reasonable assumptions<sup>2</sup>, the perspective gamut-mapping algorithms are as powerful as the three-dimensional ones in their ability to constrain illuminant chromaticity. In other words, all illuminants implied by the Colour in Perspective constraint set are also present in the ECRULE constraint sets. Therefore, differences between the gamut-mapping algorithms are due to other considerations. For real data we cannot ignore robustness with respect to noise, but even with synthetic data we found a significant difference between the perspective and the three-dimensional gamut-mapping algorithms. Thus, the main difference between gamut-mapping algorithms is their ability to estimate the solution from the constraint set, and this is where the three-dimensional ones seem to be better suited.

We have already discussed how the presence of specularities helped the three-dimensional gamut-mapping algorithms choose the solution. But we note that these algorithms do well even when there are no specularities (see, for example, Figure 4.7 or Table 4.2). To understand these results, suppose that averaging the three-dimensional constraint space makes sense. In this space, illuminants within a specific chromaticity range correspond to cones. Now consider the proposition that the illuminant chromaticity is in a small range near the edge of the constraint set. In the perspective case, this

<sup>&</sup>lt;sup>2</sup>The assumption is that in three-dimensional case, the origin is included in the canonical gamut. We agree that this is a reasonable assumption because surfaces may be arbitrarily dark due to shading.

small range is similar to any other of the same size. In the three-dimensional case, however, the corresponding cone has less volume than one closer to the middle of the constraint set. In general, the volume is a function of the shape of the three-dimensional constraint set; the perspective case cannot make use of the this extra information. Similarly, the maximum volume heuristic also chooses the result using information which is not available to the perspective algorithms.

### 4.10 Conclusions

We have investigated computational colour constancy methods under a variety of circumstances using both synthetically generated data, and a large, carefully prepared set of real images. The synthetic experiments are needed because they can be used to manipulate the various aspects of the input to the algorithms in a controlled manner. For example, using the synthetic experiments we were able to look at the distribution of the chromaticity error of the algorithms. In this experiment we verified the suspected bias in the Colour in Perspective method, and identified several other slight biases among the algorithms. More importantly, we were able to provide evidence that the algorithms of most interest are relatively free of bias. A second experiment looked at the performance of the algorithms as the number of surfaces increased. Here we found more complexity then we had expected, with the rank order of performance changing multiple times as the number of surfaces increased. In general, however, for a synthetically generated scene with a relatively small number of matte surfaces, the probabilistic versions of Colour by Correlation and the neural net method were the most able illuminant chromaticity estimators, followed closely by the threedimensional gamut-mapping algorithms with the maximum volume heuristic. A third experiment investigated the effect of the presence of significant specular reflection. Most methods performed better under these conditions with RETINEX, CRULE-MV, and ECRULE-MV making the largest gains, and with the latter two becoming comparable to the overall leader (Cby-C-MMSE). Finally, since such strong specularities are often clipped due to

limited dynamic range, we examined the effect of clipping the data with specularities. We found that among the top algorithms, the probabilistic versions of Colour by Correlation and the neural net method were the most robust with respect to clipping. Thus, in general, we conclude that ECRULE-MV is an excellent algorithm when abundant dynamic range is available, as could be the case in a specialized computer vision system. In addition, for processing more standard images, C-by-C-MMSE and NEURAL-NET are both very promising.

Unfortunately, we were not able to realize this promise with real images. Despite carefully considering the issues of camera calibration, image pre-processing, and choice of training sets, we found a relatively large discrepancy in the performance of Colour by Correlation and the neural net in the real and synthetic data case, as compared to other algorithms. Thus we conclude that there was a mismatch between the statistics used for training, and the statistics implied by our image data base. Furthermore, it may prove difficult to find an appropriate statistical model for real images, at least in the general case. It may even be the case that the statistical methods are overly sensitive to specific details of the training data, and if this is the case, they will require modification before being able to properly generalize to image data.

# **Chapter Five**

## Sensor Sharpening for Computational Colour Constancy

Sensor sharpening refers to using an appropriate linear transformation of camera responses (RGB), in place of the original responses, in order to improve coefficient colour constancy [38]. However, doing so has not yet been tested in the context of real colour constancy algorithms. Rather, the experimental results available are limited to finding the minimum error possible with and without sharpening. Since the error in current colour constancy methods is often substantially larger than the minimum error possible, we felt it necessary to investigate further the utility of sensor sharpening for colour constancy. In order to apply sensor sharpening to this practical domain, we were forced to make several decisions described below. Having done so, we were able to improve colour constancy performance in a few specific cases. However, the problems we faced lead us to develop a new method of sharpening which we feel is more appropriate in the context of computational colour constancy. The results obtained using this new method are very promising.

The efficacy of sensor sharpening is known to be dependent on the camera sensors. Thus we provide results for 3 different cameras: A Sony DXC-

930 CCD video camera (calibrated as described in Chapter 3), a Kodak DCS-200 digital camera [10], and a Kodak DCS-420 digital camera [10]. Our general conclusion is that when the sensors are already relatively sharp (e.g. the Sony camera), further sensor sharpening is not worth the trouble, and often has a detrimental effect. However, when the sensors are not sharp (e.g. the DCS-200 and DCS-420), sensor sharpening can have a substantial positive effect, depending on the algorithm, especially when the new method is used.

### 5.1 Sensor Sharpening Theory

We begin with an explanation of sensor sharpening [38]. The motivation for sensor sharpening is the observation that most colour constancy algorithms make use of a diagonal model of illumination change. To understand this model, consider a white patch under two different illuminants. Suppose that under the first illuminant the colour is [r,g,b] and under the second illuminant the colour is [r', g', b']. It is possible to map the colour of white under the first illuminant to the colour under the second by post-multiplication by a diagonal matrix:  $[r', g', b'] = [r, g, b] \operatorname{diag}(r'/r, g'/g, b'/b)$ . If the same diagonal matrix transforms the RGB of all surfaces (not just the white ones) to a good approximation, then we say that we have a diagonal model of illumination change. It turns out that the accuracy of the approximation is a function of the vision system's sensors.

The idea of sensor sharpening is to map the data by a linear transform T into a new space where the diagonal model holds more faithfully. Colour constancy algorithms which rely on the diagonal model can then proceed more effectively. The final result is then mapped back to the original RGB space with the inverse transformation. Working in the transform space is like having new sensors which are a linear transformations of the old ones. Further, the sensitivity functions of sensors that support the diagonal model tend to look sharper with narrower peaks than ones that do not—in the extreme case, if the sensors are delta-functions, the diagonal model holds exactly. From these observations, we get the name: sensor sharpening.

### SENSOR SHARPENING FOR COMPUTATIONAL COLOUR CONSTANCY 114

The main technical result in sensor sharpening is finding the transformation T. In [38], Finlayson et al. propose three methods for finding T: "sensor based sharpening", "database sharpening", and "perfect sharpening". For this work we chose database sharpening over sensor based sharpening due to the clean correspondence between the sharpening method and a colour constancy error measure. Perfect sharpening did not work well for us because our test illuminant set did not meet the key requirement of being two-dimensional, partly due to the inclusion of fluorescent lights.

In database sharpening, RGB are generated using a database of reflectance spectra, together with an illuminant spectrum and the sensors. This is done for two separate illuminants. Let A be the matrix of RGB for the first illuminant and B be the matrix for the second, with the RGB's placed row-wise. In the sharpening paradigm we map from B to A with a sharpening transform, followed by a diagonal map, followed by the inverse transform. If we express each transform by post multiplication by a matrix we get:  $A \approx BTDT^{-1}$ . In database sharpening the matrix T (and implicitly D) is found that minimizes the RMS error,  $||A - BTDT^{-1}||_2$ . The sharpening transform gives exactly the same error as the best linear transform M. In fact, T is found by diagonalizing M, where M minimizes  $||A - BM||_2$ .

Two implementation issues should be noted. First, the above procedure can lead to complex elements of T. Fortunately, this does not occur too often, and when it does occur, the imaginary components tend to be small. We set all imaginary components to zero.

A second implementation issue is as follows: The result of the diagonalization is ambiguous up to scaling and swapping of the columns of T. As is standard, we use columns of norm 1. Furthermore, we put the element of T of largest absolute value on the diagonal by swapping columns, and ensure that it is positive by multiplying the column by -1 if necessary. Then in a similar way we attempt to make the other diagonal elements as large as possible. This procedure is used to reduce the number of negative components of sharpened data.

In this work we view colour constancy as finding a transformation from the image of a scene taken under an unknown illuminant, to the image of the same scene as though it were taken under a known, "canonical", illuminant [32]. A priori, the nature of the transformation is open, but most algorithms find a diagonal transform, and it is these algorithms which interest us here. Of course, the best linear transformation will give at most the same error as any diagonal transformation, but it should be clear from the above that the generalized diagonal transform TDT<sup>-1</sup> gives us a chance of having this lower error with a diagonal model [39]. Thus we should be able to improve diagonal colour constancy if the right sharpening transform can be found.

### 5.2 Sensor Sharpening in Practice

If we wish to use the database sharpening transform described above, we immediately run into a problem. The computation is defined in terms of two illuminants. We know one of these illuminants (the canonical), but the other illuminant is precisely what we seek. Thus to carry on with the method, we need to find a substitute. One approach is to use the average of a database of illuminants which represent the illuminants that are expected or possible. We implement this by averaging the comprehensive, training set of illuminants introduced in Chapter 4. The illuminants in this set must be normalized for this method to make sense. We have also experimented with using gamut-mapping colour constancy processing to constrain the illuminant set, allowing us to take the average over a smaller set of illuminants which have been identified as being feasible. We then used this average to re-compute the sharpening transform, and then we re-computed the colour constancy result with the new sharpening. Although this iterative method has some intuitive appeal, we found that the results were not much better than using the simple average, and thus we do not include this method in the experiments reported here.

Even though applying the true database sharpening method is impossible in the colour constancy context, we still would like to know the upper bound of the expected performance using this method. Thus we also implement standard database sharpening. Here T is computed assuming that we can correctly "guess" the illuminant spectrum. Since T minimizes possible RMS mapping error, we will call this method "optimal". It should be noted, however, that for such a method to be truly optimal, the sharpening must work with, or at least not against, the specific algorithm. When the sensors are already sharp, it is more likely that the breakdown of this assumption becomes noticeable because a smaller portion of the error is due to the lack of sharpness. Thus with sharp sensors, using this "optimal" sharpening often gives worse results than not using sharpening at all. This emphasizes the main point of this chapter: Sharpening does not minimize the error of any specific practical computational colour constancy method. Thus using sharpening for this task requires further study. In this chapter we expedite the investigation by using an empirical approach.

An empirical approach is useful because the interaction between sharpening and the colour constancy algorithms is complex. One problem is that the sharpened camera responses may have negative components. Such negative components can be a problem for the algorithms based on the gamut mapping approach introduced by Forsyth [32], and extended and modified by others [8, 40]. Unfortunately, these algorithms are the ones we are most interested in improving with sharpening. There are two reasons for this. First, they are very effective algorithms (see Chapter 4), and second, their complete reliance on the diagonal model makes them ideal candidates for improvement. With these algorithms, problems with negative sensor responses can extend beyond their occurrence in input data, as these algorithms have calibration sets which must also be mapped into the sharpened space. Depending on the variation used, negative components in calibration data can present severe difficulties.

If we insist on implementing gamut mapping algorithms with sharpening, we must decide what to do with negative components. Of course, we want to have a strategy which does not impact the algorithms more than necessary. One strategy is to set the negative components to zero, or some small positive number (zero is also a problem for some of the data used by some of these algorithms). Unfortunately, modifying data in this artificial way stands a good chance of leading to poor performance, or at least worse performance than without sharpening. Similarly, simply throwing out such data, which could be useful to the algorithm if sharpening was not used, could have a negative impact which is hard to estimate. These considerations lead us to the following strategy: If an algorithm is faced with negative components in a situation where they are not acceptable, then sharpening is simply not used. Instead, the standard results without sharpening is computed. In the results we report the number times this occurred for each combination of algorithm and sharpening method tested. This strategy reliably exposes the benefits (or lack thereof) of sharpening, and makes sense in conjunction with our new method described shortly.

The difficulty with negative components in the case of the gamut mapping algorithms is a simple, specific case, of a more general problem. Given a colour constancy algorithm, does it make sense to carry it out in a specific sharpened space? In the discussion of the colour constancy methods we will make a few comments on this topic with respect to specific algorithms, but in general, a good understanding of the effect of sharpening on colour constancy algorithms is not available, and we offer our empirical results as a first step in achieving a better understanding.

## 5.3 The New Approach to Sharpening

We now introduce a simple method for sharpening which deals with some of the problems with database sharpening. We begin by changing the objective function of the database sharpening minimization problem. In its original form, this objective function minimizes the diagonal mapping error between the responses of a set of reflectances under two known illuminants. However, since in our context, one of these illuminants is not known, we minimize the average of the mapping error over a representative set of possible illuminants. Next we include a term that encourages positive sensor response values. Finally, we include a term that encourages the transform to be norm one. Since the resulting function is difficult to minimize directly, we look for suitable local minimums using gradient descent. The computation of the objective function requires a database of reflectance spectra which model the surfaces of the world, and a database of illuminants which is a representative set of the illuminants in the world. The illuminant set should be normalized. Then, for a given illuminant, *i*, all the responses to all the reflectances can be computed using the camera sensors. We put the responses into the rows of a matrix A<sub>i</sub>. We also form a similar matrix for the canonical illuminant B. Then, given a transform, T, we compute the sharpened responses by A<sub>i</sub>T and BT. We then compute the best diagonal map, in the least squares sense, between these two matrices. This is done by averaging the rows of each matrix, dividing the resulting three-tuples element-wise, and using these three ratios as the diagonal elements of a diagonal matrix. For each illuminant we thus obtain a diagonal map, D<sub>i</sub>. Finally, we can express the overall mapping error by:

$$E = \sum_{i} \left\| A_{i} T D_{i} T^{-1} - B \right\|_{F}$$
(5.1)

Where  $\|\|_{F}$  is the Frobenius matrix norm, computed by summing the squares of all elements, and then taking the square root of that sum.

To encourage positive sensor responses, we compute a penalty function from all the current sharpened responses,  $A_iT$ . Since some algorithms cannot cope with zeros, we encourage the responses to be at least a small offset larger than zero (we used 0.1 in the experiments). The penalty function we use is:

$$f(x) = \begin{cases} (x - offset)^2 & \text{if } (x < offset) \\ 0 & \text{otherwise} \end{cases}$$
(5.2)

The total penalty, P, is the sum of f(x) over all sensor responses A<sub>i</sub>T over all illuminants, *i*. Finally we add a term to keep T near norm one. Without such a term, the positivity term could be reduced by simply scaling T downwards. For this we use:

$$N = (trace(T'T) - 3)(trace(T'T) - 3)$$
(5.3)

The overall objective function was therefore:

$$\mathbf{E} + \lambda_{\mathrm{P}} \mathbf{P} + \lambda_{\mathrm{N}} \mathbf{N} \tag{5.4}$$

where  $\lambda_{\rm P}$  and  $\lambda_{\rm N}$  are Lagrange multipliers which control the relative importance of the three parts of the objective function. Finding a suitable value of  $\lambda_{\rm N}$  does not require much effort because we can use a very large value without penalty. This is because we are essentially using this term as a convenient way to enforce the size constraint. We note that the magnitude of T has no effect on the mapping error due to the occurrence of both T and T<sup>-1</sup> in the expression. Once  $\lambda_{\rm N}$  is sufficiently large, increasing it further will have very little additional impact on the resulting T. On the other hand, increasing  $\lambda_{\rm P}$  will lead to larger overall mapping error, and thus we would like to use the smallest value possible. This is discussed further below.

Thus given a T, we can compute the error, and, given a change in any of the nine components of T, we can compute the change in the error, and thus the gradient. We then change T in the direction of the gradient. If this change increases the error, we back off the amount of change until the error is truly decreased. We carry on the process until T does not change more than a small threshold.

This leaves the determination of the initial T. Since gradient descent does not necessarily converge to the global minimum, different starting points can give different results. We tried two different strategies. First, we tried starting with the sharpening matrix obtained using the average illuminant, as described above. Second, we tried starting with the identity matrix. Both methods gave good results, but, over the three camera sensors used, the identity matrix starting point proved to be the better choice, and we used this method for the results reported.

Sharpening with positivity requires further comment. Positivity has been added to sensor based sharpening [112] using two different criteria. As discussed in that work, sharpened sensor responses can be made positive by making the sharpened sensors themselves positive. The first criterion used in that work is to insist that the coefficients of the sharpening matrix are positive. The second criterion, which is weaker, and thus more useful, takes a more direct approach and simply insists that the sensors themselves are positive. The conditions on positivity imposed with our method are even weaker in two ways. First, we are only interested in positive sensor responses under expected signals; the sensors themselves can have small negative components. Second, our penalty function allows for the possibility of an occasional small negative sensor response. In fact, this is how we set the Lagrange multiplier for the positivity term. We use a value which yields "positive enough" responses for the algorithms under consideration. Thus conceptually, the Lagrange multiplier is a function of the algorithm(s) to be improved. Given our strategy of reverting back to the standard form of the algorithm if necessary, we can tolerate the occasional negative sensor response. However, since we wished to use the same sharpening transform for all the algorithms under consideration, we found it convenient to choose a value for the Lagrange multiplier which yielded very few negative sensor responses.

Finally, we note that our method also sidesteps having to deal with the occurrence of complex elements in the transform matrix. As mentioned above, these sometimes occur when standard database sharpening is used.

## 5.4 Error Measures

The emphasis of this work on sharpening compels us to measure the performance of colour constancy algorithms using the same kind of error that is minimized by the sharpening methods. Specifically, we compute the RMS RGB difference between the sensor responses of the entire reflectance set under the canonical illuminant and the responses of this world under the unknown illuminant. This is error is slightly different than the similar mapping measure used in Chapter 4 which uses the mapping error between the image of the specific scene instead of the entire world. These two measures give very similar results.

To clarify the computation of this error, we first note that when sharpening is used, the algorithms are run entirely in the sharpened space. Thus they deliver a diagonal transform which takes sharpened responses under the unknown illuminant, to sharpened responses under the canonical illuminant. We apply the diagonal transform to the sharpened responses from all our reflectances as seen under the canonical illuminant. This provides an estimate of the responses under the canonical in the sharpened space. We then apply the inverse sharpening transform to this mapped set to obtain the estimate of the responses under the canonical without sharpening. This estimate is compared to the target using the RMS of the RGB difference between them.

We now move on to a second error measure used in this study. It is common to consider computational colour constancy as estimating parameters of the scene illumination. Specifically, it is quite common to estimate the RGB of the illuminant, which is defined as the RGB of a perfect white under that illuminant. However, we are often less interested in the overall brightness of the illuminant. For example, if we are correcting a properly exposed image taken with incorrect camera balance, then the brightness of the illuminant is implicit, and we only need to correct for the illuminant chromaticity. In fact, a number of algorithms work only with chromaticity [40, 62, 63]. As discussed further below, we do not consider any of these algorithms here, but we feel that the best counter-point to the RMS RGB mapping error measure is a chromaticity error measure. The one we use here is the angle between the illuminant RGB and the estimate thereof, considered as vectors in RGB space.

### 5.5 Colour Constancy Methods

We will now discuss briefly the colour constancy algorithms investigated here. The details for all these algorithms are provided in Chapter 4. For this work we use the two versions of the Grey World method (GW and DB-GW in §4.5.1), the version of Retinex described in §4,5.2, and some of the gamut-mapping algorithms described in §4.5.3. We do not report any results for the chromaticity versions of gamut mapping. There are three reasons for this. First, since these methods do not consider overall illuminant brightness, their results cannot be used in conjunction with the RGB mapping error. Second, we have found the three-dimensional methods more effective overall (Chapter 4, [8, 107]). And third, we have found that the chromaticity gamut mapping methods are by far the most sensitive to negative sensor responses. We have verified that the new method described is in fact an excellent choice if one wishes to use sharpening in conjunction with these methods, precisely because it addresses the negative sensor response problem in a flexible way. However, doing so requires a much larger value of  $\lambda_{\rm P}$  than needed by the three-dimensional gamut mapping algorithms.

