

Designing a Colour Filter for Making Cameras more Colorimetric

by
Yuteng Zhu

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
University of East Anglia
School of Computing Sciences

September 2021

©This copy of the thesis has been supplied on condition that anyone who consults it is understood to recognise that its copyright rests with the author and that use of any information derived therefrom must be in accordance with current UK Copyright Law. In addition, any quotation or extract must include full attribution.

Abstract

If a camera were to capture colour like a human observer, fundamentally, it should sense the light information as the way the human visual system does. It is necessary to either replicate the human visual sensitivity responses or reproduce the three-number colour representations - e.g. CIE XYZ tristimulus values - to obtain an accurate colour measurement. In practice, however, the camera sensors generally deviate from the ideal sensitivities of the human visual system. Consequently, the colour triplets a camera records are device-dependent, which generally differ from the standard observer tristimulus values. The colorimetric performance can be improved by either correcting camera responses to the reference ground-truth values using sophisticated mathematical transformations or using more imaging sensors/filters to capture more information about the incident light. These methods have their disadvantages: the former increases the computational complexity and the latter increases the system complexity and the overall cost.

In this thesis, we aim to make the digital camera capture colours more like the human visual perception by placing a colour filter in front of the camera so as to alter its spectral sensitivity functions as desired. The central contribution of this study is to carefully design a colour filter for a given camera so that the ‘filter+camera’ setting having the new sensitivities becomes almost colorimetric, i.e. recording the colour triplets that can be linearly transformed to the ground-truth XYZ tristimulus values.

The starting point for this thesis is to design the filter that makes the filtered camera best achieve the Luther condition, i.e. the new effective camera sensitivity functions after filtering are a linear combination of the colour matching function of the human visual system. Under this condition, the camera can capture any incoming colour signal accurately in the sense that the captured RGBs are almost a linear transform from the XYZ tristimuli.

Next, we reformulate the problem formulation for finding the optimal filter that targets the more generalised Vora-Value goodness measure. The Vora-Value, by definition, measures the similarity between the vector spaces spanned by the spectral sensitivities of a camera and the XYZ colour matching functions underpinning the human visual system. The Vora-Value has the advantage that the best filter is related to the target human visual space and not fixed coordinates (e.g. the XYZ and RGB colour matching functions have different coordinate values but are in the same vector space).

As well as developing a method that finding a filter maximises the Vora-Value (makes the vector spaces most similar), we examine the relationship between the Vora-Value

and Luther condition optimisations. We show that the Luther-condition optimisation also maximises the Vora-Value if we find the filter that makes a linear combination of the camera sensitivities most similar to a linear transform of XYZ (that is orthonormal). This is an important result as the Luther optimisation is much simpler to implement and faster to execute. So we can use the simpler Luther-condition formulation to maximise the Vora-Value measure using a more straightforward algorithm.

A strength and weakness of the Luther and Vora-Vora optimisations is that they assume - as an explicit part of their formulations - that all spectra are equally likely. But, this is not the case in real imaging applications. So we extend our filter design algorithms in a data-driven manner that it optimises for the best colorimetric estimates given a collection of illuminants and surface reflectance data. Our extended method uses quadratic programming that allows us to add linear inequality constraints into the problem formulation. We show how to find filters that have smooth distribution and bounded transmittance (e.g. transmit at least 50% of the light) across the spectrum. Constraints like these make the filters more useful and feasible could make the filters easier to manufacture. We show that we can find smooth and highly transmissive colour filters that when placed in front of a digital camera can make the camera significantly more colorimetric and hence can be used for colour measurement applications with high demand in colour accuracy.

Access Condition and Agreement

Each deposit in UEA Digital Repository is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the Data Collections is not permitted, except that material may be duplicated by you for your research use or for educational purposes in electronic or print form. You must obtain permission from the copyright holder, usually the author, for any other use. Exceptions only apply where a deposit may be explicitly provided under a stated licence, such as a Creative Commons licence or Open Government licence.

Electronic or print copies may not be offered, whether for sale or otherwise to anyone, unless explicitly stated under a Creative Commons or Open Government license. Unauthorised reproduction, editing or reformatting for resale purposes is explicitly prohibited (except where approved by the copyright holder themselves) and UEA reserves the right to take immediate 'take down' action on behalf of the copyright and/or rights holder if this Access condition of the UEA Digital Repository is breached. Any material in this database has been supplied on the understanding that it is copyright material and that no quotation from the material may be published without proper acknowledgement.

Acknowledgements

My first and foremost thanks must go to my supervisor, Prof. Graham D. Finlayson, for his guidance and support throughout my PhD journey. His ability and vision to apply mathematical ideas into the imaging system have been inspirational to me. He has shown me the art of research and encouraged me to be an independent researcher. I would also like to thank Prof. Richard Harvey for his supervision and helpful suggestions.

I would like to thank my examiners, Dr. Michael Brill and Dr. Huw Owens for their insightful comments and feedback, which this thesis has greatly benefited from. Thanks also go to Dr. Michal Mackiewicz for being my viva chair and offering many helps during my study.

I would like to express a special thanks to Dr. Javier Vazquez-Corral for being a mentor and as well as a good friend. His helpful discussions and friendly talks are invaluable and most encouraging. Thanks also go to Dr. Gong Han for sharing me his knowledge of computer vision.

Special thanks go to Ellen Bowler for being such a good friend with many memorable moments and lots of help in proof-readings. Thanks also go to my good friends in the Colour and Imaging Lab: Seth Nixon, Fufu Fang, Ghalia Hemrit, Yi-Tun Lin, Jake McVey, Chloe Game, and Hang Zhou.

My immense gratitude goes to Prof. Ronnier M. Luo, a mentor who led me into the field of Colour Science, encouraged me to pursuit a doctoral degree, sets the best role model of being a researcher, and is always there to help. I would also like to thank Dr. Kaida Xiao and Dr. Michael Pointer for many helpful discussions and advises.

A heartfelt thank goes to my parents and family who always give me their unconditional love and support. Finally, I dedicate the greatest thanks to my boyfriend, Xingxing, for his understanding and support throughout my PhD years.