## 5.6 Colour Constancy with Sharpening

Each colour constancy algorithm relates differently to sharpening. We consider the grey world algorithms first. The illuminant estimate provided by these two algorithms is given by:

$$\mathbf{W} = \left(\frac{1}{n}\sum_{i}^{n}\mathbf{r}_{i}\right) \operatorname{diag}(\mathbf{w}_{c}./\mathbf{g}_{c})$$
(5.5)

where the  $\mathbf{r}_i$  are the observed sensor responses,  $\mathbf{w}_e$  is the RGB of white under the canonical,  $\mathbf{g}_e$  is the RGB of grey under the canonical, ./ is used to denote element-wise division, and diag() transforms a vector to a diagonal matrix. When sharpening is used this becomes:

$$\mathbf{W} = \left( \left( \frac{1}{n} \sum_{i}^{n} \mathbf{r}_{i} T \right) \operatorname{diag}((\mathbf{w}_{c} T) \cdot / (\mathbf{g}_{c} T)) \right) T^{-1}$$
(5.6)

As discussed above, we have two different definitions of grey, giving two different values of  $\mathbf{g}_{e}$ . In the case of the GW algorithm, the reflectance spectrum of grey is assumed to be half that of perfect white. Under this condition,  $\mathbf{w}_{e}$  is twice  $\mathbf{g}_{e}$  and diag( $\mathbf{w}_{e}$ ./ $\mathbf{g}_{e}$ ) can be replaced by 2. Similarly,  $\mathbf{w}_{e}T = (2\mathbf{g}_{e})T = 2(\mathbf{g}_{e}T)$ , and diag(( $\mathbf{w}_{e}T$ )./( $\mathbf{g}_{e}T$ )) can also be replace by 2. It is then straightforward to see that (5) and (6) are the same. Hence, we expect sharpening to have absolutely no effect on the estimate of the illuminant using the GW algorithm.

It should also be clear that when  $\mathbf{g}_{c}$  is not a simple scalar times  $\mathbf{w}_{c}$ , as is the case with DB-GW, that there will be some effect. Intuitively, the effect should be small, as we expect  $\mathbf{g}_{c}$  to be not too far from a scalar times  $\mathbf{w}_{c}$ .

However, given the nature of the expression, it is possible for the effect to be large. For example, the illuminant estimate becomes unstable as we consider sharpening transforms T which map any of the components of  $\mathbf{g}_{c}$  close to zero.

The possible effects in the case of the mapping error measure are more interesting. In the non-sharp case, the grey world algorithms estimate the diagonal mapping, D, mapping RGB taken under the unknown illuminant to the corresponding RGB taken under the canonical illuminant by:

$$\mathbf{D} = \operatorname{diag}\left(\mathbf{g}_{\mathbf{c}} \cdot \left(\frac{1}{n} \sum_{i}^{n} \mathbf{r}_{i}\right)\right)$$
(5.7)

In the sharp case, the estimate is for the analogous mapping,  $D^{\#}$ , which is applied in the sharpened space.  $D^{\#}$  is computed by:

$$D^{\#} = \operatorname{diag}\left(\left(\mathbf{g}_{\mathbf{c}} \mathbf{T}\right) \cdot \left(\frac{1}{n} \sum_{i}^{n} \mathbf{r} \mathbf{T}_{i}\right)\right)$$
(5.8)

Now, if the sharpening transform produces negative components, it is entirely possible that the sum in (8) can be close to zero. In this case, the error in  $D^{\#}$  can be become arbitrarily large. Thus, even in the case of the GW algorithm, where the illuminant estimate is identical to that obtained without sharpening, the RGB mapping error can be much larger than that obtained without sharpening. This may seem contradictory. To clarify, we consider computing the mapping using the estimate of W obtained in (5.5) and (6). In the non-sharp case, we have:

$$\mathbf{D} = \operatorname{diag}(\mathbf{w}_{c}./\mathbf{W}) \tag{5.9}$$

In the sharp case, we have:

$$\mathbf{D}^{\#} = \operatorname{diag}((\mathbf{w}_{c}\mathbf{T})./(\mathbf{WT}))$$
(5.10)

Thus we see that the illuminant estimate corresponding to the troublesome mapping has components close to zero in the sharp space, but not in the nonsharp space.

We now move onto the RETINEX algorithm. This method is not likely to have the problem described above, as the estimate is now the maximum of each channel, and these maxima are likely to be significantly greater than zero for any reasonable sharpening method. One consideration of relevance, however, is the degree to which the estimate makes sense in a sharpened space. With standard camera responses, the maximum of each channel is justified, because, at least under simplistic assumptions, it will converge to the response of white, as the diversity of the surfaces present in the scene increases. This is not necessarily the case when sharpening is used. For example, suppose that:

$$\mathbf{T} = \begin{vmatrix} 1 & 0 & -\frac{1}{3} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$
(5.11)

Further, suppose that the RGB corresponding to white is (1,1,1). Then the RGB corresponding to white in the sharpened space is  $(1,1,1)^*T=(1,1,\frac{2}{3})$ . In general, the RGB in the sharpened space are given by  $(R,G,B-\frac{1}{3}R)$ . It is quite possible that there are valid sensor responses which are larger in the third coordinate than the one corresponding to white. For example, if  $(\frac{1}{2},1,1)$  is a possible sensor responses, then the maximum of the third channel could reach  $\frac{5}{6}$ , which is greater than the value of  $\frac{2}{3}$  for white. It is unclear to what extent this will be a problem in practice for a given camera and sharpening combination. Furthermore, this possible negative effect needs to be weighed against the possible benefits of sharpening, namely the reduced error when we use the illuminant estimate to compute an estimate of the mapped image.

The story for the gamut mapping methods is somewhat different. Here we estimate the diagonal maps first, and then, if necessary, use the maps to compute illuminant estimates. We remind the reader that the estimation of the maps consists of two parts. First we compute the set of possible maps using applicable constraints. Then we choose a map from this constraint set. The first part is highly dependent on the diagonal model. Thus gamut mapping algorithms are likely candidates for improvement by the use of sharpening.

One possible problem in realizing this improvement is that a given method of choosing a solution may be degraded in the sharpened space. This problem is similar to the problems analyzed above for the grey world and Retinex based algorithms, but the analysis in this case is even more complex. We note that using the average of the possible maps was originally proposed to reduce the error using RGB mapping measure [8], and in Chapter 4 we found that this method was the best choice when this measure is of most interest (Table 4.2). As noted above, this measure makes the most sense in the analysis of sharpening. Furthermore, we expect that the arguments which justify this choice in the standard case are not overly eroded in the sharpened case. Even if sharpening is used, we still expect that the solutions will be distributed throughout the constraint set without overly large bias, and that the computation of the corresponding mapping error will not add too much additional bias.

On the other hand, the maximum volume heuristic may be sensitive to problems similar to those outlined above for the Retinex method. This heuristic is not fully understood in the standard case, and things become even more complex when sharpening is added. We note that this method of choosing the solution is superior to averaging when evaluated using illumination estimation error measures (see Chapter 4). Thus we suspect that the method is essentially oriented towards this task, despite being implemented in mapping space.

In summary, the effect of sharpening on the various algorithms is hard to predict. This is especially the case for the gamut mapping algorithms which are the conjunction of two different parts, each of which interacts with sharpening differently. Thus we believe that the most effective way to reduce our ignorance is to look at empirical results which we do next.

## 5.7 Results

We have investigated sensor sharpening in the case of a Sony DXC-930 CCD video camera (calibrated as described in Chapter 3), a Kodak DCS-200 digital camera [10], and a Kodak DCS-420 digital camera [10]. We provide results for each camera in turn, first numerically, in Tables 5.1, 5.2, and 5.3, and then graphically with respect to the RGB mapping error measure in Figures 5.1, 5.2,

and 5.3. The training and testing sets were as described in detail in Chapter 4. The results are the RMS average results for 1000 synthetically generated scenes, each using a randomly selected illuminant and 8 randomly chosen surfaces. For each algorithm we computed the results without sharpening, and with the three sharpening methods describe above. These were the optimal database method, which uses the actual illuminant (not normally available), database sharpening using the average training set the illuminants, and the gradient descent method.

In addition to the algorithms described above, we include the results for several comparison "algorithms". The first is the best possible result, obtained by finding the best linear fit between the RGB of the reflectance database under the canonical and test illuminants. This result is invariant to matrix multiplication, and thus is the same for all sharpening methods. The second comparison method is the best diagonal map. The optimal sharpening method minimizes the error with this method, and, as describe above, the resulting error is essentially that for the best linear map—the only difference being due to difficulties with complex elements in the sharpening matrix. The third comparison method is the result obtained by using the RGB of white under the test illuminant. The error with this method is zero using illuminant-based error measures, and would also be zero in the case of mapping error, if the diagonal model held perfectly. This comparison method provides an additional insight into whether the diagonal model is much improved, without the complexities introduced with the interactions with the algorithms. Finally, we provide the result of doing nothing in the case of the illumination-based error measure. Doing so in the case of the RGB mapping error would be misleading, as the database of illuminants is artificially normalized. (This has no effect on the other algorithms).

As expected, the results are quite camera dependent. We define the degree of sharpness of the camera sensors by the ratio of the BEST-LINEAR result to the BEST-DIAGONAL result. By this measure, the Sony DXC-930 camera is quite sharp (0.70), and the Kodak DCS-420 and the Kodak DCS-200 are less sharp (0.34 and 0.33 respectively). In the case of the already sharp camera, we found that there was very little to be gained by sharpening.

Interestingly, the average illuminant and optimal sharpening methods often made the results substantially worse. In the case of the optimal sharpening method together with the GW, DB-GW, and RETINEX algorithms, the results with the mapping error measure were unstable due to the inversion of numbers near zero, and the results were extremely poor. By contrast, the new method did not tend to make matters worse. Here the results were essentially the same as that without sharpening for every algorithm.

For the two Kodak cameras, the optimal sharpening method was again often unstable with the GW, DB-GW, and RETINEX, and again, even the better behaved average sharpening method tended to make matters worse. The gradient descent method fared better, being comparable within error to no sharpening with GW and DB-GW, and slightly better than no sharpening with RETINEX.

With the ECRULE-ICA algorithm, we found that sharpening has a large positive effect as gauged by both error measures. Here the new method is close in performance to the optimal sharpening method. In the case of the Kodak DCS-200 camera, the average illuminant sharpening method could not be used with this algorithm due to the problem with negative components, and thus gives the same result as no sharpening. In the case of the Kodak DCS-420 camera, the average illuminant sharpening method was viable, but it gave a poor result. In summary, for the ECRULE-ICA algorithm, the new method offers the same significant benefits as the optimal sharpening method, but at the same time is realizable in practice.

The results for the ECRULE-MV algorithm are a little more complex. The results using the new method again parallel the results for the optimal sharpening method, but with this algorithm, sharpening has a moderate negative effect with the mapping measure, and a significant positive effect with the illuminant based measure! Thus if the goal is to estimate and correct for chromaticity, then sharpening is still recommended.
Algorithm	Sharpening method	RMS RGB difference between mapped image and target image	Angle between illumination RGB and estimate thereof	Count of times standard results replaced sharp result due to problems with negative components
BEST-LINEAR		3.0	*	0
BEST-DIAGONAL		4.3	0.30	0
BEST-DIAGONAL	(opt)	3.0	0.31	0
BEST-DIAGONAL	(ave)	9.0	0.96	0
BEST-DIAGONAL	(gd-identity)	4.2	0.33	0
ACTUAL		4.4	0.00	0
ACTUAL	(opt)	5.4	0.00	0
ACTUAL	(ave)	20.4	0.00	0
ACTUAL.	(gd-identity)	4.3	0.00	0
NOTHIING		*	16.45	0
GW		184.5	8.35	0
GW	(opt)	8686.7	8.35	0
GW	(ave)	198.3	8.35	0
GW	(gd-identity)	185.4	8.35	0
DB-GW		46.9	6.60	0
DB-GW	(opt)	3421.2	8.80	0
DB-GW	(ave)	61.0	6.60	0
DB-GW	(gd-identity)	47.3	6.59	0
RETINEX		151.1	9.33	0
RETINEX	(opt)	620.7	10.75	0
RETINEX	(ave)	151.2	9.55	0
RETINEX	(gd-identity)	149.5	9.27	0
ECRULE-MV		71.4	5.80	0
ECRULE-MV	(opt)	71.6	5.79	354
ECRULE-MV	(ave)	75.7	6.19	0
ECRULE-MV	(gd-identity)	71.1	5.81	0
ECRULE-ICA		35.7	7.00	0
ECRULE-ICA	(opt)	35.6	6.77	355
ECRULE-ICA	(ave)	38.1	7.10	0
ECRULE-ICA	(gd-identity)	35.9	7.04	3

**Table 5.1**: Sharpening results for the Sony DXC-930 digital camera. Assuming that the algorithms provide colour constancy estimates which are normally distributed around the target values, the uncertainty in the numbers here is roughly 2%. An asterisk is used for values which are not relevant or appropriate.

Algorithm	Sharpening method	RMS RGB difference between mapped image and target image	Angle between illumination RGB and estimate thereof	Count of times standard results replaced sharp result due to problems with negative components
BEST-LINEAR		1.6	*	0
BEST-DIAGONAL		4.9	0.41	0
BEST-DIAGONAL	(opt)	1.6	0.07	0
BEST-DIAGONAL	(ave)	6.9	1.32	0
BEST-DIAGONAL	(gd-identity)	2.3	0.14	0
ACTUAL		5.1	0.00	0
ACTUAL	(opt)	1.7	0.00	0
ACTUAL	(ave)	14.6	0.00	0
ACTUAL.	(gd-identity)	2.3	0.00	0
NOTHIING		*	6.54	0
GW		141.5	2.81	0
GW	(opt)	146.9	2.81	0
GW	(ave)	158.9	2.81	0
GW	(gd-identity)	143.8	2.81	0
DB-GW		40.2	2.40	0
DB-GW	(opt)	45.0	2.47	0
DB-GW	(ave)	56.1	2.31	0
DB-GW	(gd-identity)	41.1	2.39	0
RETINEX		115.2	3.99	0
RETINEX	(opt)	2748.0	10.94	0
RETINEX	(ave)	115.5	5.05	0
RETINEX	(gd-identity)	110.3	3.55	0
ECRULE-MV		55.9	3.68	0
ECRULE-MV	(opt)	62.6	2.46	242
ECRULE-MV	(ave)	55.9	3.68	1000
ECRULE-MV	(gd-identity)	63.3	2.48	0
ECRULE-ICA		49.3	3.04	0
ECRULE-ICA	(opt)	30.7	2.21	242
ECRULE-ICA	(ave)	49.3	3.04	1000
ECRULE-ICA	(gd-identity)	32.1	2.28	0

**Table 5.2**: Sharpening results for the Kodak DCS-200 digital camera. Assuming that the algorithms provide colour constancy estimates which are normally distributed around the target values, the uncertainty in the numbers here is roughly 2%. An asterisk is used for values which are not relevant or appropriate.

Algorithm	Sharpening method	RMS RGB difference between mapped image and target image	Angle between illumination RGB and estimate thereof	Count of times standard results replaced sharp result due to problems with negative components
BEST-LINEAR		2.0	*	0
BEST-DIAGONAL		6.0	0.43	0
BEST-DIAGONAL	(opt)	2.0	0.08	0
BEST-DIAGONAL	(ave)	6.4	0.81	0
BEST-DIAGONAL	(gd-identity)	2.9	0.16	0
ACTUAL		6.2	0.00	0
ACTUAL	(opt)	2.1	0.00	0
ACTUAL	(ave)	8.2	0.00	0
ACTUAL.	(gd-identity)	2.9	0.00	0
NOTHIING		*	7.14	0
GW		147.5	2.93	0
GW	(opt)	520.2	2.93	0
GW	(ave)	151.4	2.93	0
GW	(gd-identity)	150.1	2.93	0
DB-GW		41.6	2.52	0
DB-GW	(opt)	367.3	5.59	0
DB-GW	(ave)	44.3	2.43	0
DB-GW	(gd-identity)	42.6	2.49	0
RETINEX		121.6	4.15	0
RETINEX	(opt)	6245.1	17.64	0
RETINEX	(ave)	121.6	5.02	0
RETINEX	(gd-identity)	116.2	3.75	0
ECRULE-MV		57.9	3.84	0
ECRULE-MV	(opt)	65.2	2.62	278
ECRULE-MV	(ave)	53.7	4.19	0
ECRULE-MV	(gd-identity)	66.2	2.59	0
ECRULE-ICA		49.6	3.21	0
ECRULE-ICA	(opt)	32.5	2.33	278
ECRULE-ICA	(ave)	52.3	4.09	0
ECRULE-ICA	(gd-identity)	33.6	2.43	0

**Table 5.3**: Sharpening results for the Kodak DCS-420 digital camera. Assuming that the algorithms provide colour constancy estimates which are normally distributed around the target values, the uncertainty in the numbers here is roughly 2%. An asterisk is used for values which are not relevant or appropriate.



# **Figure 5.1**: The RMS RGB mapping error between the corrected image and the target image, by algorithm-sharpening method combination for the Sony DXC-930 camera. Note that the scale would have to be 30 times larger in order to fully illustrate the extent of all the bars.

RMS RGB mapping error as a function of algorithms and sharpening methods (Sony DXC-930)



#### RMS RGB mapping error as a function of algorithms and sharpening methods (Kodak DCS-200)

**Figure 5.2**: The RMS RGB mapping error between the corrected image and the target image, by algorithm-sharpening method combination for the Kodak DCS-200 camera. Note that the scale would have to be 10 times larger in order to fully illustrate the extent of all the bars.



RMS RGB mapping error as a function of algorithms and sharpening methods (Kodak DCS-420)

**Figure 5.3**: The RMS RGB mapping error between the corrected image and the target image, by algorithm-sharpening method combination for the Kodak DCS-420 camera. Note that the scale would have to be 20 times larger in order to fully illustrate the extent of all the bars.

### 5.8 Conclusion

We have investigated applying sensor sharpening to a variety of computational colour constancy algorithms. We found that doing so with existing methods leads to a number of problems, and as a result, using these methods is only attractive in a few specific circumstances. These difficulties lead us to propose a new sharpening method which is less ambitious in terms of theoretical gains, but addresses the needs of our preferred colour constancy algorithms. The resulting sharpening transforms substantially improve colour constancy in many cases, and only rarely had a small negative impact. Thus this work validates the original sharpening idea.

# **Chapter Six**

## Improvements to Gamut Mapping Colour Constancy Algorithms

The results in Chapter 4 suggest that the three dimensional gamut mapping algorithms are some of the most promising colour constancy methods available. In this chapter we propose two methods for further improving their efficacy. The reader will recall that these algorithms consist of two stages. First, the set of possible solutions is constrained. Then a solution is chosen from the resulting set. We propose improvements to each of these two stages. To improve the construction of the solution set, we suggest a method to reduce the error arising from diagonal model failure. This method is thus an alternative to the sensor sharpening paradigm discussed in the previous chapter. However, this method is more applicable to the extreme diagonal model failures inherent with fluorescent surfaces which are considered later.

To improve solution selection, we begin with an analysis of the two current approaches, namely averaging and the maximum volume heuristic. These methods are both attractive; the one which is preferred depends on the error measure, the number of surfaces, and other factors. Thus we propose a hybrid method which is easily adjustable to be more like the one method or the other. We found that it was relatively easy to find a degree of hybridization which improves gamut mapping colour constancy in the circumstances of most interest. We will now describe the two modifications in more detail, beginning with the method to reduce the reliance on the diagonal model.

### 6.1 Diminishing Diagonal Model Error

We will begin with a brief review of Forsyth's gamut mapping method [32]. First we form the set of all possible RGB due to surfaces in the world under a known, "canonical" illuminant. This set is convex and is represented by its convex hull. The set of all possible RGB under the unknown illuminant is similarly represented by its convex hull. Under the diagonal assumption of illumination change, these two hulls are a unique diagonal mapping (a simple 3D stretch) of each other.

Figure 6.1 illustrates the situation using triangles to represent the gamuts. In the full RGB version of the algorithm, the gamuts are actually three dimensional polytopes. The upper thicker triangle represents the



Figure 6.1: Illustration of fundamentals of gamut mapping colour constancy.

unknown gamut of the possible sensor responses under the unknown illuminant, and the lower thicker triangle represents the known gamut of sensor responses under the canonical illuminant. We seek the mapping between the sets, but since the one set is not known, we estimate it by the observed sensor responses, which form a subset, illustrated by the thinner triangle. Because the observed set is normally a proper subset, the mapping to the canonical is not unique, and Forsyth provides a method for effectively computing the set of possible diagonal maps. (See Chapters two and four or [8, 32, 37, 40, 61] for more details on gamut mapping algorithms).