Contents

Abstract	i
Acknowledgements	iii
List of Figures	vii
List of Tables	x
1 Introduction	1
1.1 Problem Statement	1
1.2 Proposed Approach	3
1.3 Thesis Outline	6
2 Literature Review	8
2.1 Colorimetry	8
2.1.1 The Human Vision System	8
2.1.2 The Colour Matching Experiments	10
2.1.3 CIE Colorimetric System	12
2.2 Image Colour Formation	18
2.3 The Data Sets	19
2.3.1 Camera Spectral Sensitivity Functions	19
2.3.2 Reference Colour Matching Functions	20
2.3.3 Illuminant and Object Spectra	20
2.4 Colour Correction	20
2.4.1 Colour Correction Transformation	20
2.4.2 Linear Colour Correction	21
2.4.3 Calculating Colour Error	22
2.5 Measures of the Goodness of a Set of Spectral Sensitivities	23
2.6 Filter Design in Colour Imaging	25
2.7 Summary	28
3 Luther Condition Filter Design	29
3.1 Introduction	29
3.2 The Luther Condition	32
3.3 Luther-Condition Filter Optimisation	32
3.3.1 Algorithm for Solving the Luther Optimisation	33
3.3.2 Filter Positiveness	34

3.4	Results	35
3.4.1	Luther-filters	35
3.4.2	Evaluation of Spectral Fitting	37
3.4.3	Colour Measurement Experiment	39
3.4.3.1	Single-light Case	40
3.4.3.2	Multiple-light Case	41
3.4.3.3	Multiple Cameras	41
3.5	Conclusion	43
4	Vora-Value Filter Design and its Relation to Luther Condition Filter Optimisation	44
4.1	Introduction	44
4.2	Background	47
4.2.1	The Luther Condition under Linear Transform	47
4.2.2	The Vora-Value	48
4.2.3	Orthonormal Basis	49
4.3	Vora-Value Filter Optimisation	50
4.3.1	Gradient Ascent for Solving the Optimisation	50
4.3.2	Newton's Method for Solving the Optimisation	52
4.4	The Modified-Luther Condition Optimisation	53
4.4.1	Modified Luther Condition Filter Design	53
4.4.2	Equivalence of the Modified-Luther and Vora-Value Optimisations	55
4.4.3	Gradient and Hessian for the Regularised Optimisation	57
4.5	Results	57
4.5.1	Vora-filters	57
4.5.2	Comparison of Algorithm Performance	58
4.5.3	Vora-Values and Colour Measurement Experiment	59
4.6	Conclusion	64
5	Data-Driven Filter Design	65
5.1	Introduction	65
5.2	The Data-driven Luther Condition	67
5.3	Data-driven Filter Optimisation	68
5.3.1	ALS Algorithm for Solving the Optimisation	69
5.4	Results	70
5.4.1	Data-filters	70
5.4.2	Colour Measurement Experiments	71
5.4.2.1	Single Light Case	72
5.4.2.2	Multiple Lights Case	72
5.4.2.3	Multiple Cameras	74
5.5	Conclusion	74
6	Filter Constraints	75
6.1	Introduction	75
6.2	Adding Filter Constraints	77
6.2.1	Smoothness Constraint	77
6.2.2	Transmittance Constraint	78

6.3	Filter Optimisation with Constraints	79
6.3.1	Constrained Luther-Condition Optimisation	79
6.3.1.1	Solving for Constrained Luther-filters	79
6.3.2	Constrained Vora-Value Optimisation	81
6.3.2.1	Solving for Constrained Vora-filters	81
6.3.3	Constrained Data-driven Optimisation	82
6.3.3.1	Solving for Constrained Data-filters	82
6.3.4	Multi-initialisation Conditions	83
6.4	Results	84
6.4.1	Constrained Luther-filters	84
6.4.2	Constrained Vora-filters	86
6.4.3	Constrained Data-filters	88
6.4.4	Sampled-based Optimisation	88
6.4.4.1	High Transmissive Data-filters	90
6.4.4.2	Sampled Initialisation	91
6.4.5	Colour Filtering vs Higher-order Correction Methods	92
6.5	Discussion	94
6.6	Conclusion	94
7	Conclusion	96
7.1	Contributions	96
7.2	Future Work	97
7.3	Publications	99
A	Algorithm Implementation	101
A.1	Algorithm for the Luther-condition Optimisation	101
A.2	Algorithms for the Vora-Value Filter Optimisation	101
A.2.1	Derivation of Gradient	102
A.2.2	Relation of ALS and Gradient Algorithms	104
A.2.3	Derivation of Hessian Matrix	104
A.2.4	Positive Definite of the Hessian Matrix	106
A.3	Algorithm for the Data-driven Optimisation	107
A.4	Algorithm for the Filter Constraints	108
	Bibliography	110

List of Figures

1.1	A typical colour measurement scenario: the light emitting from the light source reflected from a colour chart before entering into the sensor system – either by an observer or a camera. Conventionally camera RGB values are mapped to the reference CIE XYZ values in the colour correction process where a 3×3 transform matrix \mathbf{M} is used.	2
1.2	The proposal of applying a prefilter for a digital camera such that its filtered RGBs can be best mapped to the reference XYZ tristimulus values using a 3×3 transform matrix.	4
2.1	Electromagnetic of light with enlarged visible spectrum. The image is obtained from https://en.wikipedia.org/wiki/Light	9
2.2	Section of human eye with a schematic enlargement of the retina. The cones and rods receptors are depicted in different shapes and colours on the right panel. This image is obtained from http://www.closerlookatstemcells.org/stem-cells-and-medicine/macular-degeneration	9
2.3	Schematic diagram of trichromatic colour matching experiment by additive mixture of lights. The intensity of the three primary Red, Green, and Blue lights can be adjusted to visually match a test light. On the right, we show the RGB colour matching functions for the CIE 1931 Standard Observer.	10
2.4	The CIE XYZ colour matching functions of 1931 Standard Observer (in solid lines) and of 1964 Standard Observer (in dashed lines).	13
2.5	Calculation of XYZ tristimulus values given the spectral information of the illuminant, the surface and the observer.	14
2.6	Relative spectral power distributions of CIE Standard Illuminants A, D50, D55, D65, D75, and E.	17
2.7	For measuring the object colours, two images are taken, once with and once without a colour filter placed in front of the camera. The colour values combined from two images are mapped to the reference XYZ tristimuli using a 6×3 correction matrix.	25
3.1	The use of a digital camera with a colour filter placed between the scene and the camera in a colour matching experiment. The goal is to find a prefilter such that the RGBs a camera measures (after a 3×3 correction matrix) are the same as the colour mixtures a human observer would give to make a match.	30

3.2	(A) shows the spectral sensitivities of a Canon 40D camera. (B) shows the least-squares fit of the camera curves (in dashed lines) to XYZ colour matching functions (in solid lines). With a specially designed colour filter shown in (C), the least-squares fit of the camera spectral sensitivity curves multiplied by the filter - shown in (D) - is much closer to the XYZ colour matching functions.	36
3.3	Correlations between (A) NRMSE and Vora-Value and (B) Vora-Value and mean ΔE_{ab}^*	38
3.4	Spectral match to the Luther condition by unfiltered NAT ive camera sensitivities and filtered LUTH er-condition optimised sensitivities for a group of 28 cameras in terms of Vora-Value. The overall average Vora-Values of the camera group for both conditions are shown in the green box.	39
3.5	(A) Mean and (B) 95-percentile ΔE_{ab}^* errors for 28 cameras. The grey-bars show the colour errors for NAT ive colour correction. The green lines with red diamonds show the results by using the LUTH er-condition optimised filters.	42
4.1	A visualisation of the vector spaces of camera (in red), filtered camera (in green), and the human visual system (in blue). A colour stimulus is projected to each of the space. We aim for the filtered camera space to be as closely as possible to the human visual space such that the colour it records can better predict that of the human visual response.	45
4.2	Three sensitivity sets: cone fundamentals, a set of orthonormal basis, and the CIE 1931 XYZ colour matching functions. These sensitivity sets are all linearly related and span the same Human Visual Space.	49
4.3	(A) Spectral sensitivities of Canon 40D camera and the optimised filters solved from the algorithms: (B) Alternating Least-Squares (in green), (C) Gradient Ascent (in blue) and (D) Newton's method (in red).	59
4.4	(A) Spectral sensitivities of Nikon D80 camera, the optimised filters solved from the algorithms: (B) Alternating Least-Squares (in green), (C) Gradient Ascent (in blue) and (D) Newton's method (in red).	60
4.5	Algorithm convergence in terms of (A) Vora-Value and (B) mean ΔE_{ab}^*	61
4.6	Results of Vora-Values for a group of 28 cameras: NAT ive sensitivities without a filter (in neutral), the LUTH er-filters (in blue) and the VORA -filters (in red). The overall averaged Vora-Values of the camera group are given in the last column.	62
4.7	(A) Mean and (B) 95-percentile ΔE_{ab}^* errors for 28 cameras. The grey-bars show the colour errors for NAT ive cameras simply use colour correction without a filter. The results by using the LUTH er-filters and VORA -filters are in dashed green lines and solid red lines respectively.	63
5.1	Schematic diagram of colour measurement for an object viewed under an illuminant given their spectral representations. Given the spectral data of the measured objects, we seek a filter for a camera such that the RGB outputs after a linear mapping become the same as the XYZ tristimulus values.	67
5.2	Spectral transmittance of the optimised Data-drive filters for single illuminant case (A) CIE D65 and (B) CIE A, and (C) multi-illuminant case of 102 illuminants for Canon 40D camera.	71

5.3	Mean (A) and 95-percentile (B) ΔE_{ab}^* errors for 28 cameras. The grey-bars show the colour errors for NAT ive cameras simply use colour correction without a filter. The dashed green lines with black diamonds show the results by using the LUT Her-condition optimised filters. The results of the DATA -driven optimisation are plotted in solid red lines.	73
6.1	An example of smooth filters constructed linearly by using 6-cosine basis and also limited to transmit at least 20% of light at any wavelength, compared to the optimised Data-filter without smoothness constraint. . .	76
6.2	(A) A set of 6-cosine basis functions, (B) a smooth filter example made from linear combinations the basis functions, and (C) a smooth filter made from linear combinations the basis functions and also bounded with min and max transmittances.	78
6.3	Spectral transmittance of Luther-optimised filters. (A) filter with no constraint (in black), filters constructed by 6-, 8-, and 10-term cosine basis functions with a fixed minimum transmittance of 20% (coloured solid lines). (B) filters with varying minimum transmittance of 20%, 30% and 40% composed by 8-cosine basis functions.	85
6.4	Spectral transmittance of optimised Vora-filters for Canon 40D camera. The filter in solid black line is solved with no constraint applied. The filters in red and dashed green lines are modelled under 8-cosine basis functions and constrained by the minimum transmittance of 20% and 30% respectively.	87
6.5	Data-driven optimised filters for Canon 40D camera: (A) filters solved with no smoothness but the min-and-max transmittance constraint, (B) and (C) filters constructed by 6- and 8-cosine basis functions respectively. Dotted purple lines (DATA_luther) show the filters initialised with the Luther-optimised filter, solid red lines (DATA_sampling) show the filters initialised with a sampling filter set. All filters shown are constrained with a fixed minimum transmittance of 20%.	88
6.6	The effect of the number of filters in the initialisation set on the colour reproduction results in terms of mean colour error assuming filters fall within the 8-dimensional space. The magnitude of the bars represents the average colour error from groups of varying the number of filters. The upper one standard deviation error bar is also shown.	92

List of Tables

3.1	Performance of cameras with LUTHer -condition optimised filters versus the NATive sensitivities with no filter applied in terms of Vora-Value and NRMSE (note that higher Vora-Value but lower NRMSE value indicate better performance)	37
3.2	ΔE_{ab}^* statistics of the colour corrected NATive camera and the colour corrected camera with the LUTHer -condition optimised filter for Canon 40D camera under different lighting cases	40
4.1	ΔE_{ab}^* statistics of the colour corrected NATive camera without filter, the colour corrected camera with the optimised LUTHer -filter and VORA -filter for Canon 40D and Nikon D80 cameras.	60
5.1	ΔE_{ab}^* statistics of the colour corrected NATive camera, the colour corrected camera with the LUTHer -condition optimised filter, and the colour corrected camera with the DATA -driven optimised filter for Canon 40D camera under different lighting cases	71
6.1	ΔE_{ab}^* statistics of the colour corrected NATive camera, the colour corrected camera with the LUTHer -condition optimised filter (without constraint), and the colour corrected camera with the LUTHer -condition optimised filters under various constrained conditions for Canon 40D camera.	86
6.2	ΔE_{ab}^* statistics of the colour corrected NATive camera, the colour corrected camera with LUTHer -condition and VORA -Value optimised filters (without constraint), and the colour corrected camera with the LUTHer - and VORA -filters under various constraining conditions for Canon 40D camera	87
6.3	ΔE_{ab}^* statistics of the colour corrected NATive camera, the colour corrected camera with the DATA -driven optimised filter solutions obtained when initialised with Luther -condition filter and Sampling filter set respectively under different constraints for Canon 40D camera	89
6.4	ΔE_{ab}^* statistics and relative exposure settings of the colour corrected camera with the optimised Sampling filter set under varying constraints of the minimum transmittance constructed by 8-cosine basis for Canon 40D camera	91
6.5	ΔE_{ab}^* statistics of the colour corrected camera with the Data-driven optimised filter by Sampling method compared to higher-order colour correction methods for Canon 40D camera	93