We now consider the case where the diagonal model is less appropriate. Here it may be possible that an observed set of illuminants does not map into the canonical set with a single diagonal transform. This corresponds to an empty solution set. In earlier work we forced a solution by assuming that such null intersections were due to measurement error, and various error estimates were increased until a solution was found. However, this method does not give very good results in the case of extreme diagonal failures, such as those due to fluorescent surfaces.

To deal with this problem, we propose the following modification: Consider the gamut of possible RGB under a single test illuminant. Call this the test illuminant gamut. Now consider the diagonal map which takes the RGB for white under the test illuminant to the RGB for white under the canonical illuminant. If we apply that diagonal map to our test illuminant gamut, then we will get a convex set similar to the canonical gamut, the degree of difference reflecting the failure of the diagonal model. If we extend the canonical gamut to include this mapping of the test set, then there will always be a diagonal mapping from the observed RGB of scenes under the test illuminant to the canonical gamut. We repeat this procedure over a representative set of illuminants to produce a canonical gamut which is applicable to those illuminants as well as any convex combination of them. The basic idea is illustrated in Figure 6.2.



The gamuts of all possible RGB under three training illuminants.

**Figure 6.2**: Illustration of the modification to the gamut mapping method to reduce diagonal model failure.

### 6.2 Improving Solution Choice

Once a constraint set has been found, the second stage of the gamut mapping method is to select an appropriate solution from the constraint set. As discussed in detail in earlier chapters, two general methods have been used to do this. First, following Forsyth [32], we can select the mapping which maximizes the volume of the mapped set. Second, as proposed by Barnard [8], we can use the average of the possible maps. When Finlayson's illumination constraint is used, then the set of possible maps is non-convex. In [8], averaging was simplified by using the convex hull of the illuminant constraint. In [61] Monte Carlo integration was used in conjunction with the two dimensional version of the algorithm, and in Chapter 4 the average was estimated by numerical integration.

### IMPROVEMENTS TO GAMUT MAPPING ALGORITHMS

In the work presented in Chapter 4, we found that both averaging and the maximum volume method have appeal. We found that the preferred method was largely a function of the error measure, with other factors such as the diversity of scene surfaces also playing a role. When the scene RGB mapping error measure is used, the average of the possible maps is a very good choice. In fact, if we are otherwise completely ignorant about the map, then it is the best choice in terms of least squares.

On the other hand, if we use an illumination estimation measure, then the original maximum volume heuristic is often the best choice. This is important because we are frequently most interested in correcting for the mismatch between the chromaticity of the unknown illuminant and the canonical illuminant. In this case, the errors based on the chromaticity of the estimated scene illuminant correlate best with our goal, and the maximum volume heuristic tends to give the best results.

In this work we will focus on estimating the chromaticity of the illuminant. Despite the success of the maximum volume heuristic, we intuitively feel, that at least in some circumstances, some form of averaging should give a more robust estimate. This intuition is strengthened by the observation that when we go from synthetic to real data, the maximum volume method looses ground to averaging (see, for example, Figure 4.12).

To analyze the possibilities, we begin by considering solution selection by averaging. This averaging take place in the space of diagonal maps, which is not quite the same as the space of illuminants. Under the diagonal model, the illuminant RGB is proportional to the element-wise reciprocal of the diagonal maps. Thus we see that for an illumination oriented error measure, we may be averaging in the wrong space, as intuitively, we want to average possible illuminants.

However, averaging the possible illuminants has some difficulties. As we go towards the origin in the space of possible diagonal maps, the corresponding proposed illuminant becomes infinitely bright. The origin is included in the constraint set because we assume that surfaces can be arbitrarily dark. Although it is rare for a physical surface to have a reflectivity of less than 3%, surfaces can behave as though they are arbitrarily dark due to shading. Thus we always maintain the possibility that the illuminant is very bright. Specifically, if (R,G,B) is a possible illuminant colour, then (kR,kG,kB) is also a possible illuminant for all k>1. Put differently, a priori the set of RGB all possible illuminants is considered to be a cone in *illuminant* RGB space [111]. When we add the surface constraints, then the cone becomes truncated. As soon as we see anything but black, we know that the origin is excluded, and specific observed sensor responses lead to specific slices being taken out of the cone.

The above discussion underscores the idea that when we average illuminants, we should ignore magnitude. However, since the work presented in Chapter 4 demonstrates that the three dimensional algorithms outperform their chromaticity counterparts, we do not want to completely throw away the brightness information. Considering the truncated cone again, we posit that the nature of the truncations matter. The problem is then how to average the possible illuminants.

Consider, for a moment, the success of the three-dimensional gamut mapping algorithms. In the space of maps, each direction corresponds to a illuminant chromaticity. Loosely speaking, the chromaticity implied by an RGB solution, chosen in some manner, is the average of the possible chromaticities, weighted by an appropriate function. For example, the maximum volume estimate simply puts all the weight in the direction of the maximum coordinate product. Similarly, the average estimate weights the chromaticities by the volume of the cone in the corresponding direction.

Given this analogy, we can consider alternative methods of choosing a chromaticity solution. Since the maximum volume method tends to give better chromaticity estimates, especially when specularities are present, we wish to consider averages which put the bulk of the weight on solutions near the maximum volume direction. Now, one possible outcome of doing so would be the discovery that the maximum volume weighting worked the best. Interestingly, this proved not to be the case. Specifically we were able to find compromises which worked better.

We now present the weighting function developed for this work. Consider the solution set in mapping space. Then, each illuminant direction intersects the possible solution set at the origin, and at some other point. For an illuminant, i, let that other point be  $(d_r^{(i)}, d_g^{(i)}, d_b^{(i)})$ . Then, the functions we use to moderate the above weighting are powers of the geometric mean of coordinates of that mapping. Formally, we have parameterized functions  $f_N$ given by:

$$f_{N}(i) = \left(d_{r}^{(i)}d_{g}^{(i)}d_{b}^{(i)}\right)^{N_{3}}$$
(6.1)

We note that the solution provided by the average of the mappings is roughly  $f_3$ . The correspondence is not exact because the averaging is done over illuminant directions, not mapping directions. Similarly, as N becomes very large, the new method should approach the maximum volume method.

In order to use the above weighting function, we integrate numerically in polar coordinates. We discretize the polar coordinates of the illuminant directions inside a rectangular cone bounding the possible illuminant directions. We then test each illuminant direction as to whether it is a possible solution given the surface and illumination constraints. If it is, we compute the weighting function, and further multiply the result by the polar coordinate foreshortening,  $sin(\phi)$ . We sum the results over the possible directions, and divide the total by the total weight to obtain the weighted average.

### 6.3 Results

We first consider the results for the method introduced to deal with diagonal model failure. Since the efficacy of the diagonal model is known to be a function of the camera sensors [32, 35-38][Chapter 5], we provide results for two cameras with distinctly different degrees of support for the diagonal model. As discussed in Chapter 5, the Sony DXC-930 video camera has quite sharp sensors, and with this camera, the changes in sensor responses to illumination changes can normally be well approximated with the diagonal model. On the other hand, the Kodak DCS-200 digital camera has less sharp sensors, and the diagonal model is less appropriate.

In the first experiment, we generated synthetic scenes with 4, 8, 16, 32, 65, 128, 256, 512, and 1024 surfaces. For each number of surfaces, we generated 1000 scenes with the surfaces randomly selected from the reflectance database and a randomly selected illuminant from the test illuminant database. These databases are described in more detail in Chapter 4. For each algorithm and number of scenes we computed the RMS of the 1000 results, as also discussed in Chapter 4. Assuming normal statistics, we can estimate the relative error in the RMS estimate by  $1/\sqrt{2N}$  [110, p. 269] For N=1000, this is roughly 2%.

For each generated scene we computed the results of the various algorithms. We considered three-dimensional gamut mapping, with and without Finlayson's illumination constraint [40]. As in Chapter 4, we will label the versions without the illumination constraint by CRULE, which is adopted from [32]. When the illumination constraint is added, we use the label ECRULE instead (Extended-CRULE). Solution selection using the maximum volume heuristic is identified by the suffix MV. For averaging in the case of CRULE, we use the suffix AVE, and in the case of ECRULE, we use the suffix ICA, indicating that the average was over the non-convex set (Illumination-Constrained-Average). This gives a total of four algorithms: CRULE-MV, CRULE-AVE, ECRULE-MV, and ECRULE-ICA. Finally, the method described above to reduce diagonal model failure will be indicated by the prefix ND (Non-Diagonal). We test this method in conjunction with each of the four previous algorithms, for a total of eight algorithms. We report the distance in (r,g) chromaticity space between the scene illuminant and the estimate thereof.

In Figure 6.3 we show the results for the Sony DXC-930 video camera. We see that when solution selection is done by averaging (AVE and ICA), the ND algorithms work distinctly better than their standard counter-parts. On the other hand, when solutions are chosen by the maximum volume heuristic, the ND algorithms performed slightly worse than their standard counterparts, provided that the number of surfaces was not large. Interestingly, as the number of surfaces becomes large, the error in all the ND versions continues to drop to zero, whereas the error in the standard versions levels off well above zero. In Chapter 4 we postulated that this latter behavior was due to the limitations of the diagonal model, and the present results confirm this.

In Figure 6.4 we show the results for the Kodak DCS-200 digital camera. The sensors of this camera are not very sharp, and thus it is not surprising that the new extension significantly improves the performance of all four algorithms.

We now turn to results with generated data for the solution selection method developed above. For this experiment we included the ND extension to reduce the confound of diagonal model failure. We label the new method with the suffix SCWIA (Surface-Constrained-Weighted-Illuminant-Average) followed by the value of the parameter N in Equation (6.1). The results are shown in Figure 6.5. First we point out that solution selection by the original averaging method out-performs the maximum volume heuristic when the number of surfaces is small, but as the number of surfaces increases, the maximum volume heuristic quickly becomes the preferred method.

Turning to the new method, we see that it indeed offers a compromise between these two existing methods, with the new method tending towards the maximum volume method as N increases. More importantly, as long as N is 6 or more, the new method invariably outperforms solution selection by averaging. Furthermore, for N in the range of 9-24, the performance of the new method is better than the maximum volume heuristic, except when the number of surfaces is unusually large. When the number of surfaces becomes large, the maximum volume heuristic eventually wins out.

An important observation is that the results for N in the range of 9-24 are quite close, especially around 8 surfaces. This is fortuitous, as we have observed in Chapter 4 that 8 synthetic surfaces is roughly comparable in difficulty to our image data. Thus we are most interested in improving performance in the range of 4-16 surfaces, and we are encouraged that the results here are not overly sensitive to N, provided that it is roughly correct. Based on our results, N=12 appears to be a good compromise value for general purpose use.



**Figure 6.3**: Algorithm chromaticity performance versus the number of surfaces in generated scenes, showing the main gamut mapping algorithms and their non-diagonal counterparts. These results are for the Sony DXC-930 video camera which has relatively sharp sensors (the diagonal model is a good approximation in general). The error in the plotted values is roughly 2%.



**Figure 6.4**: Algorithm chromaticity performance versus the number of surfaces in generated scenes, showing the main gamut mapping algorithms and their non-diagonal counterparts. These results are for the Kodak DCS-200 digital camera which has relatively dull sensors (the diagonal model is not very accurate). The error in the plotted values is roughly 2%.



**Figure 6.5**: Algorithm chromaticity performance versus the number of surfaces in generated scenes, showing the selected gamut mapping algorithms, including ones with the new solution selection method. These results are for the Sony DXC-930 video camera. The error in the plotted values is roughly 2%.

Next we present some numerical results in the case of the Sony camera which shows the interactions of the two modifications. These are shown in Table 6.1. The main point illustrated in this table is that the slight disadvantage of the ND method, when used in conjunction with MV, does not carry over to the new solution selection method. To explain further, we note that the positive effect of reducing the diagonal model error can be undermined by the expansion of the canonical gamut, which represents an increase in the size of the feasible sets. The positive effect occurs because these sets are more appropriate, but, all things being equal, their larger size is an increase in ambiguity. Thus when the ND method is used in conjunction with a camera which supports the diagonal model, then, as the results here show, the method can lead to a decrease in performance. In our experiments on generated data, the negative effect is present in the case of MV, but in the case of averaging, the effect is always slightly positive. When ND is used in

Number of Surfaces	4	8	16
ECRULE-MV	0.064	0.044	0.032
ECRULE-ICA	0.058	0.050	0.044
ECRULE-SCWIA-3	0.057	0.051	0.045
ECRULE-SCWIA-6	0.054	0.043	0.036
ECRULE-SCWIA-9	0.054	0.041	0.032
ECRULE-SCWIA-12	0.055	0.040	0.031
ECRULE-SCWIA-18	0.057	0.040	0.030
ECRULE-SCWIA-24	0.058	0.041	0.029
ND-ECRULE-MV	0.065	0.047	0.033
ND-ECRULE-ICA	0.057	0.049	0.043
ND-ECRULE-SCWIA-3	0.060	0.054	0.048
ND-ECRULE-SCWIA-6	0.054	0.044	0.036
ND-ECRULE-SCWIA-9	0.054	0.041	0.031
ND-ECRULE-SCWIA-12	0.055	0.041	0.030
ND-ECRULE-SCWIA-18	0.057	0.041	0.029
ND-ECRULE-SCWIA-24	0.059	0.042	0.029

**Table 6.1**: Algorithm chromaticity performance for some of the algorithms developed here, together with the original methods, for generated scenes with 4, 8, and 16 surfaces. The numbers are the RMS value of 1000 measurements. The error in the values is roughly 2%.

conjunction with the new solution method, the results are also minimally compromised by this negative effect. This is very promising, because, in general, the diagonal model will be less appropriate, and the method will go from having little negative impact, to having a substantial positive effect. This has already been shown in the case of the Kodak DCS-200 camera, as well as when the number of surfaces is large. Increasing the number of surfaces does not, of course, reduce the efficacy of the diagonal model, but under these conditions, the diagonal model becomes a limiting factor.

Finally we turn to results with real image data. The images used for this experiment are the 321 carefully calibrated images described in detail in Chapter 4. We provide the results of some of the algorithms discussed above, as well as several comparison methods. For these we use the labeling of algorithms introduced in Chapter 4. We remind the reader that we use NOTHING to indicated the result of no colour constancy processing, and AVE-ILLUM for guessing that the illuminant is the average of a normalized illuminant database. The method labeled RETINEX estimates the illuminant RGB by the maximum found in each channel. GW estimates the illuminant based on the image average on the assumption that the average is the response to a perfect grey. DB-GW is similar, except that the average is now assumed to be the response to grey as defined by the average of a reflectance database. CIP-ICA is essentially a chromaticity version of ECRULE-ICA described in [111]. The method labeled NEURAL-NET is another chromaticity oriented algorithm which uses a neural net to estimate the illuminant chromaticity [63, 65, 66]. C-by-C-MAP is the Colour by Correlation method using the maximum posterior estimator [62]. Finally, C-by-C-MSE is Colour by Correlation using the minimum mean square error estimate. Again, all these comparison methods are described in detail in Chapter 4.

Table 6.2 shows the results over the 321 test images. The results from the image data generally confirm those from the generated data in the case of the new selection method. On the other hand, the ND method improves matters significantly in just one case, has essentially no effect in several others, and when used in conjunction with the new selection method, it has a small negative effect. Since the camera used already supports the diagonal model well, these varied results are understandable.

Algorithm	Solution Selection Method (If Applicable)					
	MV	AVE/ICA	SCWIA-6	SCWIA-9	SCWIA-12	SCWIA-15
CRULE	0.045	0.046				
ECRULE	0.041	0.047	0.043	0.039	0.037	0.037
ND-CRULE	0.047	0.039				
ND-ECRULE	0.042	0.048	0.045	0.041	0.040	0.040
NOTHING		0.125				
AVE-ILLUM		0.094				
GW	0.106					
DB-GW	0.088					
RETINEX	0.062					
CIP-ICA	0.081					
NEURAL-NET	0.069					
C-by-C-MAP	0.072					
C-by-C-MMSE	0.070					

**Table 6.2**: The image data results of the new algorithms compared to related algorithms. The numbers presented here are the RMS value of the results for 321 images. Assuming normal statistics, the error in these numbers is roughly 4%.

### 6.5 Conclusion

We have described two improvements to gamut mapping colour constancy. These improvements are important because earlier work has shown that this approach is already one of the most promising. For the first improvement we modified the canonical gamuts used by these algorithms to account for expected failures of the diagonal model. When used with a camera which does not support the diagonal model very well, the new method was clearly superior. When used with a camera with sharp sensors, the resulting method improved gamut mapping algorithms when the solution was chosen by averaging. When the maximum volume heuristic was used, there was a slight decrease in performance. This decrease was erased when the method was combined with the second improvement. Furthermore, we posit that any decreases in performance must be balanced against the increased stability of the new method as the number of surfaces becomes large.

We are also encouraged by the results of the new method for choosing the solution. Our findings contribute to the understanding of the relative behavior of the two existing methods. Furthermore, the flexibility of the new method allows us to select a variant which works better than either of the two existing methods for the kind input we are most interested in.

# **Chapter Seven**

## **Colour Constancy with Fluorescent Surfaces**

Fluorescent surfaces are common in the modern world, but they present problems for machine colour constancy because fluorescent reflection typically violates the assumptions needed by most algorithms. The complexity of fluorescent reflection is likely one of the reasons why fluorescent surfaces have escaped the attention of computational colour constancy researchers. In this chapter we take some initial steps to rectify this omission. We begin by introducing a simple method for characterizing fluorescent surfaces. It is based on direct measurements, and thus has low error and avoids the need to develop a comprehensive and accurate physical model. We then modify and extend several modern colour constancy algorithms to address fluorescence. The algorithms considered are CRULE and derivatives [8, 32, 40, 61], Colour by Correlation [62], and neural net methods [63, 65, 66]. Adding fluorescence to Colour by Correlation and neural net methods is relatively straight forward, but CRULE requires the use of the ND extension introduced in the previous chapter. We present results for both synthetic and real image data for fluorescent capable variants of CRULE and

Colour by Correlation, and we compare the results with the standard versions of these and other algorithms.

### 7.1 Characterizing Fluorescent Surfaces

We begin by introducing a simple method for characterizing fluorescent surfaces. It is based on direct measurements, and thus has low error and avoids the need to develop, fit, and test physical models. Such models are necessarily quite complex and limited to the kinds of surfaces exhibiting the processes being modeled (an elegant model for one case is developed in [113]). We remind the reader that the key characteristic of fluorescent surfaces is that some of the light energy they absorb is re-emitted at longer wavelengths (lower energy). If we represent the incident light spectra as a vector of samples over wavelength , then reflectance can be described by the multiplication of that input vector by a triangular matrix. This is much more complex than the non-fluorescent case where a diagonal matrix is sufficient. Although it is possible to measure this matrix, doing this effectively requires equipment which is not readily available. Thus we introduce a more direct method for obtaining the data required.

Given a fluorescent surface candidate, we measure the spectra of the reflected light under a number of illuminants using a Photoresearch PR-650 spectraradiometer. We also measure the spectra of the illuminants providing the input energy to the fluorescent surface. Then, to simulate the surface under a new illuminant spectra, we first compute the positive linear combination of the test illuminants which is closest to the new illuminant spectra using constrained least squares optimization. The reflected energy of the fluorescent spectra under the new illuminant is then approximately that same linear combination applied to the measured test response spectra set. A simple example should make this clear. Assume that when the fluorescent surface is illuminated by a spectra A, the result is spectra A', and similarly, let B' be the response to stimulus B. Then if a illuminant C is roughly A+2B, then the response, C', is roughly A'+2B'. This procedure is used to simulate

fluorescent reflection to obtain the data sets required by colour constancy algorithms.

### 7.2 Colour Constancy with Fluorescent Surfaces

We now turn to the algorithms themselves. We feel that the most effective computational colour constancy methods currently available are CRULE and derivatives [8, 32, 40, 61], Colour by Correlation [62], and neural net methods [63, 65, 66]. Adding fluorescence to Colour by Correlation and neural net methods is relatively straight forward. In either case, our characterization of fluorescent surfaces is used to augment the world used for training (neural nets) or building correlation matrices (Colour by Correlation).