Chapter 1

Introduction

1.1 Problem Statement

At the sensor level (in normal lighting), the human eye makes three colour measurements. Specifically, there are long-, medium-, and short-wavelength cone mechanisms that - as the names suggest - have spectral sensitivity functions biased in the respective parts of the visible spectrum [97]. To model the response of the cones, the light - a spectral power distribution - entering the eye is integrated with the spectrum with each of the three cone sensitivities to return three numbers that correlate with the eye's response to the light. Often it is convenient to linearly transform these numbers to different coordinate systems, such as the CIE XYZ which is standardised for colour measurement [65].

In situ, we measure colour using a spot measurement device called a colorimeter [38]. This is a contact device placed over the material of interest. The colorimeter gives the XYZ triplets for the surface. Of course, measuring many colours using a colorimeter is a tedious operation. One has to place the measurement device of different 'spots' and then make a per spot measurement. Thus, there has been a long standing interest in the field to use a digital RGB camera as a measurement device. Here an RGB image is captured of an object or a scene and then the captured RGBs are mapped to the corresponding colour coordinates.

A schematic diagram of the camera colour measurement problem is shown in Figure 1.1. Physically, the colour of an object, when viewed by a human observer, is formed by the light first emerging from the illuminant and reflected from the object before arriving at the eye and stimulating on the retina. As we see, the colour is formed mainly related to three physical factors: the illuminant, the object and the observer. A colour is quantified as if viewed by a CIE standard observer [65] with a well-specified colour

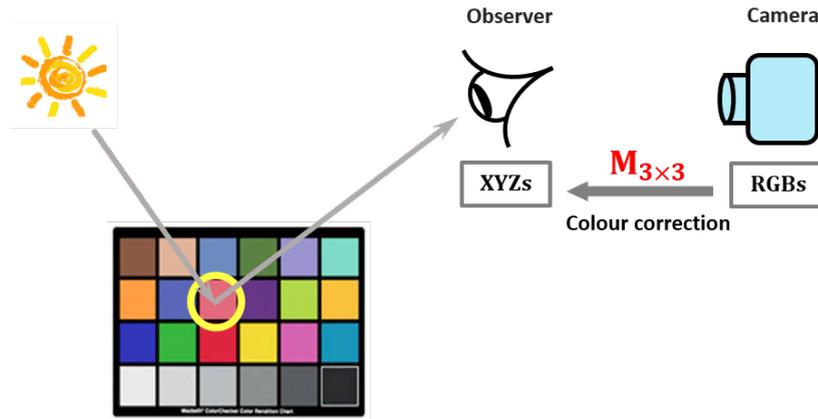


FIGURE 1.1: A typical colour measurement scenario: the light emitting from the light source reflected from a colour chart before entering into the sensor system – either by an observer or a camera. Conventionally camera RGB values are mapped to the reference CIE XYZ values in the colour correction process where a 3×3 transform matrix \mathbf{M} is used.

matching functions (CMFs) related to how the human visual system reacts to colour stimulus (the XYZ CMFs are a linear transform from the cone fundamentals [36]). In such a way, the three-number values of a colour, i.e. CIE XYZ coordinates [65], will be determined.

In Figure 1.1, we show how cameras are used for colour measurement. Here an RGB response is measured by a camera and this is typically colour corrected by a linear transform to approximate the ground-truth XYZ tristimuli (though sometimes non-linear transforms are also used). While in general the approximation is quite good, some mapped colours are far from the actual correct XYZ tristimulus values. One might assume that the solution to this problem would be to ensure that the camera sensitivities were the same as the XYZ CMFs (or a linear transform thereof).

Digital cameras, however, are designed to achieve more than the objective of recording colour; other goals such as keeping noise levels low and cost effectiveness are also important [30, 61, 85]. Additionally, in manufacturing, the feasibility of camera sensors and filters are limited to the choices of the materials and techniques available. Actually, commercial cameras rarely meet the condition and its spectral sensitivity responses are generally far from the desirable cone fundamentals [79]. As a result, camera RGB responses are device-dependent and have no simple relation to the ground-truth human visual colour representations.

Practically, for most applications, e.g. photography and video, it is more important that we can transform the recorded device RGBs to drive a display so that the image captured by a camera either looks the same to a human observer or records triplets of numbers referenced to the human visual system. The colour correction of transforming

device-dependent RGBs to the device-independent XYZ tristimuli is an essential process for better accuracy in respect of colour measurement.

A considerable amount of literature has attempted to establish the best mapping between RGBs and XYZs. Popular methods include various approaches based on least-squares of linear and polynomial regressions [3, 26, 28, 29, 35, 55, 94], look-up-tables [37], and more recently, neural networks [13, 47]. Despite the various methods proposed in the literature, the most widely used method of colour correction is to use a 3×3 matrix transform. This mapping, which we will refer to as linear colour correction (LCC), has some important properties. First, as demonstrated in [21], if object reflectances (the percentage of light the surface reflects at each wavelength) and illuminants spectra can be well described by a 3-dimensional linear model, the mapping from RGB to XYZ has to be a 3×3 matrix and the linear colour correction method provides an exact colour mapping. Another advantage is that the transform scales linearly with the scene radiance changes. Suppose that for a surface in the scene, we have the RGB vector ρ and the XYZ vector χ with the 3×3 linear transform matrix \mathbf{M} . If the scene is made twice as intense, e.g. doubling the intensity of the incoming light, correspondingly we have 2ρ and 2χ . The same matrix \mathbf{M} will correctly map the new camera responses to the XYZs. Finally, the 3×3 matrix method has few parameters - and so is easy to estimate - and it is simple to integrate into image and video processing pipelines.