Extending Forsyth's CRULE method follows from the ND extension introduced in the previous chapter. In fact, the desire to deal with fluorescent surfaces motivated this extension. With this extension we model the gamuts for each illuminant, and use the result to adjust the canonical gamut. Given the fluorescent surface characterization developed above, we can easily include the effect of having fluorescent surfaces in our database of reflectances used to generate these gamuts.

### 7.3 Experimental Results

As an initial step in our investigation of fluorescence we measured a number of candidate surfaces, and trimmed this set down to 9 strongly fluorescent ones. These included 3 printed surfaces from a laundry detergent box, 2 surfaces from a multi-coloured child's cloth ball, 2 different colours of flagging tape, and 2 different vividly coloured pieces of paper. As described above, our method of characterizing the fluorescent surfaces required measuring their reflectance spectra under a number of representative illuminants. For non-fluorescent spectra we used the set of roughly 2000 spectra collected from several sources, and described more fully in Chapter 4. The illuminant data sets used for algorithm calibration (training) and generated data experiments were also the same as in Chapter 4. Finally, we note that the camera used for the experiments was the same as used in previous chapters, and calibrated as described in Chapter 3.

For the purposes of this study, we will assume that the goal of the algorithms is to estimate the response of the vision system to a perfect white patch. However, it is often the case that we are most interested in the chromaticity of the illuminant, and several of the algorithms of interest only compute the illuminant chromaticity. Hence, we only report chromaticity results. The specific error metric used is the distance in (r,g) chromaticity space between the illuminant chromaticity and the estimate thereof.

We present the results using the modified algorithms, as well as a number of comparison algorithms. We label the algorithms as introduced in Chapters 4 and 6. The new algorithms considered in this chapter are fluorescent capable versions of three-dimensional gamut-mapping, and Colour by Correlation. We use the prefix FL to denote algorithms which have been extended to deal with fluorescent surfaces. A key to the labeling of all the algorithms studied in this chapter is provided in Table 7.1.

We first present some results using generated data. The use of generated data eliminates calibration problems, and simplifies analysis of the effects of statistical assumptions. We present the results of two experiments on generated data. We will first describe both experiments and then discuss the results.

Our first experiment looked at the performance of the algorithms developed to deal with fluorescent surfaces and their standard counter-parts. We generated synthetic scenes with 4, 8, 16, 32, 65, 128, 256, 512, and 1024 surfaces. We arranged for 30% of the randomly selected surfaces to be fluorescent. For each number of surfaces, we generated 1000 scenes with the surfaces randomly selected from the reflectance database and a randomly selected illuminant from the test illuminant database. For each algorithm and number of scenes we computed the RMS of the 1000 results, as also discussed in Chapter 4. Assuming normal statistics, we can estimate the relative error

in the RMS estimate by  $1/\sqrt{2N}$  [110, p. 269] For N=1000, this is roughly 2%. The results of the first experiment are plotted in Figure 7.1.

In our second experiment with generated data, we look more closely at the performance of our algorithms using synthetic scenes with 4 and 8 surfaces, again using 1000 of each. In Chapter 4 we observed that scenes with 4-16 synthetic surfaces are most similar in difficulty to the image data we are most interested in. For this experiment we considered the performance of the algorithms both with and without fluorescent surfaces. This is important because we want to know the degree to which modifications for fluorescence degrades the performance of the algorithms when no such surfaces are present. The results of this experiment are shown in Table 7.2.

ECRULE	CRULE with illumination constraint
MV	Solutions are chosen by max volume heuristic
ICA	Solutions are the average over a non-convex feasible set.
SCWIA	Solutions are the average over feasible illuminant chromaticities, weighted by a function chosen to emphasize illuminants with chromaticities around the MV solution, as described in Chapter 6.
ND	Gamut mapping algorithm is extended to reduce diagonal model failure
FL	Algorithm is extended for fluorescence.
RETINEX	Estimate illuminant by the max RGB in each channel.
GW	Estimate illuminant colour by assuming that image average is the colour of a 50% reflectance
DB-GW	Estimate illuminant colour by assuming that image average is the colour of the average of a reflectance database.
C-by-C-MAP	Colour by Correlation [62], with a Gaussian mask to smooth the correlation matrix and maximum likelihood estimate. For the FL-C-by-C variant, an abundance of fluorescent surfaces are included in the construction of the correlation matrix
C-by-C- MMSE	Colour by Correlation [62], with a Gaussian mask to smooth the correlation matrix and mean likelihood estimate. For the FL-C-by-C variant, an abundance of fluorescent surfaces are included in the construction of the correlation matrix
Neural Net	Neural net trained to estimate illuminant chromaticity based on the observed image colours [63, 65, 66]. In this work, the neural net was <i>not</i> trained to deal with fluorescent surfaces.

**Table 7.1**: Key to the algorithms studied in this chapter.

#### COLOUR CONSTANCY WITH FLUORESCENT SURFACES

The first conclusion from these two experiments is that the presence of fluorescent surfaces does, as predicted, degrade every algorithm not designed to deal with them. Interestingly, when fluorescent surfaces are present, the ECRULE-MV algorithm loses ground to several other algorithms. This algorithm has very good performance when fluorescent surfaces are not present, providing better illumination chromaticity estimation than the ECRULE-ICA algorithm. However, when fluorescent surfaces are present, both Figure 7.1 and Table 7.2 indicate that ECRULE-ICA is the better choice. Thus it appears that while the maximum volume choice may be very good under ideal conditions, it is less robust than using some form of averaging.

We now consider the methods introduced in this chapter. In general we see that the proposed modifications for fluorescent surfaces produced significant improvements when such surfaces were present. The most impressive gain was with the extended Colour by Correlation method, but this is likely due in part to an unnatural advantage that it does not enjoy in the case of real image data. Specifically, the second Colour by Correlation algorithm was trained on data statistically similar to the test data. We remind the reader that the DB-GW algorithm has a similar advantage on the synthetic data in the case where there are no fluorescent surfaces present.

Equally promising is the performance of the modified algorithms when no fluorescent surfaces were present. Naturally we cannot expect the modified algorithms to be optimal under these conditions. However, we are encouraged that the penalty was generally of the order of 10% or less. For example, with 8 surfaces, the FL-ECRULE-SCWIA-12 gives the best result when there are fluorescent surfaces ((r,g) error is 0.049). When there are no such surfaces present, the error is 0.045. This compares favorably with the error with this algorithms standard counterpart, ECRULE-SCWIA-12, which gives an error of 0.040. For many applications this could well be an acceptable penalty for the significant increase in robustness.



**Figure 7.1**: The performance of the algorithms developed for fluorescence surfaces and their standard counterparts with generated scenes with 30% fluorescence surfaces .

Algorithm	Generated scenes v	s with 4 surfaces Generated scenes with 8 su		vith 8 surfaces
	No fluorescence	With 30% fluorescence	No fluorescence	With 30% fluorescence
NOTHING	0.114	0.114	0.114	0.114
AVE-ILLUM	0.086	0.086	0.086	0.086
GW	0.076	0.170	0.058	0.144
DB-GW	0.069	0.146	0.048	0.117
RETINEX	0.093	0.164	0.067	0.116
NEURAL-NET	0.052	0.062	0.039	0.050
C-by-C-MAP	0.060	0.083	0.042	0.075
C-by-C-MMSE	0.049	0.082	0.034	0.074
FL-C-by-C-MAP	0.064	0.049	0.045	0.030
FL-C-by-C-MMSE	0.051	0.039	0.037	0.025
CRULE-MV	0.076	0.133	0.051	0.104
CRULE-AVE	0.079	0.112	0.060	0.085
ECRULE-MV	0.064	0.089	0.044	0.081
ECRULE-ICA	0.059	0.069	0.051	0.065
ECRULE-SCWIA-12	0.055	0.077	0.040	0.069
ND-CRULE-MV	0.083	0.135	0.055	0.105
ND-CRULE-AVE	0.075	0.111	0.054	0.079
ND-ECRULE-MV	0.065	0.094	0.048	0.084
ND-ECRULE-ICA	0.058	0.068	0.050	0.063
ND-ECRULE-SCWIA-12	0.055	0.080	0.041	0.072
FL-CRULE-MV	0.080	0.123	0.055	0.076
FL-CRULE-AVE	0.094	0.096	0.078	0.061
FL-ECRULE-MV	0.065	0.086	0.048	0.064
FL-ECRULE-ICA	0.064	0.060	0.057	0.050
FL-ECRULE-SCWIA-12	0.057	0.064	0.045	0.049

**Table 7.2**: Algorithm chromaticity performance on generated scenes with 4 and 8 surfaces, with and without fluorescent surfaces.

#### COLOUR CONSTANCY WITH FLUORESCENT SURFACES

We have also tested the algorithms under study on real image data. We constructed 6 scenes which all included at least one known or suspected fluorescent surface. We took images of these scenes under 11 different illuminants, resulting in 66 images. Seven images were culled due to problems with the experiment, leaving a total of 59 input images. The illuminant data set is the same as the one used in Chapter 4. Figure 7.2 shows the 6 scenes under the canonical illuminant. Figure 7.3 shows one of the scenes under all the illuminants. We also look at the performance of the algorithms when fluorescent surfaces are absent. Here we present the results for the 321 input images used in Chapter 4. These images are from 33 scenes which are relatively free of fluorescent surfaces.

The dynamic range of all images in both data sets was extended using reduced illumination levels and averaging multiple frames. This gives us the opportunity to explore colour constancy in the context of a high dynamic range vision system, as well as more standard vision systems, which can be simulated by truncating the higher range data. The effect on the results is to give the Retinex based algorithm, and the maximum volume algorithms, an advantage. This is especially true when there are specularities present (Chapter 4, [107]).

In general, results from this real image data demonstrate that modeling fluorescence is again beneficial, although the large improvement in the case of Colour by Correlation has been reduced to be more comparable to that of the gamut mapping algorithms. This is likely due in part to the mismatch between the statistics used for training and the somewhat arbitrary statistics in the image data. We also note that Colour by Correlation has many possible implementations, and we are still working on finding a robust set of parameters for that algorithm. In the case of the gamut mapping algorithms, we see that the performance on the real image data is excellent, with the FL-ECRULE-SCWIA-12 algorithm providing the best results for the data set with fluorescent surfaces, and very nearly matching the best algorithm in for the data set without fluorescent surfaces.

## **INSERT COLOUR PLATE HERE**

Figure 7.2: The six scenes with fluorescent surfaces used for image data .

## **INSERT COLOUR PLATE HERE**

Figure 7.3: One of the scenes with fluorescent surfaces under all 11 illuminants .

Algorithm	Scenes without fluorescent surfaces	Scenes with fluorescent surfaces
NOTHING	0.125	0.115
AVE-ILLUM	0.094	0.088
GW	0.106	0.161
DB-GW	0.088	0.135
RETINEX	0.062	0.110
NEURAL-NET	0.069	0.065
C-by-C-MAP	0.072	0.082
C-by-C-MMSE	0.070	0.081
FL-C-by-C-MAP	0.068	0.069
FL-C-by-C-MMSE	0.068	0.067
CRULE-MV	0.045	0.109
CRULE-AVE	0.046	0.091
ECRULE-MV	0.041	0.075
ECRULE-ICA	0.047	0.060
ECRULE-SCWIA-12	0.037	0.064
ND-CRULE-MV	0.047	0.102
ND-CRULE-AVE	0.039	0.084
ND-ECRULE-MV	0.042	0.080
ND-ECRULE-ICA	0.048	0.059
ND-ECRULE-SCWIA-12	0.040	0.066
FL-CRULE-MV	0.043	0.094
FL-CRULE-AVE	0.062	0.067
FL-ECRULE-MV	0.039	0.073
FL-ECRULE-ICA	0.052	0.053
FL-ECRULE-SCWIA-12	0.038	0.050

**Table 7.3**: Algorithm chromaticity performance on image data for two sets of scenes. The first is a data set of 321 images with no known significantly fluorescent surfaces. The second is the set of 59 images taken for this work. The images in this latter set have at least one fluorescent surface.

### 7.4 Conclusions

We have shown how to modify the three leading machine colour constancy methods to deal with fluorescent surfaces. Dealing with such surfaces has been ignored until now, but we argue that doing so is important, as such surfaces are common in the modern world, and yet they dramatically degrade the performance of existing algorithms. Although further work is needed to estimate the frequency of occurrence of such surfaces, we pass on to the reader the following anecdotal datum. Our interest in exploring fluorescent surfaces arose because such surfaces were present in 20% of the randomly constructed scenes used to provide preliminary data for research into colour constancy performance. Clearly we had to deal with fluorescence before we could proceed towards our goal of having colour constancy algorithms for real world applications.
# **Chapter Eight**

# **Colour Constancy with Specular and Non-Specular Surfaces**

There is a growing trend in machine colour constancy research to use only image chromaticity information, ignoring the magnitude of the image pixels. This is natural because the main purpose is often to estimate only the chromaticity of the illuminant. However, as we have seen in the previous chapters, the magnitudes of the image pixels also carry information about the chromaticity of the illuminant. One such source of information, discussed in Chapter 4 is through image specularities. As is well known in the computational colour constancy field, specularities from inhomogeneous materials (such as plastics and painted surfaces) can be used for colour constancy. This assumes that the image contains specularities, that they can be identified, and that they do not saturate the camera sensors. These provisos make it important that colour constancy algorithms which make use of specularities also perform well when the they are absent. A further problem with using specularities is that the key assumption, namely that the specular component is the colour of the illuminant, does not hold in the case of coloured metals.

### COLOUR CONSTANCY WITH SPECULAR AND NON-SPECULAR SURFACES 165

In this chapter we investigate a number of colour constancy algorithms in the context of specular and non-specular reflection. We then propose extensions to several variants of the three-dimensional gamut mapping algorithms which make use of specularities if they exist, but do not rely on their presence. In addition, our approach is easily extended to include coloured metals, and is the first colour constancy algorithm to deal with such surfaces (but see [15, 19, 76, 114, 115] for related work). Finally, our method provides an estimate of the overall brightness, which chromaticity-based methods cannot do, and other RGB based algorithms do poorly when specularities are present.

### 8.1 Introduction

The use of specularities for machine colour constancy has its origin in the dichromatic model of reflectance [18, 24]. This model separates the light reflected from inhomogeneous materials such as plastics and paints into a diffuse (body) component, and a specular (interface) component. The body reflection blends the spectral reflectance properties of the object with that of the illumination, whereas the specular component has the same spectral makeup as the illuminant. Reflections from different parts of the same surface have varying amounts of the two reflection components due to changes in geometry, and various researchers have used this property to estimate the illuminant colour [18, 20, 69, 70, 72, 73]. Alternatively, since the maximal specular reflection is typically much larger than the body reflection, a bright specularity can be a good estimate of the illuminant colour as is, if it can be identified as a specularity. Either way, using specular reflection for colour constancy typically requires an implicit physical segmentation of the image pixels, and the difficulties in doing this have, in part, inspired the present work.

In this chapter we will assume that the goal of the algorithms is to estimate the response of the vision system to a perfect white patch. This response will loosely be referred to as the colour of the illuminant. It is most natural for that response to be the same dimension as the number of sensors in the vision system, and thus, for a standard colour camera, the response would be the (R,G,B) of a white patch under that illuminant. However, as discussed in previous chapters, it is often the case that we are most interested in the chromaticity of the illuminant, and an estimate of that chromaticity will suffice. This being the case, a number of colour constancy algorithms have been developed which work entirely in some chromaticity space [40, 61-63, 65, 66], and much progress has been made by taking advantage of the simplifications afforded by this strategy.

Nonetheless, if we now consider the case where specularities are present, we observed in Chapter 4 that certain RGB based algorithms, such as the original CRULE algorithm, estimate the illuminant chromaticity surprisingly well—even though they were not designed to optimize chromaticity estimation. The success of these algorithms when specularities are present is limited by the dynamic range of the vision system. We expect more dynamic range to become available to machine vision systems (see [102] for information about one high dynamic range camera), but currently, specularities tend to be clipped, and such pixels must be excluded as unreliable. As clipping becomes severe, these methods degrade, especially Retinex [107]. We also note that using these algorithms for illumination brightness estimation fails when strong specularities are present.

Chromaticity-based approaches, on the other hand, cannot use specular information on a pixel by pixel basis, and cannot provide illuminant brightness estimation. However, as noted above, we are often most interested in illuminant chromaticity estimation, and these approaches tend to be robust with respect to specularities. This is because specularities in chromaticity space simply desaturate colours, leading to colours which are perhaps less useful to the algorithm, but are nonetheless plausible [40], and thus the degradation is graceful. The essence of this observation also applies in the case of coloured metals.

In contrast to the above algorithms (and others), which we analyze post hoc with respect to their abilities to ignore or take advantage of specularities, several researchers have developed computational colour constancy methods which explicitly use and rely on specularities [18, 20, 69, 70, 72, 73]. In favorable situations, these methods can work well, but strong specularities are not always present, and as noted above, are often clipped. Furthermore, specularities from coloured metals are not the same colour as the illuminant, and these methods do not address this. These considerations lead us to propose extensions to several of the variants of Forsyth's CRULE method which take advantage of specularities if they exist, but continue to be strong algorithms if there are no specularities present.

## 8.2 Extending CRULE for specularities

We will now describe the extension to the three-dimensional gamut method to make use of specularities. The three dimensional gamut mapping approach is described in detail in Chapters 2 and 4, and we ask the reader to refer to the descriptions in §2.3.4 or §4.5.3 if a review is necessary.

In order to extend the three-dimensional gamut approach for specularities, we model specular reflection and modify the canonical gamuts appropriately. The canonical gamuts are polytopes in RGB space, having roughly the shape of two multi-faceted pyramids which are joined together at their identical bases. We normally include the origin as one of the vertices (and thus it is the apex of one of the pyramids), because, a priori, the observed RGB could all be due to surfaces which are arbitrarily dark as a result of being obliquely illuminated. At the other extreme (the apex of the other pyramid) there is a vertex corresponding to the whitest reflectance. To include specularities we take that vertex, and move it away from the origin, along the line connecting to the origin. Thus the hull facets adjacent to the origin remain the same, but the ones adjacent to the RGB of white are stretched away from the origin. In other words, we add a single reflectance to our world which is a multiple of a uniform reflectance. The multiple should be large enough to accommodate a bright specularity taking the dynamic range of the vision system into account, but the exact specification of the value is not very important. We have experimented with factors of 2, 4, and 8, with 8 being used for the results. The concept is illustrated using two dimensions in Figure 8.1.

### COLOUR CONSTANCY WITH SPECULAR AND NON-SPECULAR SURFACES 168

While very simple, the method naturally models real specularities which are always a combination of the specular reflection and the underlying body reflection. Both the specular reflection and the body reflection are part of the convex hull, and thus any convex combination of them is also in the hull. Finally, to include the specular reflection of coloured metals (brass, copper, gold), we add multiples of the reflections for these substances into the canonical gamut. The colour of specularities is still quite restricted, being somewhere between white and the colour of copper, but the existence of metallic specularities will now work with, instead of against, the information provided by the other colours.

The new canonical gamut is then used as part of standard RGB based gamut-mapping algorithms. For example, we can include Finlayson's illumination constraint [40], and the extension for dealing with diagonal model failure introduced in Chapter 6. Similarly, all of the methods for



Projection of the canonical gamut. The broken line shows the inclusion of specularities. The dotted line shows the gamut used by chromaticity methods. Here the gamuts are cones in RGB space.

**Figure 8.1**. Illustration of gamut extension used for specularities. The gamuts are actually polytopes in 3 dimensional RGB space.

choosing a solution from the constraint set can be used. We remind the reader that the main ones are the original maximum volume method, averaging over the feasible set, and the hybrid method introduced in Chapter 6.

This method works well even if there are no specularities. The work of Finlayson and Hordley [61] suggests that the most important facets in the nonspecular case are the ones adjacent to the origin; specifically the ones not modified by our method. The arguments in that work also imply that our method should be at least as strong as any chromaticity-based gamut-mapping algorithm, regardless of the presence of specularities. Of course, when specularities are present, our algorithm should excel. Finally, when there are strong un-clipped specularities, our algorithm estimates the overall illuminant brightness better than all other algorithms.