All colour correction methods fail since RGBs are not linearly related to XYZs. Indeed, the discrepancy between corrected RGBs and their XYZ counterparts can be large, especially for some saturated colours. Colour correction is imperfect because the camera spectral sensitivities are not linearly related to XYZ CMFs nor are surface reflectance spectra sufficiently well described by a 3-dimensional linear model [21].

When the camera sensor sensitivities can be represented as a linear combination of CIE CMFs, which is also termed as the Luther condition [40, 52], we say the camera is fully colorimetric. Under this condition, the RGB responses produced by the camera sensors can be corrected to the reference CIE XYZs precisely without any error; that is to say, a camera can carry out colorimetric measurement.

Our study will investigate how we can use a digital camera to deliver accurate colorimetric measurement while retaining the simplicity of using a linear colour correction.

1.2 Proposed Approach

The idea we propose to solve the problem is to place a specially designed colour filter in front of the camera with the goal of making the filtered camera measurements more

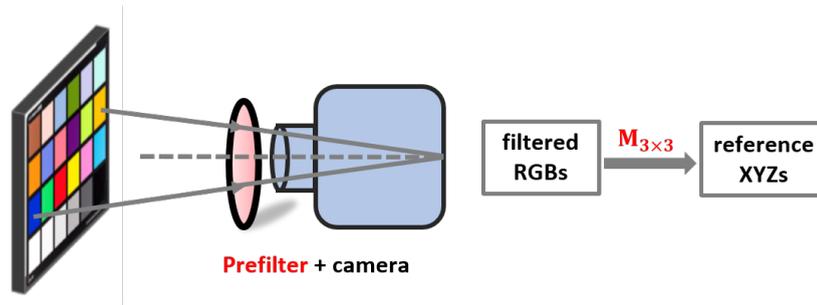


FIGURE 1.2: The proposal of applying a prefilter for a digital camera such that its filtered RGBs can be best mapped to the reference XYZ tristimulus values using a 3×3 transform matrix.

colorimetric. Figure 1.2 illustrates the idea of using a prefilter for a camera such that the filtered RGBs are as linearly related to the target XYZs as possible. Placing a spectrally-selective colour filter in front of a camera will alter the spectra entering the camera; or equivalently, it can be seen as the filter will change the camera responds to the incoming lights. When we place a filter in front of the camera, we multiply the camera’s spectral sensitivities by the filter transmittance to calculate the new effective camera sensitivities.

How we design and solve for the best filter to make a camera colorimetric are the central concern of this thesis. We will start the filter design by looking into the Luther condition, i.e. the camera sensitivities are a linear combination from the CIE XYZ CMFs. We will seek colour filters that best satisfies the Luther condition when placed in front of a camera. We formulate the problem as a mathematical optimisation and develop algorithms to solve for the optimal filter. We will show that with the addition of an optimised filter, the filtered camera can meet the Luther condition to a remarkable degree and thus becomes significantly more colorimetric.

Now, we reformulate the problem formulation so we can find the optimal filter that targets the more generalised Vora-Value goodness measure [88]. The Vora-Value, in effect, measures the similarity between the vector spaces spanned by the spectral sensitivities of a camera and the XYZ colour matching functions underpinning the human visual system. The Vora-Value has the advantage that the best filter is related to the target basis and not fixed coordinates (e.g. the XYZ CMFs) with respect to that basis.

As well as developing a method for finding a filter that optimises the Vora-Value (makes the vector spaces most similar), we examine the relationship between the Vora-Value and Luther condition optimisations. We show that the optimal Luther-filter can simultaneously maximise the Vora-Value if we find the filter that targets at the orthonormal basis of the XYZ CMFs, i.e. the camera sensitivities are most similar to a special linear transform of XYZ CMFs that is orthonormal). This is an important result as the

Luther-optimisation is much simpler to implement and faster to execute. Therefore, we can use the simpler Luther-condition formulation to maximise the Vora-Value measure with better algorithm performance.

The filter design by the Luther-condition is a data-independent method as it makes no prior knowledge about the world but only assumes that any and all spectral stimuli are equally possible to be captured by a camera. Arguably the data-independent method is superior in that the filters it solves for should work in the most general cases. The real-world lights and reflectances, however, do not vary arbitrarily but tend to be smooth [14, 67, 93]. Most naturally occurring spectra can be well described in a linear model with a small number of parameters [15, 23, 56, 57, 99]. In the literature, many studies show that surface reflectance spectra can be adequately approximated with six to eight-dimensional basis. Illuminants, by contrast, are much less describable by small parameter models. Artificial lights such as fluorescent and LED lights can have very spiky peaks with varying peak number and positions. And yet, illuminants are also far from being arbitrary as artificial lights are usually designed to have colours near the Planckian locus [65], a requirement to score highly on colour rendering indices [10]. Also, as for the natural daylight, they can be modelled by a 2-dimensional basis [42].

Given the known features of the real-world spectra, possibly a more useful variant of the Luther condition would be one that is data-driven. That is, we use the spectral data collection which a camera most often encounter in the real world; based on the data, the best mapping can be derived for the camera RGB responses to be transformed to XYZs. With this in mind, we extend our filter design optimisation for making the sensor responses as close to the CIE XYZ tristimulus values as possible, given the knowledge of the real-measured surfaces and illuminants spectra data.

Earlier, we focus on finding the optimal colour filter that makes a digital camera more colorimetric. Our approach is mathematical. We seek the best filter without worrying about whether we can fabricate the filter we design. However, the filters are, necessarily, limited to the fabrication process. Hence, the colour filters in turn are extended to incorporate physical constraints: smoothness and bounded transmittance. The filters are constructed by a group of smooth filter basis and also restricted to have transmittance greater than a minimum threshold. Not only does this allow us to control the optimisation but these filters are plausibly easier to manufacture. The colour filters we solve for are smooth, relatively high transmissive, physically realisable. The validity of these filter design methods and the optimised filters is demonstrated through simulations of colour measurement for a collection of digital cameras.