### 8.3 Experiments with Generated Data

We have tested the above methods both on generated data, and on image data. For the former, we generated data without specularities, with nonmetallic specularities, and with metallic specularities. To model the metallic specularities we measured the specular reflectance of a number of metallic objects using a Photoresearch PR-650 spectraradiometer. The metallic samples included several brass and copper surfaces, as well as grey metallic surfaces such as aluminum and stainless steel. We modeled non-specular reflectance using the database of roughly 2000 reflectance spectra used in Chapter 4 and described more full there. For each simulated "world" we ran all the algorithms on 1000 randomly selected groups of 8 surfaces under randomly selected illuminants. The test set of illuminants was the same as that used for synthetic data in Chapter 4. For each set of generated data with metallic and non-metallic specularities we also simulated pixel clipping, for a total of five algorithm test conditions.

The degree of specularity modeled deserves further comment. As mentioned above, we provide results using canonical hulls which model specularities which are up to 8 times as bright as a perfect white patch. Naturally, the results will be best if the generated test data conforms to this model. However, we do not feel that this is a fair test, as this information is not generally known. Furthermore, the basic idea is to extend the hull sufficiently so that it models any expected specularities. Thus we choose to limit the magnitude of specularities added to the generated surfaces to be significantly less than that which was modeled by the extended canonical hulls. In the case of the standard, non-metallic specularities, the maximum added specularity was twice the brightness of a perfect white patch. Since, metallic specularities tend to be brighter than those from dielectrics, we added metallic specularities which were up to a factor of four times as bright as a perfect white. The amount of specularity actually added was arranged to be uniformly distributed between zero and the maximum value. The specularities were added to generated surfaces with a probability of 25 percent.

We provide the results of the algorithms using two different error measures. The first measures the ability of the algorithms to estimate the chromaticity of the illuminant. Here we use the distance in (r,g) space between the chromaticity of the illuminant and the estimate thereof. We also look at the ability of the algorithms to estimate the brightness of the illuminant. Here we report the RMS error in the difference between the actual illuminant R+G+B, and the estimate thereof. Investigating the brightness performance is important because we know that specularities tend to degrade this performance, and our proposed modifications are motivated, in part, by the need to deal with this problem.

In Figure 8.2 we plot the chromaticity results using the new extensions, in conjunction with the gamut-mapping modifications developed in Chapter 6. The performance of a number of possible other variants and some comparison algorithms is tabulated in Table 8.2 (a key to the algorithms is provided in Table 8.1). The results generally confirm that the modifications to gamut-mapping have the intended effect. Modeling standard specularities improve the results when they were present, and likewise with metallic specularities. We note, however, that the improvement in chromaticity performance when standard specularities are modeled is small when used in conjunction with the MV or SCWIA solution selection method. This is consistent with the observation that, when these solution selection methods are used, three-dimensional gamut-mapping algorithms already exploit specularities. By contrast, when we compare the results with the ICA solution selection method, then we see that modeling the specularities leads to a significant improvement (an error of 0.33 for SP-ECRULE-ICA compared with 0.46 for ECRULE-ICA).

With metallic specularities, the improvement using the extension for metallic specularities is substantial. Unfortunately, this modification also significantly degrades chromaticity performance when such specularities are not present. Thus further work is required to make this modification more robust, and identify when the overall effect will be beneficial.

CRULE	Gamut mapping solution selection using the original constraints based on the observed pixels values as described in [32].
ECRULE	CRULE with the illumination constraint described in [40].
MV	Solutions are chosen by the maximum volume heuristic introduced in [32].
ICA	Solutions are the average over a non-convex feasible set.
SCWIA	Solutions are the average over feasible illuminant chromaticities, weighted by a function chosen to emphasize illuminants with chromaticities around the MV solution, as described in Chapter 6.
ND	Gamut mapping algorithm is extended to reduce diagonal model failure as described in Chapter 6.
SP	Gamut mapping algorithm is extended for standard specularities as described in this chapter.
MET	Gamut mapping algorithm is extended for metallic specularities as described in this chapter.
RETINEX	Estimate illuminant by the max RGB in each channel.
GW	Estimate illuminant RGB by assuming that the image average is the colour of a 50% reflectance
DB-GW	Estimate illuminant RGB by assuming that the image average is the colour of the average of a reflectance database.
C-by-C-MAP	Colour by Correlation [62] with maximum likelihood estimate.
C-by-C-MMSE	Colour by Correlation [62], with mean likelihood estimate.
Neural Net	Neural net trained to estimate illuminant chromaticity based on the observed image colours [63, 65, 66].

Table 8.1: Key to the algorithms studied in this chapter.



**Figure 8.2**: Chromaticity performance of some of the algorithms developed here under 5 different test conditions. More detailed results are available in Table 8.2.

	No specularities	Simulated specularities	Simulated specularities and simulated clipping	Simulated metallic specularities	Simulated metallic specularities and clipping
NOTHING	0.114	0.114	0.114	0.114	0.114
AVE-ILLUM	0.086	0.086	0.086	0.086	0.086
GW	0.058	0.035	0.054	0.071	0.069
DB-GW	0.048	0.033	0.046	0.047	0.053
RETINEX	0.067	0.033	0.059	0.090	0.091
CIP-ICA	0.077	0.078	0.078	0.078	0.078
NEURAL-NET	0.038	0.031	0.038	0.040	0.043
C-by-C-01	0.078	0.079	0.079	0.079	0.079
C-by-C-MAP	0.042	0.031	0.040	0.042	0.046
C-by-C-MMSE	0.034	0.026	0.033	0.036	0.039
CRULE-MV	0.052	0.029	0.047	0.078	0.075
CRULE-AVE	0.061	0.053	0.060	0.054	0.063
ECRULE-MV	0.046	0.027	0.042	0.062	0.067
ECRULE-ICA	0.051	0.046	0.050	0.051	0.057
ECRULE-SCWIA-12	0.041	0.029	0.039	0.053	0.058
ND-ECRULE-MV	0.047	0.027	0.043	0.065	0.069
ND-ECRULE-ICA	0.049	0.045	0.048	0.053	0.058
ND-ECRULE-SCWIA-12	0.041	0.028	0.038	0.056	0.060
SP-ECRULE-MV	0.052	0.032	0.048	0.058	0.060
SP-ECRULE-ICA	0.050	0.033	0.046	0.047	0.055
SP-ECRULE-SCWIA-12	0.046	0.027	0.042	0.052	0.055
MET-ECRULE-MV	0.070	0.065	0.072	0.050	0.067
MET-ECRULE-ICA	0.062	0.066	0.065	0.044	0.059
MET-ECRULE-SCWIA-12	0.063	0.062	0.065	0.044	0.060
SP-ND-ECRULE-MV	0.053	0.032	0.049	0.060	0.062
SP-ND-ECRULE-ICA	0.046	0.029	0.042	0.050	0.055
SP-ND-ECRULE-SCWIA-12	0.044	0.025	0.040	0.055	0.056
MET-ND-ECRULE-MV	0.067	0.060	0.069	0.047	0.065
MET-ND-ECRULE-ICA	0.057	0.060	0.059	0.040	0.055
MET-ND-ECRULE-SCWIA-12	0.058	0.056	0.060	0.040	0.056

**Table 8.2**: RMS error in (r,g) chromaticity estimates for 1000 generated scenes with 8 surfaces.

### COLOUR CONSTANCY WITH SPECULAR AND NON-SPECULAR SURFACES 174

We now turn to illuminant brightness estimation performance. A selection of results are shown in Figure 8.4, and more detailed results are provided in Table 8.3. Here it is clear that specularities degrade the performance of standard algorithms, and that the proposed modifications mitigate this degradation. The performance of the new methods is consistently good across the 5 different test conditions. We remind the reader that the maximal degree of specularity used for the generated surfaces was different than that modeled by the methods. We also note that the performance of the standard algorithms is notably worse in the metallic specularity case as compared to the standard specularity case largely because of the experimental design. Specifically, we modeled specularities up to twice the brightness of white in the first case, and four times in the second case. Finally, we note that the number of algorithms in Table 8.3 is substantially less than that for Table 8.2 simply because the balance of the algorithms only compute illuminant chromaticity.

Algorithm	No specularities	Simulated specularities	Simulated specularities and simulated clipping	Simulated metallic specularities	Simulated metallic specularities and clipping
GW	310	217	264	274	268
DB-GW	145	731	258	893	263
RETINEX	267	428	230	651	241
CRULE-MV	178	577	182	855	205
CRULE-AVE	303	1632	554	2139	595
ECRULE-MV	172	579	178	924	202
ECRULE-ICA	218	1331	422	1947	482
SP-ECRULE-MV	264	232	274	302	259
SP-ECRULE-ICA	163	244	173	867	230
MET-ECRULE-MV	312	314	336	294	333
MET-ECRULE-ICA	185	211	213	193	211

Table 8.3: RMS error in illuminant R+G+B estimates for 1000 generated scenes with 8 surfaces.



**Figure 8.3**: Illuminant magnitude performance of some of the algorithms developed here under 5 different test conditions. More detailed results are available in Table 8.3.

### 8.4 Experiments with Image Data

We also provide some results on images from two data sets. The first data set is the one introduced in Chapter 4. We remind the reader that it consists of scenes with and without significant specularities, but with few metallic specularities. In this data set there are 33 scenes taken under 11 different illuminants. Several images were culled due to problems, leaving 321 test images. To keep the presentation more in line with that for the synthetic experiments, we divided this data set into two groups. The images in the one group were judged to have minimal specularities. There were 223 such images. The other group of 98 images was judged to have significant standard specular reflections.

The second image data set was developed specifically for this study. We used 14 scenes with metallic specularities under the same 11 illuminants used in the first set. Again, some images were culled, leaving 149. Figure 8.4 shows the 14 scenes under the canonical illuminant, and Figure 8.5 shows one of the 14 scenes under all 11 illuminants. The images were taken at low enough light to minimize clipping due to specularities, and the dynamic range was extended by averaging multiple frames. This allows us to investigate strong specularities, and the possibilities afforded by higher dynamic range cameras.

We plot the results of incorporating the gamut-mapping extensions in the case of the two image data in Figure 8.6. Again we focus on the results of using these algorithms in conjunction with the gamut-mapping modifications introduced in Chapter 6. For the interested reader, the results using a number of additional variants and comparison algorithms is provided in Table 8.4.

The image data results generally confirm the results found with synthetic data. Again, the largest improvement occurred when the algorithm extended for metallic specularities was used on images of scenes with metallic surfaces. Interestingly, the penalty for using this method when there were no such surfaces was less than in the case of generated data. This may be due to the specific collection of images—the penalty may be larger with a different set of images. However, it is equally plausible that the complexities in image data reduces the performance of the less sophisticated algorithms proportionally more. Thus we feel that additional data is required to clarify the overall efficacy in modeling metallic specularities in the manner described here.

### 8.5 Conclusions

We have considered computational colour constancy in the context of scenes with both specular and non-specular surfaces. We have also proposed an algorithm which is explicitly designed to make use of both types of information. This is in contrast to most current colour constancy algorithms which focus on using only the matte surfaces or only the specularities. Unlike other algorithms using specular information, our method does not need to identify groups of pixels as corresponding to the same surface under different geometry. Instead, the method implicitly uses the information inherent in the brightness of the image pixels. However, since the method extends the already capable gamut mapping approach, the method can give good results even when specularities are not present. In addition, our method is easily extended to deal with specular reflection from coloured metals, and is the first colour constancy algorithm to do so. Adding this capability significantly improves the illuminant chromaticity estimation performance when such surfaces are present. Unfortunately, it often incurs a non-negligible penalty when such surfaces are not present. Additional work is required to evaluate further the overall effect, and more importantly, to reduce the penalty.

This work also makes a contribution to the ability of the threedimensional gamut mapping algorithms to estimate the illuminant brightness when specularities are present. Under these conditions, practically all algorithms which estimate illuminant brightness are severely handicapped. Thus it is very encouraging that our method gives relatively stable estimates of the brightness when specularities are present, absent, and when a significant portion of them are clipped.

## **INSERT COLOUR PLATE HERE**

Figure 8.4: The 14 scenes with metallic specularities used for image data.

## **INSERT COLOUR PLATE HERE**

Figure 8.5: One of the scenes with metallic surfaces under all 11 illuminants.



Algorithm test condition

**Figure 8.6**: Chromaticity performance of some of selected algorithms on three image data sets with and without clipping. More detailed results can be found in Table 8.4.

### COLOUR CONSTANCY WITH SPECULAR AND NON-SPECULAR SURFACES 181

Algorithm	RMS of (r,g) chromaticity error over 223 images with minimal specularities	RMS of (r,g) chromaticity error over 98 images with standard specularities		RMS of (r,g) chromaticity error over 149 images with some metallic specularities	
	(±5%)	(土)	7%)	(±6%)	
			With clipping		With clipping
NOTHING	0.126	0.124	0.124	0.129	0.124
AVE-ILLUM	0.095	0.093	0.093	0.101	0.096
GW	0.088	0.139	0.141	0.123	0.122
DB-GW	0.071	0.118	0.119	0.102	0.100
RETINEX	0.052	0.080	0.083	0.103	0.095
CIP-ICA	0.081	0.083	0.079	0.096	0.095
NEURAL-NET	0.058	0.090	0.091	0.093	0.085
C-by-C-01	0.071	0.087	0.077	0.092	0.090
C-by-C-MAP	0.062	0.090	0.090	0.097	0.089
C-by-C-MMSE	0.060	0.089	0.089	0.095	0.087
CRULE-MV	0.036	0.059	0.065	0.104	0.098
CRULE-AVE	0.043	0.053	0.064	0.090	0.090
ECRULE-MV	0.035	0.052	0.059	0.091	0.088
ECRULE-ICA	0.045	0.052	0.057	0.082	0.081
ECRULE-SCWIA-12	0.032	0.049	0.052	0.080	0.080
ND-ECRULE-MV	0.034	0.046	0.061	0.080	0.075
ND-ECRULE-ICA	0.045	0.047	0.055	0.070	0.072
ND-ECRULE-SCWIA-12	0.034	0.051	0.053	0.071	0.070
SP-ECRULE-MV	0.047	0.053	0.059	0.090	0.088
SP-ECRULE-ICA	0.043	0.053	0.056	0.082	0.081
SP-ECRULE-SCWIA-12	0.038	0.051	0.051	0.081	0.080
MET-ECRULE-MV	0.056	0.052	0.057	0.082	0.082
MET-ECRULE-ICA	0.052	0.052	0.055	0.076	0.078
MET-ECRULE-SCWIA-12	0.049	0.050	0.048	0.072	0.075
SP-ND-ECRULE-MV	0.047	0.048	0.060	0.080	0.075
SP-ND-ECRULE-ICA	0.041	0.046	0.053	0.070	0.072
SP-ND-ECRULE-SCWIA-12	0.038	0.047	0.051	0.071	0.070
MET-ND-ECRULE-MV	0.052	0.053	0.054	0.069	0.070
MET-ND-ECRULE-ICA	0.047	0.049	0.051	0.062	0.068
MET-ND-ECRULE-SCWIA-12	0.044	0.054	0.045	0.059	0.064

**Table 8.4**: Chromaticity performance on a number of algorithms on three image data sets with and without clipping.

# **Chapter Nine**

## Gamut Mapping for the Real World

The preceding three chapters have introduced several extensions to gamutmapping algorithms geared towards improving colour constancy on image data. Specifically, these were the new methods for solution selection, the improvements to deal with diagonal model failure, and the modification allowing the incorporation of specular information. In this chapter we will present the results of combining these methods and applying them to image data.

## 9.1 Combining the Methods

The previous chapters have described five separate extensions to gamut mapping, which can be broken down into three basic ideas: The new solution selection method, the modification for diagonal model failure and its application to fluorescent surfaces, and the modification for specularities and its application to metallic surfaces. These three ideas are independent, and thus additional algorithms can be constructed by combining instantiations from each group. We have already discussed a few such algorithms in previous chapters. In this chapter we take this idea further. The main point of interest is the degree to which increased robustness can be achieved by combining these methods. We are interesting in creating algorithms that do not result in excessive decrements in performance when used on data which does not fit the proposed model. If an improvement under one set of conditions is accompanied by a degradation under other conditions, then the overall performance is likely to reduced, rendering the 'improvement' an actual degradation. Thus a number of combinations of the extensions to gamut mapping are tested on diverse images, and compared to other existing algorithms. This allows a consideration of the costs, as well as the benefits of the new extensions.

### 9.2 Results

To present the results, we will rely on the labeling scheme developed in the preceding chapters. For example, we will express the combination of the fluorescent extension with the extension for metallic specularities by the prefix MET-FL. The results of each algorithm are provided for the data set of standard images, the data set of images with metallic surfaces, the data set of images with fluorescent surfaces, and the combined result over all images. We remind the reader that the weighting of the overall data set is 321 standard images, 149 images with metallic surfaces, and 59 images with fluorescent surfaces, for a total of 529. The results for a selection of the algorithms are shown in Figure 9.1, and more detailed results are presented in Table 9.1.

One new observation is that when the modifications for specularities (SD/MET) are combined with those for diagonal model failure (ND/FL), the efficacy of the maximum volume estimate is diminished in comparison to averaging. This is accompanied with a downward shift of the optimal value of the exponent used with the SCWIA method. In general, we do not wish to find a precise optimal value of this exponent for every algorithm, as it is likely to vary with a number of circumstances. However, we have found that in the case of the SD/MET with ND/FL combinations, a value of 6 tends to be

somewhat better than the value of 12 we have been using so far, and thus we report the results using a value of 6 in these cases.

In general, the results are promising. Our search for more robust methods has produced a number of algorithms which out-perform existing methods over the 529 images. The two most significant improvements resulted from the work on solution selection and modeling fluorescence. Interestingly, although there were only 59 images with fluorescent surfaces, using the extension for fluorescence had a significant overall effect. Thus this method offers a benefit when these surfaces are present, with little cost when they are not. The extension for metallic surfaces also improved the overall results, although unfortunately, not when the fluorescent extension was in place. To clarify, both the fluorescent extension and the metallic extension reduced the error, but the error with the fluorescent extension alone was close to minimal (SP-FL-ECRULE-SCWIA-6 had slightly less error), and adding the metallic extension on top of it increases the error. We note that the some of these finer distinctions are close to our error estimates, and could well be due to natural variation in the difficulty of a specific set of images for specific algorithms.

### 9.3 Conclusion

The analysis in this chapter has given us confidence that our methods are indeed able to improve the robustness of computational colour constancy. Specifically, we have been able to develop methods which do not suffer much degradation when used in a wide variety of circumstances. This is important, because we currently do not have a good description of the circumstances under which colour constancy will be used. For example, we do not know how often a consumer camera will be pointed at fluorescent surfaces or a metallic specularity. Thus we are encouraged that a number of the algorithms developed in this thesis do well in general, and, as a group, significantly outperform existing algorithms. This is more important than which algorithm is "best", as the exact ranking will certainly change with a second set of images.