In this thesis, we aim to improve the colorimetric property of digital cameras by carefully design a spectrally-selective colour filter that can be placed in front of a camera. To briefly summarize, there are mainly six novel contributions in this thesis:

- We design a colour filter that makes an off-the-shelf camera better approximate the Luther condition, i.e. its effective sensitivities are linearly related to the CIE XYZ colour matching functions.
- The filter can be designed given any sensitivity set representing of the human visual system (e.g. cone fundamentals, RGB and XYZ colour matching functions or any linear combination thereof) under the optimisation criteria of the Vora-Value.
- We prove how a filter solved from the Luther condition method can be optimal in terms of Vora-Value by introducing the concept of orthonormal basis vectors.
- The filter is also designed in a data-driven way to better represent the colour measurement of the real scenes where the real world colour spectra are used.
- The physical feasibility of the filters is improved by adding reasonable constraints (smoothness and transmissivity) on the filters and incorporated into the optimisation formulation.
- Significant improvement has been gained by employing a simple sampling method to tackle the issue of local minima in the optimisation.

1.3 Thesis Outline

Chapter 2 presents the background to this thesis, especially the CIE colorimetry for establishing the foundation of the colour measurement. A survey of existing methods for the design and evaluation of filters is also presented.

Chapters 3, 4 and 5 propose different methods for finding optimal colour filters given a digital camera to improve the colour measuring ability. The design of the colour filter is formulated as a parameterised optimisation problem. Chapter 3 sets out to find a filter that best satisfies the Luther condition. It is well known that a camera with spectral sensitivities that are a linear transform from the XYZ CMFs can be used to measure colour without error. With the help of a colour filter, a camera becomes better linearly related to the human visual sensitivities, leading to a significant gain in colour accuracy.

The Vora-Value is a measure of the similarity between the vector spaces spanned by the spectral sensitivities of an RGB camera and the XYZ CMFs. In Chapter 4, we show

that we can maximise the Vora-Value using a gradient ascent procedure. We then go on to consider the relationship between the Luther-condition and Vora-Value optimisations since they return different ‘optimal filters’. We prove an important result. Specifically, if we run the Luther-condition optimisation where we use an orthonormal variant of the XYZs (a linear transform of the XYZ CMFs that are orthonormal), then the Luther-condition optimisation also maximises the Vora-Value. This is practically important since the Luther optimisation is much easier to formulate. That is, we can use the simpler Luther optimisation formulation to maximise the Vora-Value target.

In Chapter 5, we extend our optimisation to find the best colour filter given a corpus of measured lights and surfaces. We consider various scenarios when a data-driven approach might be used, including mapping corresponding RGBs for surfaces across pairs of lights and mapping RGBs measured for multiple lights to a single target light.

In Chapter 6, constraints are placed on the filter and incorporated into the optimisation to ensure that the filter solutions are smooth and transmissive (and so plausibly more easy to manufacture). Smoothness is incorporated by modelling the filter transmittances as the weighted sum of the first few terms in a cosine basis. Transmission is bounded by reformulating the optimisation as a quadratic programming problem. We show how we can find a filter that optimises the fit to the data (mapping RGBs to XYZs) and has a transmittance in a given range (e.g. always at least 50% transmittance across the visible spectrum). We also consider the whole algorithm formulation in more detail in this chapter, Indeed, all the optimisations presented to this point are not guaranteed to find a global optimum, but rather the optimisations are search procedures that converge to a good solution. We show how better filters can be found using multiple initialisation points (i.e. by sampling).

A summary of the thesis contributions and suggestions for future work in Chapter 7 completes the thesis.

Chapter 2

Literature Review

In this chapter, we present the background that we build upon in this thesis. In Section 2.1, we review the human visual system and colorimetry. In Section 2.2, we show the model of the image colour formation calculating the camera triplets. The data sets that we are going to use in the experiments are described in Section 2.3. Various methods for colour correction - mapping RGBs to colorimetric counterparts - are discussed in Section 2.4. Quantitative measures for determining how well camera systems can measure colour are given in Section 2.5. In Section 2.6, we review the prior art in filter design for colour measurement. A brief summary completes this chapter.

2.1 Colorimetry

2.1.1 The Human Vision System

The perception of colour begins with light. Light is the magic key that unlocks the door of the wonderful world for every creature that has a vision system. The vision system of all creatures is sensitive to only a fraction of the electromagnetic spectrum. The visible spectrum for human vision runs, approximately, from 400 nanometres to 700 nanometres (nm), namely the visible spectrum, as shown in Figure 2.1.

This spectrum can be observed through a rainbow or by a prism as Issac Newton demonstrated back in 1666. In his famous experiment, the ‘colourless’ sun beam was refracted into a colourful spectrum, i.e. violet, indigo, blue, green, yellow, orange and red. As compelling as this experiment is, it is in fact a little misleading. Naively one might suppose colour perception is related to the colour of monochromatic stimuli. But, it is more complex than this. In fact, colours are properties of a vision system; that is to say,

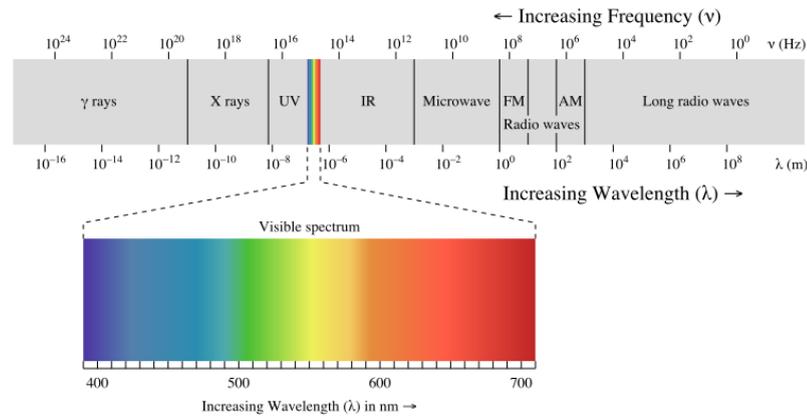


FIGURE 2.1: Electromagnetic of light with enlarged visible spectrum. The image is obtained from <https://en.wikipedia.org/wiki/Light>.

colour is primitively a physiological sensation towards a physical stimulus, rather than a physical property of materials.

The human vision system (HVS) is a sophisticated sensing system. Initial stages start from the eye, mainly including the radiant flux falling upon the cornea, traversing through optical media, and arriving at the retina, see Figure 2.2. Among multi-layers of the retina, there exists two types of photoreceptors – cones and rods, names derived from their shapes (see the shapes of the coloured pigments as shown on the right panel of Figure 2.2). These two photoreceptors are responsible for different tasks. The rods work at low light levels – scotopic vision – with a low spatial acuity, while cones are active at high light levels – photopic vision – stimulating colour perception and high spatial acuity. They work together under mesopic vision, e.g. under light level as moonlight.

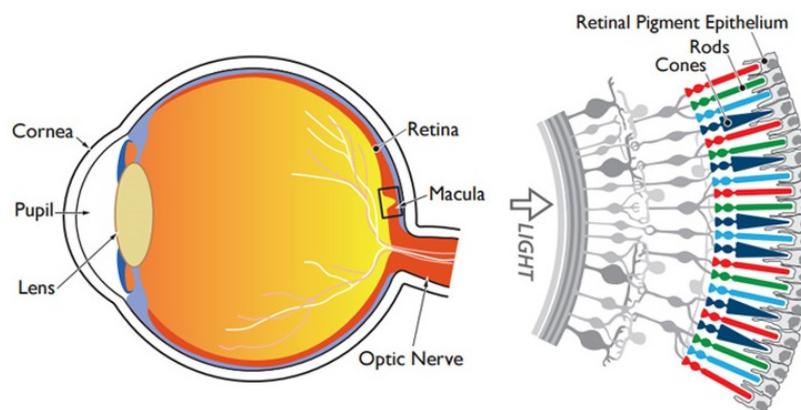


FIGURE 2.2: Section of human eye with a schematic enlargement of the retina. The cones and rods receptors are depicted in different shapes and colours on the right panel. This image is obtained from <http://www.closerlookatstemcells.org/stem-cells-and-medicine/macular-degeneration>.

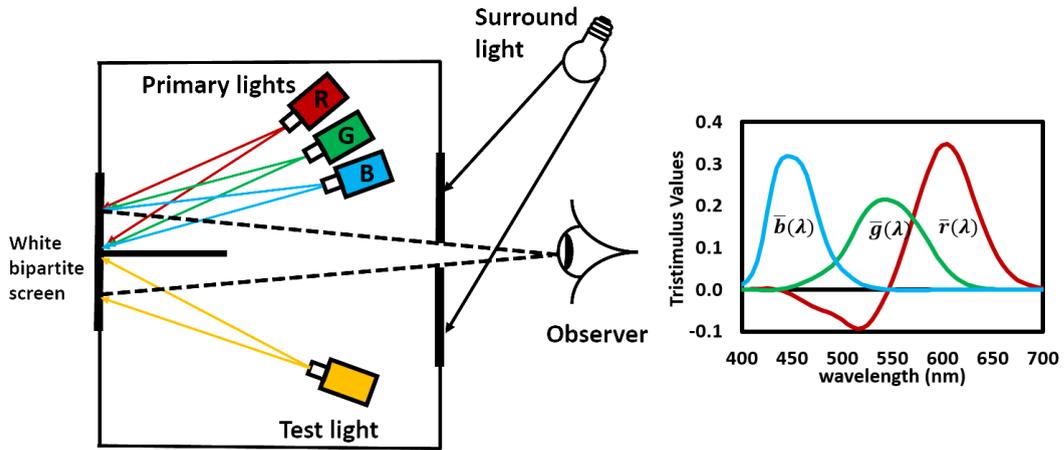


FIGURE 2.3: Schematic diagram of trichromatic colour matching experiment by additive mixture of lights. The intensity of the three primary Red, Green, and Blue lights can be adjusted to visually match a test light. On the right, we show the RGB colour matching functions for the CIE 1931 Standard Observer.

For a human with normal colour vision, the retina hosts three classes of cones. They are referred to as long-, middle- and short-wavelength sensitive cones (L, M and S), according to the part of the visible spectrum to which they are most sensitive. These cone types indicate that the colour vision of the human visual system is trichromatic.

2.1.2 The Colour Matching Experiments

The foundation of colour science is built on the colour matching experiment. The colour matching functions derived from the colour matching experiment provide a quantitative link between the physical light stimuli and the colours perceived by the human vision system, and is the cornerstone of the colorimetry science.

The success of colour matching experiment relies on the empirical fact that, for a human observer with normal colour vision, a light of any spectral power distribution can be visually matched to a carefully chosen mixture of three primary lights. That is, we have the ability to match a colour of any light as the mixture of three primary lights.

A simple apparatus for the colour matching experiment is illustrated in Figure 2.3. In the experiment, the observer views a bipartite field where one side is lit by a test light while the other side is lit by the light mixtures of three primaries (i.e. carefully chosen monochromatic red, green, blue lights). The observer's task is to adjust the intensities of three primary lights on one side in order to match the bipartite visual field of the fixed test light on the other side. A colour match is reached when the two visual stimuli appear visually indiscernible to the human observer. Sometimes, no match is possible and in this case, one of the primary lights should be added to the test light field. Mathematically,

we can model this as if we were subtracting some of the primary light. See [33] for more detail.

Grassmann made this colour additive mixture more explicit mathematically by setting out specific rules for additive colour matching [32, 45]. Grassmann proposed to treat the colour matching as a linear system and formulated four laws, which are called Grassmann's laws:

- Symmetry: If colour stimulus A matches colour stimulus B, then they still match when we exchange their colour fields.
 \implies *if* $A \equiv B$ *then* $B \equiv A$.
- Transitivity: If A matches B and B matches C, then A matches C.
 \implies *if* $A \equiv B$ *and* $B \equiv C$ *then* $A \equiv C$.
- Proportionality: If A matches B, then αA matches αB where α is a positive scalar indicating the power increase or reduction of the colour stimulus.
 \implies *if* $A \equiv B$ *then* $\alpha A \equiv \alpha B$.
- Additivity: if A matches B and C matches D, then the mixture of A and C matches the mixture of B and D.
 \implies *if* $A \equiv B$ *and* $C \equiv D$ *then* $A + C \equiv B + D$.

We adopt the notation of using the symbol ' \equiv ' to denote the visual match as in [16].