Vector distance between (r,g) of illuminant and estimate thereof



Algorithm	RMS of (r,g) chromaticity error over 321 standard images	RMS of (r,g) chromaticity error over 59 images with fluorescent surfaces.	RMS of (r,g) chromaticity error over 149 images with metallic specularities	RMS of (r,g) chromaticity error over all 529 images used.
	(±4%)	(±10%)	(±6%)	(±3%)
NOTHING	0.125	0.115	0.129	0.125
AVE-ILLUM	0.094	0.088	0.101	0.095
GW	0.106	0.161	0.123	0.118
DB-GW	0.088	0.135	0.102	0.098
RETINEX	0.062	0.110	0.103	0.081
CIP-ICA	0.081	0.077	0.096	0.085
NEURAL-NET	0.069	0.065	0.093	0.076
C-by-C-MMSE	0.070	0.081	0.095	0.079
FL-C-by-C-MMSE	0.068	0.067	0.093	0.076
CRULE-MV	0.045	0.109	0.104	0.075
CRULE-AVE	0.046	0.091	0.090	0.067
ECRULE-MV	0.041	0.075	0.091	0.063
ECRULE-ICA	0.047	0.060	0.082	0.060
ECRULE-SCWIA-12	0.037	0.064	0.080	0.056
ND-ECRULE-MV	0.042	0.080	0.080	0.060
ND-ECRULE-ICA	0.048	0.059	0.070	0.056
ND-ECRULE-SCWIA-12	0.040	0.066	0.071	0.054
FL-ECRULE-MV	0.039	0.074	0.072	0.054
FL-ECRULE-ICA	0.052	0.053	0.068	0.057
FL-ECRULE-SCWIA-12	0.038	0.050	0.062	0.047
SP-ND-ECRULE-MV	0.050	0.085	0.080	0.065
SP-ND-ECRULE-ICA	0.043	0.059	0.070	0.054
SP-ND-ECRULE-SCWIA-6	0.042	0.063	0.069	0.054
MET-ND-ECRULE-MV	0.052	0.082	0.069	0.061
MET-ND-ECRULE-ICA	0.047	0.058	0.062	0.053
MET-ND-ECRULE-SCWIA-6	0.043	0.062	0.058	0.050
SP-FL-ECRULE-MV	0.044	0.074	0.075	0.058
SP-FL-ECRULE-ICA	0.042	0.047	0.063	0.050
SP-FL-ECRULE-SCWIA-6	0.038	0.046	0.062	0.047
MET-FL-ECRULE-MV	0.056	0.073	0.070	0.062
MET-FL-ECRULE-ICA	0.050	0.049	0.059	0.052
MET-FL-ECRULE-SCWIA-6	0.046	0.047	0.056	0.049

**Table 9.1**: Chromaticity performance of selected comprehensive algorithms on image data.

# **Chapter Ten**

# Colour by Correlation in a Three-Dimensional Colour Space

Colour by Correlation [62] is a promising new method for computational colour constancy. It is promising because it can combine more sources of information than gamut-mapping, and thus is potentially more effective. The extra source of information that becomes available is the statistical distribution of expected surfaces and illuminants, and how their interactions affect the expected statistical distribution of the observed camera responses. However, the current version of Colour by Correlation uses only chromaticity information. We remind the reader that the work in the previous chapters (most explicitly presented in Chapter 4) has shown that it is beneficial to use the magnitude information, even if only the illuminant chromaticity is being sought. Thus it is natural to modify Colour by Correlation so that it can also use this information.

In this chapter we provide details of the changes required to have this algorithm work in a three-dimensional colour space. The modified algorithm naturally allows extensions for both metallic and non-metallic specularities, in analogy with the previous work on gamut-mapping for these conditions. In addition, the algorithm can deal with fluorescent surfaces, much like the two-dimensional version, as investigated in Chapter 7. We remind the reader that details of two-dimensional Colour by Correlation method are provided in Chapter 4.

### 10.1 The Extension to Three Dimensions

We begin by considering what a three-dimensional analog to the twodimensional algorithm entails. In the two-dimensional case, the observed two-dimensional descriptors (chromaticities) were tested against possible theories of the distributions of those descriptors, each theory corresponding to one of the illuminants in the training set. In the three-dimensional version, we wish to do the same with three-dimensional descriptors. However, we run into the problem that the brightness of the illuminant changes the observed values. In effect, not only does each illuminant produce a theory, but every brightness level of each illuminant produces a theory. Thus we must attempt to match over possible illuminant brightnesses, as well as over illuminants. This leads to several problems.

The first problem is that, a priori, the illuminant can have any nonnegative brightness. This is different than chromaticity which is naturally constrained, and thus easily discretized. To solve this problem we propose making an initial estimate of the illuminant brightness using some other means. For this, we found a grey world type estimate to be adequate. Specifically, we compute the average of R+G+B over the image pixels, and multiply the result by a factor chosen to give the best estimate when the same procedure was applied to synthetic data. The value of the factor used for the experiments was 4.3. We use L to represent the estimate of the estimate of the illuminant brightness.

Having determined an estimate of the illuminant brightness, we reason that it is unlikely to be wrong by more than a factor of k=3. Now, on the assumption that the illuminant brightness is between L/k and kL, we discretize this range on a logarithmic scale, giving us a finite number of

possible illuminant brightness theories. We verified that the specific choice of k=3 gives the same results as providing the algorithm with the exact illuminant brightness in its place. Clearly, a larger or smaller value could be more appropriate, depending on circumstances.

The next problem that we faced is that the literal analogy of the twodimensional method leads to unmanageably large correlation matrices. There are two contributions to the increase in size. First, the matrix row length increases because of the added descriptor—the rows now store linearized versions of three-dimensional arrays where two-dimensional arrays were previously stored. Second, the strategy of considering each illuminant at each brightness level implies, a priori, that we would further need to increase the number of rows by a factor of the brightness resolution because now we would need a row for every brightness of every illuminant. The combined effect of these two factors lead to correlation matrices which are simply too large.

Fortunately, the second increase in size is not necessary. We instead loop over the possible brightnesses, and simply scale the input by an appropriate amount each time. Conceptually, this amounts to the same thing as having a correlation matrix row for each illuminant at each brightness. In practice, however, it leads to a subtle problem due to the discretization. If we consider the alternative of building a correlation matrix row for each possible brightness, we see that as the proposed illuminant brightness decreases, the bins become proportionally more populated. For example, if the illuminant brightness is halved, then the same data is put into half as many bins. Now the algorithm proceeds by summing terms for each observed response. The terms are the logarithms of quantities proportional to the probability that a proposed illuminant occurs with the observed response. If we consider each term to be negative, then we see that decreasing the number of terms increases the sum. Since we are trying to maximize this sum, the algorithm will favor low brightness values, because these tend to put the observations into as few bins as possible, leading to fewer terms. This is an artifact of the discretization, and clearly is not wanted.

We discuss two possible approaches to deal with this problem. First, we can allow duplicate entries into the bins. In order to minimize the effect of duplicates present in the data, the data could be pre-processed to remove all initial duplicates. This method gives excellent results when used in conjunction with generated data. However, we have not had equal success with image data.

A second approach to the above problem is to compensate for the discretization problem directly. We reason as follows: If we were to have constructed correlation matrices for each brightness level, then the frequency counts placed in the bins to compute the probabilities would have been roughly inversely proportional to the brightness. Thus the probabilities themselves would be inversely proportional to the brightness, and to obtain a fair estimate, we need to divide each probability in the product by a value proportional to the brightness. In the log representation, this means that we subtract the log of the brightness times the number of occupied bins. This method also yields excellent results when used in conjunction with generated data. More importantly, the results using image data are also promising. We feel that this algorithm can be substantially improved, and one of the key areas for further study is this discretization problem.

We now consider the choice of three-dimensional descriptors. One natural choice is RGB. However, given the asymmetry of the role of brightness and chromaticity in computational colour constancy, we feel that a better choice is to use (r,g) chromaticity, together with R+G+B. This has several advantages over using RGB. First, due to the above mentioned asymmetry, we may wish to use different resolutions for the chromaticity and the brightness. Second, this choice provides conceptual clarity, in that our method then subsumes the two-dimensional version as the sub-case where there is only one division for the R+G+B coordinate. Finally, we find it convenient to have only one coordinate which can be arbitrarily large.

The algorithm as described is easily extended to model complex physical scenes. For example, we can model fluorescent surfaces, as already done in the two-dimensional case in Chapter 7, and we can model specular surfaces, including metallic ones, as was done for gamut-mapping in Chapter 8. The Colour by Correlation method has an advantage over the gamut-mapping methods in that the expected frequency of occurrence of these phenomena can be modeled. Unfortunately we currently do not know these statistics for the real world, and hence it is difficult to exploit this in the case of image data. Nevertheless, doing so holds promise for the future because if some estimate of the likelihood of occurrence of these classes of surfaces could be made, then three-dimensional Colour by Correlation would be more robust than the extended versions of three-dimensional gamut mapping. This is due to the fact that it can allow for the possibility of, for example, metallic surfaces, while compensating for the fact that there is only a low likelihood that such surfaces are present. Gamut-mapping, on the other hand, is forced to use uniform statistics.

### 10.2 Algorithm Summary

We now provide a summary of the method. The implementation of the algorithm consists of two parts. First the correlation matrices are built, and then these matrices are used to perform colour constancy. The first stage is a one time operation, and consequently, we are not concerned about resource usage. We begin with a data set of illuminant and reflectance spectra. Ideally, we would know the expected frequency of occurrence of these surfaces and illuminants, but since we do not, we assume that there are all equally likely. The reflectance and illuminant spectra sets are the ones used in Chapter 4 and described more fully there.

We use the colour space (r,g,L) where L=R+G+B, r=R/L, and g=G/L. We divide the space into discrete bins. The resolution of the discretization of the three components do not need to be equal. There is no reason to make the first two different from each other, but, as discussed above, it can be advantageous to use a different value for the third. For all experiments we used 50 divisions for (r,g), which is consistent with the discretization resolution used in this thesis for two-dimensional Colour by Correlation, as well as the neural net method. When specularities are added, as discussed shortly, the overall number of bins required for L increases. We express the resolution for L in terms of the number of bins devoted to matte reflection. For the experiments with generated data, we generally used a value for L which also leads to 50 divisions for matte reflection, but this is likely higher resolution than is necessary, and in fact, preliminary results indicate that a smaller number is likely better. Thus for the image data experiments, we used 25 divisions.

Given a discretization of colour space, we then map this space into a vector, using any convenient method. We note that since half of the values in (r,g) are impossible, a more compact representation can be used than the naive one. Since the three-dimensional correlation matrices are large, we make use of this observation to reduce storage requirements.

Thus we form a two-dimensional array, where each row is the above linearization of colour space, and the rows correspond to training illuminants. We then build up the matrix by computing, for each illuminant, the RGB of the reflectances in our database. We then compute the frequency of occurrence of the colours within each discrete cell in our colour space. These frequencies are proportional to the probabilities; they can be converted to probabilities by dividing by the total number of surfaces. Finally, for convenience, we store the logarithm of the probabilities.

To add fluorescent surfaces, we compute the responses which occur for each illuminant using the model described in Chapter 7. The relative expected frequency of such surfaces is expressed by simply adjusting the frequency counts during the construction of the correlation matrix. In our experiments with fluorescent surfaces, we set the frequency of occurrence of any fluorescent surface to be about 15%. Since we only model 9 such surfaces, the frequency of occurrence of each was set to be 50 times that of each of the surfaces in the set of roughly 2000 reflectances.

We can also model specular reflection. This is a little more involved than handling fluorescent surfaces. First, we need to extend the number of bins in the L direction, as specular reflection is modeled as reflection which exceeds that of a perfect white. Then, we must model both the relative frequency of occurrence of specularities, as well as the frequency of each degree of specular reflection. It should be clear that the model can well be used with metallic specularities, but we do not study those here.

The second part of the algorithm is the use of the above matrix for colour constancy. We wish to compute the likelihood of an illuminantbrightness combination. We loop over the possible illuminants, and then the possible brightnesses, to obtain an estimate for each combination. To compute a maximum likelihood estimate, we simple keep track of the maximum value reached and the corresponding illuminant and brightness. However, since we are also interested in studying the mean likelihood estimate, we store all values in order to make that estimate from them as a second step. We now provide additional details of the likelihood calculation.

Again, for each proposed illuminant, we loop over a discretization of possible brightnesses on a log scale. We remind the reader that the range is set by an initial rough estimate of the brightness. We generally use 101 brightness levels; This is likely excessive. For each proposed brightness level, we scale the input accordingly, using the brightness of the proposed illuminant. We then form a vector representing the observed scene assuming this brightness level. The components of this vector correspond to the linearized form of the discretized colour space.

To compute the entries of this vector we begin by initializing all components to zero. We then compute the corresponding bin for each colour observed in the scene. If we are using the first method to solve the discretization problem discussed in the previous section, then we store a count of the number of colours falling in each bin. Alternatively, if we are using the second method we simply note the presence of the colour with a count of one. All bins corresponding to colours not observed remain zero.

To obtain the likelihood of the proposed illuminant-brightness combination, we simply take the dot product of the computed vector with the row in the correlation matrix corresponding to the proposed illuminant. Since the values stored in the correlation matrix are the logarithms of probabilities, the dot product computes the logarithm of the product of the probability contributions for each observation (see Equation 4.5). If we are using the second method to compensate for the discretization problem discussed above, we then adjust the result by subtracting the logarithm of the proposed brightness times the count of the occupied bins.

### 10.3 Experiments

We tested the new algorithm on generated and image data. For the first two sets of results on generated data we used the first method of dealing with the discretization problem. For the third set of results with generated data, as well as for the image data results, we used the second method. For the experiments with generated data we used the same set of test illuminants as used in Chapter 4. We remind the reader that both the training illuminant set and the test illuminant set were designed to systematically cover (r,g) space, but the test illuminant set covered that space four times more densely.

Figure 10.1 shows the chromaticity performance of the method using both maximum likelihood and mean likelihood estimation as a function of the number of surfaces in the generated scenes. We also provide the results for corresponding two-dimensional versions of the algorithms, as well as the results for two gamut mapping methods—the original CRULE-MV method, and the ND-ECRULE-SCWIA-12 method introduced in Chapter 6.

The results clearly show that the new method excels when tested on data with similar statistics to that used for training. The error drops to the minimum possible given the discretization when only 16 surfaces are used, clearly out-performing the other algorithms.

For the second experiment we looked at the performance of the method under a variety of conditions. We developed three-dimensional algorithms for fluorescent surfaces and specular reflection, and tested these, along with the algorithm for matte surfaces, under the conditions of the Mondrian world, the Mondrian world with fluorescent surfaces, and the Mondrian world with specularities. The test conditions were similar to the training conditions, especially in the fluorescent case. In the specular case, the rough discretization of specular reflection used for creating the correlation matrices only approximates what the algorithms were tested against.



**Figure 10.1**: The chromaticity performance of the new method as compared to the twodimensional version of the algorithm and two gamut mapping methods. For both Colour by Correlation methods we provide results using both maximum likelihood (MAP) and mean likelihood (MMSE) estimation.

### COLOUR BY CORRELATION IN A THREE DIMENSIONAL COLOUR SPACE 196

Again the results, shown in Table 10.1, are very promising. As expected, the algorithms do very well when tested under the conditions they were designed for. More promising is that the algorithms seem quite robust to the absence of these conditions. For example, adding fluorescent capability reduced the error from 0.060 to 0.022 when fluorescent surfaces were present, but using the algorithm with fluorescent capability in the case of standard surfaces incurred minimal penalty (0.026 instead of 0.025). (These figures are using the MMSE estimator). In general, it is clear that for generated data, these algorithms perform better than any of the others which are listed in Table 10.1.

For the third experiment, we tested the second method of dealing with the discretization problem discussed above. The results are shown in Table 10.1. We also include additional comparison algorithms in this table. Again, the new methods do significantly better than the next best strategy, which is the ND-ECRULE-SCWIA-12 algorithm developed in Chapter 6. Using the MMSE estimator, the three-dimensional Colour by Correlation error is 0.24; using the MAP estimator it is 0.29; and using ND-ECRULE-SCWIA-12 it is 0.39. The results also indicate that the second method of dealing with our discretization problem may be better than the first, as the errors are lower, but we note that the difference can also easily be explained by random fluctuations within our error estimates.

We also tested the method on the image data set collected for the study presented in Chapter 4. As mentioned above, we have not yet been able to significantly improve upon the two-dimensional method using the first method of dealing with our discretization problem. Using the second method, however, the results, shown in Table 10.3, are promising. We see that the error of the new method (0.46) is approaching that of the best performers listed, namely ECRULE-SCWIA-12 (0.37) and ECRULE-MV (0.041). This error is significantly less than that for the two-dimensional counter-part (0.077).

	Synthetic scenes with 8 matte surfaces	Synthetic scenes with 8 matte and fluorescent surfaces	Synthetic scenes with 8 matte and specular surfaces
NOTHING	0.116	0.114	0.110
AVE-ILLUM	0.088	0.086	0.084
GW	0.057	0.116	0.034
DB-GW	0.047	0.092	0.032
RETINEX	0.066	0.104	0.033
CIP-MV	0.197	0.181	0.205
CIP-AVE	0.128	0.118	0.134
CIP-ICA	0.076	0.072	0.077
NEURAL-NET	0.038	0.046	0.034
SP-NEURAL-NET	0.040	0.046	0.033
C-by-C-01	0.078	0.071	0.079
C-by-C-MAP	0.044	0.059	0.040
C-by-C-MMSE	0.037	0.048	0.033
RGB-C-by-C-MAP	0.030	0.066	0.043
RGB-C-by-C-MMSE	0.025	0.060	0.037
FL-RGB-C-by-C-MAP	0.033	0.023	*
FL-RGB-C-by-C-MMSE	0.026	0.022	*
SPEC-RGB-C-by-C-MAP	0.038	*	0.023
SPEC-RGB-C-by-C-MMSE	0.032	*	0.017
CRULE-MV	0.050	0.103	0.027
CRULE-AVE	0.061	0.088	0.052
ECRULE-MV	0.045	0.078	0.026
ECRULE-ICA	0.051	0.065	0.045
FL-ECRULE-MV	0.049	0.061	0.027
FL-ECRULE-ICA	0.058	0.051	0.056
SP-ND-ECRULE-MV	0.053	0.085	0.029
SP-ND-ECRULE-ICA	0.047	0.062	0.026

**Table 10.1**: Algorithm chromaticity performance under three different conditions of variants of the new methods designed for the various conditions, as well as that for a number of comparison algorithms. For these results, the first method of dealing with our discretization problem was used.

Algorithm	Performance estimating (r,g) chromaticity of the illuminant
	(±4%)
NOTHING	0.111
AVE-ILLUM	0.083
GW	0.055
DB-GW	0.047
RETINEX	0.061
CIP-MV	0.203
CIP-AVE	0.132
CIP-ICA	0.074
NEURAL-NET	0.039
C-by-C-01	0.076
C-by-C-MAP	0.043
C-by-C-MMSE	0.035
STD-RGB-C-by-C-MAP	0.029
STD-RGB-C-by-C-MMSE	0.024
CRULE-MV	0.048
CRULE-AVE	0.062
ECRULE-MV	0.043
ECRULE-ICA	0.050
ECRULE-SCWIA-12	0.040
ND-ECRULE-MV	0.045
ND-ECRULE-ICA	0.049
ND-ECRULE-SCWIA-12	0.039

**Table 10.2**: Algorithm chromaticity performance in the Mondrian world of the new method (MAP and MMSE), as well as that for a number of comparison algorithms. For these results, the second method of dealing with our discretization problem was used.

Algorithm	Performance estimating (r,g) chromaticity of the illuminant.
	(±4%)
NOTHING	0.125
AVE-ILLUM	0.094
GW	0.106
DB-GW	0.088
RETINEX	0.062
C-by-C-01	0.075
C-by-C-MAP	0.084
C-by-C-MLM	0.081
C-by-C-MMSE	0.077
RGB-C-by-C-MAP	0.047
RGB-C-by-C-MMSE	0.046
CRULE-MV	0.045
CRULE-AVE	0.046
ECRULE-MV	0.041
ECRULE-ICA	0.047
ECRULE-SCWIA-12	0.037

**Table 10.3**: Algorithm chromaticity performance on 321 images of the new method with two estimators (MAP and MMSE), as well as that for a number of comparison algorithms. For these results, the second method of dealing with our discretization problem was used.

## 10.4 Conclusion

We have shown how to modify the Colour by Correlation algorithm to work in a three-dimensional colour space. This was motivated by the observations that the correlation method is more powerful than the chromaticity gamut-mapping method due to the use of statistical information, and that three-dimensional gamut mapping is also more effective than its chromaticity counterpart due to the use of information inherent in the pixel brightness. We wished to combine these two features into one algorithm The resulting algorithm is also suitable for modification
to deal with complex physical surfaces such as fluorescence, and standard and metallic specularities. In fact, if the frequency of occurrence of these surfaces could be estimated, then this algorithm could also exploit these statistics. In summary, this algorithm is able to use more sources of information than any other, and thus is potentially the most powerful colour constancy method.

We tested a number of versions of the algorithm on synthetic and image data. The results with synthetic data are excellent, and it seems that these algorithms are in fact the best performers in this situation. The results with image data are also promising. In this case the new methods perform significantly better than their two dimensional counterparts. Currently, however, the performance still lags a little behind the best algorithms for image data. It is quite possible that the performance gap between real and image data can be reduced, as we have only recently begun to study the algorithm in this context. However, the work in Chapter 4 has shown that statistical algorithms do tend to shine during synthetic testing, and therefore, we must be cautious not to over-sell the method until the image data performance exceeds that of the current best methods.

Finally we note that the algorithm as described is computationally quite expensive, both in terms of memory use, and CPU time. Since our initial intention was to push the limits of the error performance, we have not addressed ways to speed up the algorithm. If the performance on image data can be made comparable to that for generated data, then an important next step is to consider what can be done in this regard.

# **Chapter Eleven**

## Is Machine Colour Constancy Good Enough?

In this final chapter of this thesis, we ask whether machine colour constancy has matured to the point where it would be useful in other aspects of machine vision. We wish to relate the colour constancy error measures used in the previous chapters to the performance on some appropriate real world task. For this task we choose colour-based object recognition. The object recognition strategy is Swain and Ballard's "colour indexing" method [41], which is based on comparing histograms of the distribution of image colours. Colour indexing fails miserably when the ambient light illuminating the object to be recognized differs from that used in constructing the database of model images. Swain and Ballard suggest using colour constancy preprocessing as a way of addressing this problem; however, it has since been solved by introducing illumination-independent representations (e.g., relative colour instead of absolute colour [116], moment-based representations of colour histograms [117], normalized angles of image channel vectors [94], or simply normalizing the images [96]). Nonetheless, if colour constancy methods work then it seems a natural task for them to be used in preprocessing images prior to indexing as Swain and Ballard originally

suggested. Furthermore, we point out that indexing is not the same task as object recognition. For example, illumination invariant indexing cannot locate a uniformly colored object in an image.

Clearly, the fact that colour indexing is sensitive to variations in the ambient scene illumination is to be expected, since it is an entirely colourbased method and the scene illumination directly affects the image RGB colour<sup>1</sup>. The question we address here is whether or not existing colour constancy algorithms are effective enough at generating illuminationindependent colour descriptors that colour indexing will work under the typical range of scene illuminations that are encountered in practice such as daylight, tungsten light, and fluorescent office lighting. Since our goal is to test colour constancy, not to develop a new and improved object-recognition scheme, we use colour indexing without modification.

### 11.1 Colour Indexing and Colour Constancy

Colour constancy processing as developed in the proceeding chapters provide illuminant independent descriptors of the scene surface colours. Specifically, we use the surface colours as they would have appeared under some chosen 'canonical' illuminant. As discussed in previous chapters, many colour constancy methods estimate only the chromaticity of the colours under the canonical illuminant and ignore the intensity component. Since the magnitudes of the observed responses is a function of illumination geometry, chromaticity also makes sense for indexing. Thus, in this chapter, colour indexing will be based on the standard chromaticity coordinate space:

$$r=R/(R+G+B); g=G/(R+G+B)$$
 (11.1)

Colour constancy algorithms will be used to convert between chromaticity 'images', in other words from the chromaticity under the unknown illumination  $rg_{unknown}(x,y)$  they will provide an estimate of what the

<sup>&</sup>lt;sup>1</sup> RGB space defined as the output of our SONY DXC-930 3-CCD colour video camera. Strictly speaking 'colour' is what a human observer perceives, but in this work we will also use it to refer to a pixel's RGB.

chromaticity  $rg_{canonical}(x,y)$  would have been under the canonical illumination.

Colour indexing is performed using 2-dimensional chromaticity histograms. Swain and Ballard did the majority of their tests using RGB but they included some tests with rg-chromaticity space. The method is quite simple. First a database of model (chromaticity) histograms is created from images of the objects that we wish the system to recognize. The objects need to be separated from the background before the database is built. This segmentation can be done manually if need be. Given an image of an object to be recognized—call it the 'test' object—its chromaticity histogram is determined. Unlike the case for the model objects, the test object does not need to be separated from the image background. The test histogram T is then intersected with each model histogram M in the database, where intersection is defined as,

$$H(T, M) = \sum_{j=1}^{\infty} \min(T_j, M_j) / \sum_{j=1}^{\infty} M_j$$
(11.2)

The model with the highest histogram intersection score is used to identify the unknown object.

In our implementation the chromaticity histograms are 16x16. This sampling might be too coarse for a very large image database, but for our purposes the coarse sampling should help tolerate inaccuracies in colour constancy.

### 11.2 The Test Images

The images used for the experiments are of 20 different, relatively colourful objects, taken under the 11 illuminants used previously, and described more fully in Chapter 4. In total, 220 images were used. The objects under the canonical illuminant are shown in Figure 11.1, and a sample object under all 11 illuminants is shown in Figure 11.2. The images were taken using the same camera and paradigm as all other images used for this thesis, with one important exception: With every illuminant change, the object was rotated or moved in some way. This created a real indexing task.

# **INSERT COLOUR PLATE HERE**

Figure 11.1. The 20 objects in the image database as seen under a single illuminant

# **INSERT COLOUR PLATE HERE**

Figure 11.2. The cruncheroos object as seen under the 11 test illuminants

We consider the images taken under the canonical illuminant as a model database. The task of indexing is to take images, and find them in the model set. This is made non-trivial by two factors. First, the orientation, and the exact size of the object in the test image is different than it is in the model set, and second, the colours are different due to the illumination change. Of course, 1/11 of the time, these factors are absent, but the colour problem still exists because the colour constancy algorithms do not know that the colour is already correct.

### 11.3 Results

First we comment that the colour constancy results confirm the image data colour constancy performance work developed in this thesis. We stress that the images used in this experiment were taken after all algorithms had been developed, and thus is a completely independent test of colour constancy performance. Specifically, we see that the three-dimensional gamut-mapping algorithms are collectively the best performers, Interestingly, averaging does better on this data set than the maximum volume solution selection method, with the new method of choosing the solution being the same as averaging within error. We feel this slight deviation from earlier experiments is due to the overall lack of specularities in this data set. The essential non-effect of the ND extension to gamut mapping is consistent with the earlier findings, as the camera supports the diagonal model well, and there are no known fluorescent surfaces in these images.

We now consider the colour constancy results in relation to indexing performance. Our experiments confirm the obvious hypothesis that colour constancy is likely to improve colour indexing in situations where the illumination impinging on the test object is different from that used in constructing the model database. In Figure 11.3 we plot indexing performance versus colour constancy error. This graph shows a clear correlation between colour indexing performance and colour constancy error.



Indexing performance versus colour constancy performance for 25 algorithms with and without simulated clipping

Figure 11.3. Object recognition performance as a function of colour constancy error.

Algorithm	Colour constancy performance	Score (1)	Score (2)	Score (3)
ACTUAL	0.000	0.877	0.930	0.934
NOTHING	0.068	0.423	0.493	0.513
AVE-ILLUM	0.048	0.523	0.602	0.623
GW	0.067	0.336	0.486	0.502
DB-GW	0.058	0.505	0.609	0.632
RETINEX	0.055	0.609	0.702	0.711
CIP-MV	0.079	0.245	0.300	0.333
CIP-AVE	0.071	0.155	0.220	0.248
CIP-ICA	0.045	0.518	0.605	0.620
NEURAL-NET	0.048	0.523	0.614	0.635
SPEC-NEURAL-NET	0.053	0.518	0.600	0.620
C-by-C-01	0.043	0.545	0.611	0.628
C-by-C-MAP	0.050	0.509	0.602	0.620
C-by-C-MLM	0.051	0.527	0.618	0.635
C-by-C-MMSE	0.050	0.532	0.614	0.635
CRULE-MV	0.047	0.668	0.748	0.767
CRULE-AVE	0.039	0.718	0.784	0.804
ECRULE-MV	0.040	0.705	0.780	0.795
ECRULE-ICA	0.033	0.677	0.745	0.761
ECRULE-SCWIA-12	0.032	0.727	0.789	0.802
ND-CRULE-MV	0.048	0.668	0.743	0.754
ND-CRULE-AVE	0.038	0.809	0.861	0.866
ND-ECRULE-MV	0.041	0.695	0.777	0.788
ND-ECRULE-ICA	0.031	0.655	0.736	0.748
ND-ECRULE-SCWIA-12	0.032	0.664	0.755	0.764

**Table 11.1:** Colour constancy error and indexing performance by algorithm. The error is the vector difference in the chromaticity of the illuminant and the estimate thereof. The results are the RMS of 220 values. Score(1) is the percentage of correct matches. Score(2) is score(1) plus 1/2 a similar score for rank 2. Score(3) is score(2) plus 1/3 a similar score for rank 3. Score (3) is used for Figure 11.3. See Chapter 4 and Chapter 6 for an explanation of the algorithm labeling, and the details on the algorithms themselves.

The recognition performance measure used in Figure 11.3 is based on a weighted average of colour indexing's rankings. During the recognition phase, colour indexing calculates match strengths for each model in the database. If the strongest match is in fact the correct object, then we say that we have a rank one match. If the correct object is the algorithm's second choice, then we have a rank two match, and so on. For each algorithm, we obtain the number of matches for each rank. The ranking is the position of the correct matches in the ordered list of match scores. To distill these results into a single representative value, we use a weighted sum of the numbers of the first three ranks: the weight for rank one is one, the weight for rank two is 1/2, and the weight for rank three is 1/3. Matches beyond rank 3 are considered failures and count as zero. We normalize the rank by the maximum possible to obtain a final score.

As with all images used for this thesis, this data set has an extended dynamic range with the images being purposefully underexposed in order to prevent any clipping. As explored in several other chapters, this is an advantage to algorithms which are sensitive to clipping. To further explore the capabilities of the algorithms in conjunction with indexing, we simulated exposing the images as they would have been using our automatic aperture which tends to over-expose images. We then re-ran the matching experiments; the results are plotted with filled triangles in Figure 11.3. Finally, for the interested reader, detailed results with no clipping are provided in Table 11.1.

Our original question was: Is machine colour constancy good enough? After a preliminary experiment [107] we felt that answer was no. Based on our new results, we feel that the answer is still no, although some progress is being made. Table 11.1 shows the indexing performance of colour indexing using each algorithm for preprocessing. The best performers find about 65-70% of the objects (rank 1 matches, score(1) in the table); whereas, the results (88% rank 1) based on using the actual illuminant RGB for colour correction indicate that both colour indexing and the diagonal model of illumination change will together support better performance. We note that one algorithm does give a 81% rank 1 performance, but looking at the data as a whole we feel that this is likely abnormally high due to random fluctuations—given the scatter, some algorithm chosen post hoc will likely do this well. We would be more inclined to present this figure as real if it were closer to the fitted line. Thus we conclude that we still have some distance still to go before machine colour constancy is up to the task of supporting object recognition.

In terms of our methodology, we feel that we gave machine colour constancy every reasonable chance, and thus the results should be considered closer to the "best case" than the "worst case". For example, we used illumination with spatially uniform chromaticity and were careful to remove noise through temporal and spatial averaging. We have also taken some trouble to develop a good camera model as required by some of the algorithms. Finally, the database was relatively small, and we avoided bad matches due to colours appearing coincidentally in the background by placing the objects on black cloth.

Having said that, we wish to emphasize some aspects of the experiment that were not open to compromise. First, and foremost, the data is real image data, and the objects are random everyday objects as opposed to, for example, planer non-specular "Mondrians". It is perhaps also of some significance that the images were taken by a research assistant who had little understanding of the intended purpose of the experiment—possibly eliminating any unintended bias in the choice of objects to test. Finally, the range of the illuminants is designed to encompass the bulk of the range encountered in common natural lighting and standard man-made lighting situations.

### 11.4 Conclusion

We tested machine colour constancy algorithms using the computer vision task of colour-based object recognition based on colour histogram intersection. We expected that colour constancy processing would provide colour descriptors that would be accurate enough that colour indexing performance would be close to that obtained when there is no change in the ambient illumination. This is did not turn out to be the case. Colour constancy pre-processing did, however, yield a significant improvement over doing no pre-processing, it simply was not enough of an improvement. Figure 11.3 shows that the degree of improvement in histogram matching appears almost linearly related to the error in illuminant chromaticity prediction.

The results of Brainard et al. [3] indicate that human colour constancy is not all that accurate and state (p. 2101) "Our results represent neither complete constancy nor a complete absence of constancy." The results of our experiments raise the question as to whether or not human colour constancy would be sufficiently accurate for histogram-based object recognition?

# **Chapter Twelve**

## **Conclusions and Contributions**

The work presented here makes a substantial contribution to the goal of making computational colour constancy effective with real image data. Before I embarked on this research, little was known about the relative effectiveness of the various colour constancy methods, especially when applied to images. Without this knowledge, it is difficult to judge which ideas to explore, and which directions to take. By paying close attention to the results on real data, I have been able to make relevant improvements to several algorithms. By the same token, other clever ideas were rejected because they showed less promise once scrutinized. In addition, this work has identified algorithms whose performance on generated data is excellent, but whose performance on images is still lacking. Hopefully the work here will stimulate further improvements to those algorithms, and thus reduce these gaps in real and simulated performance.

The first contribution of this thesis is in the area of camera calibration for colour constancy. The faithful application of computational colour constancy methods to real data requires careful consideration of the camera model used, and the calibration method used to fit the model parameters. To deal with these issues, I developed a camera calibration method which works

better than existing methods on linearized data. Furthermore, by simultaneously solving for the camera sensors and the linearization function, this method achieves a better result than that available when these two aspects of the calibration are sought separately.

The calibration of the vision system goes beyond the model used to predict camera response from input spectra distributions. Computational colour constancy algorithms make two other assumptions which are addressed in this work. First they assume that the system does not have any spatial biases. In other words, all pixels can be considered the same. This is not the case for the system used for the experiments, and therefore I implemented methods to compensate for this. Second, colour constancy algorithms assume that the vision system has infinite dynamic range. Again, this is not the case, and to explore fully colour constancy performance on real data, I adopted two strategies. First, I extended the dynamic range of the system by averaging multiple frames. Second, for some of the experiments, I simulated the effect of lower dynamic range, and the implied clipping of bright pixels.

The faithful application of computational colour constancy methods to real data also requires consideration of image pre-processing. This consideration has largely been ignored. However, the work presented here has shown that image pre-processing is important. For example, it is common to treat each pixel as a different surface. If this is done, then it is easy to create scenes which are unfairly difficult for the grey world algorithm. However, to optimize this algorithm, it is arguable that only distinct surfaces should be used as data points, or, at a minimum, that the effect of patch area should be reduced. Normally this is not done because of the difficulties involved, but in this work, these issues were investigated. Furthermore, I argued that a fair comparison of algorithms requires that each algorithm is used in conjunction with pre-processing tuned especially for that algorithm. Thus I presented results using this strategy.

The controlled comparison of colour constancy algorithms presented in this thesis is important for several reasons. First, the detailed results themselves are of use to those wishing to apply current methods. For such results to be useful, they must be comprehensive. This is because there is no

best algorithm, but rather algorithms which work better or worse depending on the circumstances. This thesis provides the most detailed account of these relative strengths and weaknesses that is currently available. A second contribution is the creation of a paradigm for testing computational colour constancy algorithms. This paradigm includes camera modeling, the choice of the illuminant test and training sets, the diversity of scene surfaces, the degree of specular reflection present, and the role of dynamic range and image pre-processing. Finally, the detailed results provide the foundation and justification for the second part of the second part of the thesis, as well as pointing to additional directions for future work.

One specific conclusion that can be drawn from the comparison work is that the three-dimensional gamut mapping algorithms are very robust when applied to image data, more so than the statistically based methods. This observation induced two research paths. The first path was to study and improve these algorithms. The second path was the development of a three dimensional version of Colour by Correlation. This followed because the results with the gamut mapping algorithms strongly suggested that using the full RGB information is very beneficial, even when only a chromaticity result is sought.

Following the first path lead to a number of contributions to gamut mapping colour constancy. The first was an analysis of the current two methods of choosing a solution from the constraint set. This lead to a new method of choosing the solution. The new method combines the benefits of averaging, while weighting the average more heavily in the direction suggested by the maximum volume heuristic. The degree of weighting is easily configurable. I proposed this strategy because the comparison work had shown that the maximum volume heuristic is a surprisingly good method of choosing the solution when the chromaticity of the illuminant is of most interest, as is often the case. However, there are also good arguments supporting the averaging choice. In addition, experiments with more complex scenes, especially in conjunction with the more sophisticated algorithms developed to deal with them, indicated that some form of averaging was desirable in these cases. Thus it became natural to ask whether

there was a better solution between the two methods, or alternatively, whether evidence could be found suggesting that the optimal was in fact one of the current solutions. The present work indicates that, for most situations, the former is the case. Specifically, the results using the new hybrid method are almost always better than that using the two basic methods.

A second strategy for improving gamut mapping algorithms is to relax their reliance on the diagonal model. This is especially important for cameras with broader, significantly overlapping, sensors. The first approach taken was to apply the method of sensor sharpening. Sensor sharpening is a method which improves the efficacy of the diagonal model in general, and thus can potentially improve many algorithms in addition to the gamut mapping ones. Prior to this thesis, sensor sharpening had not been tested in the context of real colour constancy algorithms. I put the idea of sensor sharpening into practice, and found that current sensor sharpening methods could improve colour constancy in a limited number of situations, but that their application presented several difficulties. This lead me to develop a new sharpening method which is more appropriate for actual, rather that theoretical, colour constancy processing. This new method improves colour constancy most of the time. Furthermore, it is stable in that negative effects, when they occur, are small.

This work also proposes a second strategy for relaxing the diagonal model, specific to gamut mapping. This method modifies the canonical gamut to compensate for expected deviations from the model. In the case of the Sony DXC-930 camera, which supports the diagonal model quite well, this modification generally improves performance by a modest amount. In the case of the Kodak DCS-200 digital camera, which has less sharp sensors, the improvement is significant.

The above modification was originally developed to deal with fluorescent surfaces, and a significant contribution of this thesis is the inaugural treatment of such surfaces in a colour constancy context. To deal with fluorescent surfaces, I proposed a simple method of characterizing them. This was then used in conjunction with the modification above in order to yield gamut mapping algorithms which can deal with such surfaces. Results

using both synthetic data, and 59 real images are very promising. In addition, the characterization of fluorescent surfaces is applicable to the statistically based algorithms; namely, neural net methods and Colour by Correlation. To further investigate colour constancy with fluorescent surfaces, I tested this in the case of Colour by Correlation, and again, the results were very encouraging.

The final modification to gamut mapping algorithms proposed in this work allows them to use specularities to an advantage. Specularities have long been put to use by colour constancy algorithms, but existing algorithms that use specular information are limited in that they require such information to be present. The method presented here combines the use of specular and non-specular information. Thus the algorithm can provide good performance when good specularities are present, or when there is a reasonable diversity of matte surfaces, or some combination of these conditions. Furthermore, it should be noted that the method uses the fact that specular reflection tends to be relatively bright. Most current methods which use specularities do not consider this. Finally, the method is applicable to specular reflection from coloured metallic surfaces, and is, in fact, the first colour constancy algorithm which can use such input gainfully.

A second line of research inspired by the comparison work was to extend the Colour by Correlation method to use pixel brightness. The original Colour by Correlation method used only the chromaticity of the input. However, the success of the three dimensional gamut mapping methods, in conjunction with their close relation to the correlation idea, strongly suggested that Colour by Correlation could be improved by moving it into RGB space. Implementing this idea was not completely mechanical, due to the non-symmetric role of brightness and chromaticity in colour constancy. However, once a workable analog was found, the previous work on fluorescent surfaces, and metallic and non-metallic specularities, made it trivial to provide versions of the method for these more general circumstances. This is very promising, because if statistical models for the occurrence of these phenomena in real images could be found, this algorithm would be able to make use of that information, and would then be combining

more sources of information than any other algorithm. Thus potentially, this algorithm is the most powerful one to date. Indeed, the results with generated data were excellent, and it seems that this algorithm is the single most effective algorithm on synthetic data.

Obtaining good results with real images proved to be somewhat more difficult. I found that with real images, the algorithm is more sensitive to certain tuning parameters than it is in the synthetic case. However, I was able to find parameters which provide real image performance approaching that of the best algorithms, and substantially better than the chromaticity versions. Given that this method is still relatively unexplored, it seems likely that it can be further improved so that the real data results are in line with the generated data results.

I turn now to a different aspect of the work presented in the thesis. Although many of the experiments address colour constancy performance without reference to its possible applications, it is fair to ask: "How good is computational colour constancy?" Specifically, how do the performance figures relate to some real world task? To answer this question, I used colour indexing performance as a metric for colour constancy . Indexing can be used as measure indicative of colour based object recognition performance, which, unlike indexing, seems to required generalized colour constancy. In this work, I investigated the relationship between colour constancy performance and indexing performance. I found that as colour constancy improves, so does indexing, but that current colour constancy methods are far from ideal for this task. In short, the existing error in colour constancy algorithms is still significant for object recognition, and should be addressed in further research.

Having come this far in the exposition, the reader may rightfully ask, "What algorithm should be used?" Unfortunately, the exact answer depends on what the world is like, or what the application domain is. We still do not have a good understanding of the statistics of the world. For example, it is difficult to say what kind of world a consumer digital camera will be exposed to. However, it is interesting that the best algorithm over the 529 images used for the bulk of this work is three-dimensional gamut-mapping combined with the proposed modifications. Specifically, this final algorithm included

the extensions for fluorescent surfaces and non-metallic specularities, together with the new method of choosing the solution, with a similar algorithm with the extension for metallic specularities being not far behind. Although this algorithm may not always be the best performer, it is likely to give reasonable performance over a wide range of circumstances.

# Bibliography

- 1. J. J. McCann, S. P. McKee, and T. H. Taylor, Quantitative Studies in Retinex Theory, Vision Research, 16, pp. 445-458 (1976).
- 2. D. A. Brainard and B. A. Wandell, Asymmetric color matching: how color appearance depends on the illuminant, *Journal of the Optical Society of America A*, **9**, pp. 1433-1448 (1992).
- 3. D. H. Brainard, W. A. Brunt, and J. M. Speigle, Color constancy in the nearly natural image. I. Asymmetric matches, *Journal of the Optical Society of America A*, **14**, pp. 2091-2110 (1997).
- 4. M. Lucassen, Quantitative Studies of Color Constancy: Utrecht University, 1993.
- 5. D. Marr, Vision: Freeman, 1982.
- G. Sharma and H. J. Trussell, Characterization of Scanner Sensitivity, Proc. IS&T and SID's Color Imaging Conference: Transforms & Transportability of Color, pp. 103-107 (1993).
- 7. G. E. Healey and R. Kondepudy, Radiometric CCD camera calibration and noise estimation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**, pp. 267-276 (1994).
- 8. K. Barnard, Computational colour constancy: taking theory into practice, : Simon Fraser University, School of Computing (1995).
- 9. P. L. Vora, J. E. Farrell, J. D. Tietz, and D. H. Brainard, Digital color cameras--Response models, (1997), available from http://color.psych.ucsb.edu/hyperspectral/..
- 10. P. L. Vora, J. E. Farrell, J. D. Tietz, and D. H. Brainard, Digital color cameras--Spectral response, (1997), available from http://color.psych.ucsb.edu/hyperspectral/..
- 11. G. Wyszecki and W. S. Stiles, Color Science: Concepts and Methods, Quantitative Data and Formulas, 2 ed. New York: Wiley, 1982.
- 12. B. K. P. Horn, *Robot Vision*: MIT Press, 1986.
- 13. H. Lee, E. J. Breneman, and C. P. Schulte, Modeling Light Reflection for Computer Color Vision, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **4**, pp. 402-409 (1990).
- 14. J. M. Speigle and D. H. Brainard, Luminosity thresholds: effects of test chromaticity and ambient illumination, *Journal of the Optical Society of America A*, **13**, pp. 436-451 (1996).
- B. A. Maxwell and S. A. Shafer, Physics-based segmentation of complex objects using multiple hypotheses of image formation, *Computer Vision and Image Understanding*, 65, pp. 269-295 (1997).

- K. J. Dana, B. v. Ginneken, S. K. Nayar, and J. J. Koenderink, Reflectance and texture of real-work surfaces, *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 151-157 (1997).
- 17. N. F. E. Nicodemus, J. C. Richmond, J. H. Hsia, I. W. Ginsberg, and T. Limperis, *Geometrical considerations an nomenclature for reflectance*, 1977.
- 18. S. A. Shafer, Using color to separate reflection components, *COLOR Research and Application*, **10**, pp. 210-218 (1985).
- 19. G. Healey, Using color for geometry-insensitive segmentation, *Journal of the Optical Society of America A*, **6**, pp. 920-937 (1989).
- 20. S. Tominaga and B. A. Wandell, Standard surface-reflectance model and illuminant estimation, *Journal of the Optical Society of America A*, **6**, pp. 576-584 (1989).
- 21. S. Tominaga, Dichromatic Reflection Models for a Variety of Materials, *COLOR Research and Application*, **19**, pp. 277-285 (1994).
- 22. L. B. Wolff, Generalizing Lambert's Law for Smooth Surfaces, *Proc. 4th European Conference on Computer Vision*, pp. II:40-53 (1996).
- M. Oren and S. K. Nayar, Seeing beyond Lambert's law, International Journal of Computer Vision, 14, pp. 227-251 (1995).
- 24. S. K. Nayar, K. Ikeuchi, and T. Kanade, Surface reflection: physical and geometric perspectives, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**, pp. 611-634 (1991).
- 25. J. Cohen, Dependency of The Spectral Reflectance Curves of The Munsell Color Chips, *Psychonomic Science*, **1**, pp. 369-370 (1964).
- 26. L. T. Maloney, Evaluation of linear models of surface spectral reflectance with small numbers of parameters, *Journal of the Optical Society of America A*, **3**, pp. 1673-1683 (1986).
- 27. J. P. S. Parkkinen, J. Hallikainen, and T. Jaaskelainen, Characteristic spectra of Munsell Colors, *Journal of the Optical Society of America A*, **6**, pp. 318-322 (1989).
- 28. M. J. Vrhel, R. Gershon, and L. S. Iwan, Measurement and Analysis of Object Reflectance Spectra, *COLOR Research and Application*, **19**, pp. 4-9 (1994).
- D. B. Judd, D. L. MacAdam, and G. Wyszecki, Spectral Distribution of Typical Daylight as a Function of Correlated Color Temperature, *Journal of the Optical Society of America*, 54, pp. 1031-1040 (1964).
- 30. E. R. Dixon, Spectral distribution of Australian daylight, *Journal of the Optical Society* of America, **68**, pp. 437-450 (1978).
- 31. D. H. Marimont and B. A. Wandell, Linear models of surface and illuminant spectra, *Journal of the Optical Society of America A*, **9**, pp. 1905-1913 (1992).
- 32. D. Forsyth, A novel algorithm for color constancy, *International Journal of Computer Vision*, **5**, pp. 5-36 (1990).
- 33. J. v. Kries, Beitrag zur Physiologie der Gesichtsempfinding, Arch. Anat. Physiol., 2, pp. 5050-5524 (1878).

- 34. G. West and M. H. Brill, Necessary and sufficient conditions for von Kries chromatic adaptation to give colour constancy, *J. Math. Biol.*, **15**, pp. 249-258 (1982).
- 35. J. A. Worthey, Limitations of color constancy, *Journal of the Optical Society of America* [Suppl.], **2**, pp. 1014-1026 (1985).
- 36. J. A. Worthey and M. H. Brill, Heuristic analysis of von Kries color constancy, *Journal of the Optical Society of America A*, **3**, pp. 1708-1712 (1986).
- 37. G. D. Finlayson, Coefficient Color Constancy, : Simon Fraser University, School of Computing (1995).
- G. D. Finlayson, M. S. Drew, and B. V. Funt, Spectral Sharpening: Sensor Transformations for Improved Color Constancy, *Journal of the Optical Society of America A*, **11**, pp. 1553-1563 (1994).
- 39. G. D. Finlayson, M. S. Drew, and B. V. Funt, Color Constancy: Generalized Diagonal Transforms Suffice, *Journal of the Optical Society of America A*, **11**, pp. 3011-3020 (1994).
- 40. G. D. Finlayson, Color in perspective, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18**, pp. 1034-1038 (1996).
- 41. M. J. Swain and D. H. Ballard, Color Indexing, *International Journal of Computer Vision*, 7, pp. 11-32 (1991).
- 42. G. Buchsbaum, A spatial processor model for object colour perception, *Journal of the Franklin Institute*, **310**, pp. 1-26 (1980).
- R. Gershon, A. D. Jepson, and J. K. Tsotsos, From [R, G, B] to Surface Reflectance: Computing Color Constant Descriptors in Images, *Perception*, pp. 755-758 (1988).
- 44. E. H. Land and J. J. McCann, Lightness and Retinex theory, *Journal of the Optical Society of America*, **61**, pp. 1-11 (1971).
- 45. E. H. Land, The Retinex theory of Color Vision, *Scientific American*, **237**, pp. 108-129 (1977).
- 46. E. H. Land, Recent advances in Retinex theory and some implications for cortical computations: Color vision and the natural image, *Proceedings of the National Academy of Science*, **80**, pp. 5163-5169 (1983).
- 47. E. H. Land, Recent advances in Retinex theory, Vision Research, 26, pp. 7-21 (1986).
- 48. E. H. Land, An alternative technique for the computation of the designator in the Retinex theory of color vision, *Proceedings of the National Academy of Science*, **83**, pp. 3078-3080 (1986).
- 49. B. K. P. Horn, Determining lightness from an image, *Computer Vision, Graphics, and Image Processing*, **3**, pp. 277-299 (1974).
- 50. A. Blake, Boundary conditions for lightness computation in Mondrian world, *Computer Vision, Graphics, and Image Processing*, **32**, pp. 314-327 (1985).
- 51. A. Hurlbert, Formal connections between lightness algorithms, *Journal of the Optical Society of America A*, **3**, pp. 1684-1692 (1986).
- 52. D. A. Brainard and B. A. Wandell, Analysis of the Retinex theory of Color Vision, *Journal of the Optical Society of America A*, **3**, pp. 1651-1661 (1986).

- 53. B. V. Funt, M. S. Drew, and M. Brockington, Recovering Shading from Color Images, *Proc.* Second European Conference on Computer Vision, pp. 124-132 (1992).
- 54. J. J. McCann, Magnitude of color shifts from average quanta catch adaptation, *Proc. IS&T/SID Fifth Color Imaging Conference: Color Science, Systems and Applications*, pp. 215-220 (1997).
- 55. A. Moore, J. Allman, and R. M. Goodman, A Real-Time Neural System for Color Constancy, *IEEE Transactions on Neural networks*, **2**, pp. 237-247 (1991).
- 56. L. T. Maloney and B. A. Wandell, Color constancy: a method for recovering surface spectral reflectance, *Journal of the Optical Society of America A*, **3**, pp. 29-33 (1986).
- 57. B. A. Wandell, The synthesis and analysis of color images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**, pp. 2-13 (1987).
- 58. G. D. Finlayson, B. V. Funt, and K. Barnard, Color Constancy Under Varying Illumination, *Proc. Fifth International Conference on Computer Vision*, pp. 720-725 (1995).
- 59. D. H. Brainard and W. T. Freeman, Bayesian color constancy, *Journal of the Optical Society of America A*, **14**, pp. 1393-1411 (1997).
- 60. M. D'Zmura and G. Iverson, Color constancy. I. basic theory of two-stage linear recovery of spectral descriptions for lights and surfaces, *Journal of the Optical Society of America A*, **10**, pp. 2148-2165 (1993).
- 61. G. Finlayson and S. Hordley, A theory of selection for gamut mapping colour constancy, *Proc. IEEE Conference on Computer Vision and Pattern Recognition* (1998).
- 62. G. D. Finlayson, P. H. Hubel, and S. Hordley, Color by Correlation, *Proc. IS&T/SID Fifth Color Imaging Conference: Color Science, Systems and Applications*, pp. 6-11 (1997).
- 63. B. Funt, V. Cardei, and K. Barnard, Learning Color Constancy, *Proc. IS&T/SID Fourth Color Imaging Conference: Color Science, Systems and Applications*, pp. 58-60 (1996).
- 64. B. Funt, V. Cardei, and K. Barnard, Neural network color constancy and specularly reflecting surfaces, *Proc. AIC Color* 97 (1997).
- 65. V. Cardei, B. Funt, and K. Barnard, Modeling color constancy with neural networks, *Proc. International Conference on Vision Recognition, Action: Neural Models of Mind and Machine* (1997).
- 66. V. Cardei, B. Funt, and K. Barnard, Adaptive Illuminant Estimation Using Neural Networks, *Proc. International Conference on Artificial Neural Networks* (1998).
- 67. B. V. Funt, V. C. Cardei, and K. Barnard, Method of estimating chromaticity of illumination using neural networks, in *United States Patent 5*,907,629. United States (1999).
- 68. H.-C. Lee, Method for determining the color of a scene illuminant from a color image, in *US Patent* 4,685,071 (1986).
- 69. H.-C. Lee, Method for computing the scene-illuminant chromaticity from specular highlights, *Journal of the Optical Society of America A*, **3**, pp. 1964-1699 (1986).
- 70. M. D'Zmura and P. Lennie, Mechanisms of color constancy, *Journal of the Optical Society* of America A, 3, pp. 1662-1672 (1986).

- 71. S. Tominaga and B. A. Wandell, Component estimation of surface spectral reflectance, *Journal of the Optical Society of America A*, **7**, pp. 312-317 (1990).
- S. Tominaga, Realization of color constancy using the dichromatic reflection model, Proc. IS&T/SID 2nd. Color Imaging Conference: Color Science, Systems and Applications, pp. 37-40 (1994).
- 73. W. M. Richard, Automated detection of effective scene illuminant chromaticity from specular highlights in digital images, : Center for Imaging Science, Rochester Institute of Technology (1995).
- 74. G. J. Klinker, S. A. Shafer, and T. Kanade, Using a color reflection model to separate highlights from object color, *Proc. First International Conference on Computer Vision*, pp. 145-150 (1987).
- 75. R. Gershon, A. D. Jepson, and J. K. Tsotsos, Highlight identification using chromatic information, *Proc. First International Conference on Computer Vision*, pp. 161-170 (1987).
- 76. G. J. Klinker, S. A. Shafer, and T. Kanade, A physical approach to color image understanding, *International Journal of Computer Vision*, **4**, pp. 7-38 (1990).
- 77. C. Novak and S. Shafer, Anatomy of a color histogram, *Computer Vision and Pattern Recognition*, pp. 599-605 (1992).
- 78. C. Novak and S. Shafer, Estimating scene properties from color histograms, (1992), available from
- 79. S. K. Nayar and R. M. Bolle, Computing Reflectance Ratios from an Image, *Pattern Recognition*, 7(1993).
- 80. H. Lee, Illuminant Color from shading, *Proc. Perceiving, Measuring and Using Color* (1990).
- M. S. Drew, Outlier Detection and Physical Model in the Orientation-from-Color Problem, Proc. Workshop on Performance vs. Methodology in Computer Vision, pp. 124-133 (1994).
- 82. M. Tsukada and Y. Ohta, An Approach to Color Constancy Using Multiple Images, *Proc. Third International Conference on Computer Vision*, pp. 385-389 (1990).
- 83. Y. Ohta and Y. Hayashi, Recovery of illuminant and surface colors from images based on the CIE daylight, *Proc. Third European Conference on Computer Vision*, pp. II:235-245 (1994).
- 84. K. Barnard, G. Finlayson, and B. Funt, Colour constancy for scenes with varying illumination, *Computer Vision and Image Understanding*, **65**, pp. 311-321 (1997).
- 85. B. V. Funt, M. S. Drew, and J. Ho, Color constancy from mutual reflection, *International Journal of Computer Vision*, **6**, pp. 5-24 (1991).
- 86. B. V. Funt and M. S. Drew, Color space analysis of mutual illumination, *IEEE Transactions* on *Pattern Analysis and Machine Intelligence*, **15**, pp. 1319-1326 (1993).
- 87. J. Matas, R. Marik, and J. Kittler, Illumination Invariant Colour Recognition, *Proc. 5th British Machine Vision Conference* (1994).
- 88. S. D. Buluswar and B. A. Draper, Color recognition in outdoor images, *Proc. Sixth International Conference on Computer Vision*, pp. 171-177 (1998).

- 89. J. Matas, R. Marik, and J. Kittler, On representation and matching of multi-coloured objects, *Proc. Fifth International Conference on Computer Vision*, pp. 726-732 (1995).
- 90. S. K. Nayar and R. M. Bolle, Reflectance Based Object Recognition, (1992), available from
- 91. J. Matas, Colour-based Object Recognition, : University of Surrey (1996).
- 92. B. V. Funt and G. D. Finlayson, Color Constant Color Indexing, Simon Fraser University School of Computing Science (1991), available from
- 93. G. D. Finlayson, Colour Object Recognition, : Simon Fraser University, School of Computing (1992).
- 94. G. D. Finlayson, S. S. Chatterjee, and B. V. Funt, Color Angular Indexing, *Proc. 4th European Conference on Computer Vision*, pp. II:16-27 (1996).
- 95. M. S. Drew, J. Wei, and Z.-N. Li, Illumination-Invariant Color Object recognition via Compressed Chromaticity Histograms of Normalized Images, *Proc. Sixth International Conference on Computer Vision*, pp. 533-540 (1998).
- 96. G. D. Finlayson, B. Schiele, and J. L. Crowley, Comprehensive colour image normalization, *Proc. 5th European Conference on Computer Vision*, pp. I:475-490 (1998).
- 97. P. M. Hubel, D. Sherman, and J. E. Farrell, A comparison of method of sensor spectral sensitivity estimation, *Proc. IS&T/SID 2nd. Color Imaging Conference: Color Science, Systems and Applications*, pp. 45-48 (1994).
- 98. W. K. Pratt and C. E. Mancill, Spectral estimation techniques for the spectral calibration of a color image scanner, *Applied Optics*, **15**, pp. 73-75 (1976).
- 99. G. Finlayson, S. Hordley, and P. Hubel, Recovering device sensitivities with quadratic programming, *Proc. IS&T/SID Sixth Color Imaging Conference: Color Science, Systems and Applications*, pp. 90-95 (1998).
- 100. G. C. Holst, CCD Arrays, Cameras, and Displays, 2 ed. Bellingham, WA: SPIE Press, 1998.
- G. D. Finlayson, M. S. Drew, and B. V. Funt, Enhancing von Kries Adaptation via Sensor Transformations, *Proc. Human Vision, Visual Processing, and Digital Display IV*, pp. 473-484 (1993).
- 102. U. Seger, G. H.-G. Graf, and M. E. Landraf, Vision assistance in scenes with extreme contrast, *IEEE Micro*, **13**, pp. 50-56 (1993).
- B. V. Funt, M. Brockington, and F. Tong, Conformal transplantation of lightness to varying resolution sensors, *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 563-569 (1993).
- 104. K. Barnard, G. Finlayson, and B. Funt, Colour constancy for scenes with varying illumination, *Proc. 4th European Conference on Computer Vision*, pp. II:1-15 (1996).
- 105. B. Funt and V. C. Cardei, Bootstrapping Color Constancy, Proc. Human Vision and Electronic Imaging IV, pp. 421-428 (1999).
- 106. E. L. Krinov, Spectral Reflectance Properties of Natural Formations: National Research Council of Canada, 1947.

- 107. B. Funt, K. Barnard, and L. Martin, Is Colour Constancy Good Enough?, *Proc. 5th European Conference on Computer Vision*, pp. I:445-459 (1998).
- 108. J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation*: Addison-Wesley Publishing Company, 1991.
- 109. G. Sharma and H. J. Trussel, Set theoretic estimation in color scanner characterization, *Journal of Electronic Imaging*, **5**, pp. 479-489 (1996).
- 110. J. L. Devore, *Probability and Statistics for Engineering and the Sciences*. Monterey, CA: Brooks/Cole Publishing Company, 1982.
- 111. G. Finlayson and S. Hordley, Selection for gamut mapping colour constancy, *Proc. British Machine Vision Conference* (1997).
- 112. G. Finlayson and M. Drew, Positive bradford curves through sharpening, *Proc. IS&T/SID* Seventh Color Imaging Conference: Color Science, Systems and Applications, pp. To appear (1999).
- 113. P. Emmel and R. D. Hersch, Spectral colour prediction model for a transparent fluorescent ink on paper, *Proc. IS&T/SID Sixth Color Imaging Conference: Color Science, Systems and Applications*, pp. 116-122 (1998).
- 114. B. Maxwell, Segmentation and interpretation using multiple physical hypotheses of image formation, : CMU (1996).
- 115. B. A. Maxwell and S. A. Shafer, Physics-based segmentation: moving beyond color, *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 742-749 (1996).
- 116. B. V. Funt and G. D. Finlayson, Color Constant Color Indexing, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **17**(1995).
- G. Healey and D. Slater, Global color constancy: recognition of objects by use of illumination invariant properties of color distributions, *Journal of the Optical Society of America A*, **11**, pp. 3003-2010 (1994).