Now we will determine how the human visual system responds to the change of the spectral power distribution of the testing light. Let \mathbf{p}_R , \mathbf{p}_G , and \mathbf{p}_B denote the fixed primary lights used in the colour matching experiment, and \mathbf{E} denote an arbitrary testing light. Using the notation, a colour match between a test colour stimulus \mathbf{E} and the additive mixture of the primary stimuli can be expressed as

$$\mathbf{E} \equiv r \mathbf{p}_R + g \mathbf{p}_G + b \mathbf{p}_B \quad (2.1)$$

where the scalars r , g and b denote the intensities of the corresponding primary lights and are called the tristimulus values of stimulus \mathbf{E} .

The spectral stimulus values are derived when the monochromatic stimuli with unit intensity are employed. Let \mathbf{E}_λ represent a monochromatic light at wavelength λ , a colour match is written as

$$\mathbf{E}_\lambda \equiv r_\lambda \mathbf{p}_R + g_\lambda \mathbf{p}_G + b_\lambda \mathbf{p}_B \quad (2.2)$$

where the multipliers r_λ , g_λ and b_λ are the tristimulus values of \mathbf{E}_λ . When each monochromatic light across the visible spectrum is sequentially matched by the primaries, we can have the complete spectral colour matching functions.

The curves plotted in Figure 2.3 show the results of a colour matching experiment - where the test stimulus is a unit power stimulus at every wavelength across the visible spectrum - for an observer viewing a 2°-bipartite field. The light primaries used for deriving the colour matching functions are monochromatic R, G, and B stimuli with wavelengths of $\lambda_R = 700$ nm, $\lambda_G = 546.1$ nm, and $\lambda_B = 435.8$ nm, respectively [38]. Note that these monochromatic lights with specific peak wavelength values are reference primaries used as standards for colorimetric specifications [33].

To verify the equation in (2.2), consider a monochromatic light \mathbf{E}_λ at $\lambda = 500$ nm. From the RGB colour matching functions, we read $r_\lambda = -0.072$, $g_\lambda = 0.085$ and $b_\lambda = 0.048$. In this example, Equation (2.2) becomes

$$\mathbf{E}_{500\text{ nm}} \equiv -0.072 \mathbf{p}_R + 0.085 \mathbf{p}_G + 0.048 \mathbf{p}_B \quad (2.3)$$

Notice that the intensity value of the red primary is negative, indicating that the red primary is added to the test field. In the actual match of the 500 nm monochromatic light, the mixture of the green and blue primary lights are adjusted to match the mixture of the test light and the red primary light. Equivalently, we can rewrite as

$$\mathbf{E}_{500\text{ nm}} + 0.072 \mathbf{p}_R \equiv 0.085 \mathbf{p}_G + 0.048 \mathbf{p}_B \quad (2.4)$$

By now, we are able to quantify the colour matching behavior by three functions that relate the matching intensities of the primary light to the test light. Literally, the colour matching results do not offer us the exact spectral match between the test light and the primary lights, but the same visual appearance of the bipartite visual fields.

2.1.3 CIE Colorimetric System

The Commission Internationale d'Eclairage (CIE) is an international authority that has standardised the colorimetric specifications since its foundation in 1913 [65]. It recommends standards to describe colour, to enable colour measurement and to facilitate colour communication. These standards provide a consistent definition essential for colorimetry and for the meaningful exchange of colour information between different media.

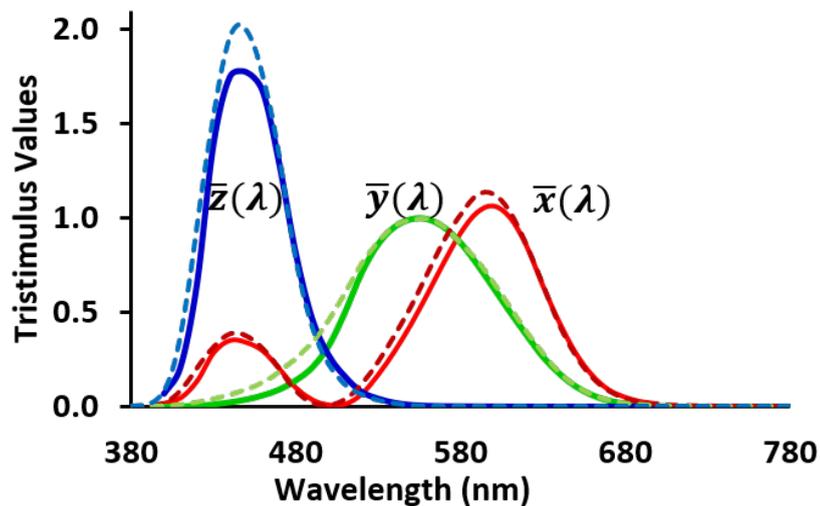


FIGURE 2.4: The CIE XYZ colour matching functions of 1931 Standard Observer (in solid lines) and of 1964 Standard Observer (in dashed lines).

Standard Colorimetric Observer

The CIE has established the 1931 Standard Colorimetric Observer, often referred as 2° Observer. This standard observer was derived from the results of two important colour matching experiments at the time, performed by Guild [33] and Wright [102]. Their experiments have led to the CIE 1931 RGB colour matching functions [24], shown in Figure 2.3.

The standard observer is used because scientific works have shown that individuals differ in the number of photosensitive cones and rods [79, 80], as well as the overall colour sensitivities [5, 77, 78, 102]. It is desirable to have a standard observer for quantifying the sensitivity responses of the visual system.

In Figure 2.3, it can be seen that RGB curves have negative values which made calculation a little cumbersome (in the time of early 1930s when computers were not available). To circumvent this problem, the CIE made some changes to redefine a new set with all positive values. The modification for the XYZ colour matching functions, $\bar{x}(\lambda)$, $\bar{y}(\lambda)$, $\bar{z}(\lambda)$ of the CIE 1931 standard observer, see solid lines in Figure 2.4, were made in certain manner [103]. Firstly, they are a linear combination from the RGB matching curves and their tristimulus values in the CIEXYZ functions are all positive. Secondly, the $\bar{y}(\lambda)$ function reserves the property of photometric quantities of the light which is set as identical as spectral luminous efficiency function [65, 83].

The 2° Observer is suitable for viewing angles between 1° and 4° , but fails to maintain high accuracy under some application conditions with larger viewing angle. Later in 1959, two groups (Stiles and Burch [78], Speranskaya [77]) performed the matching

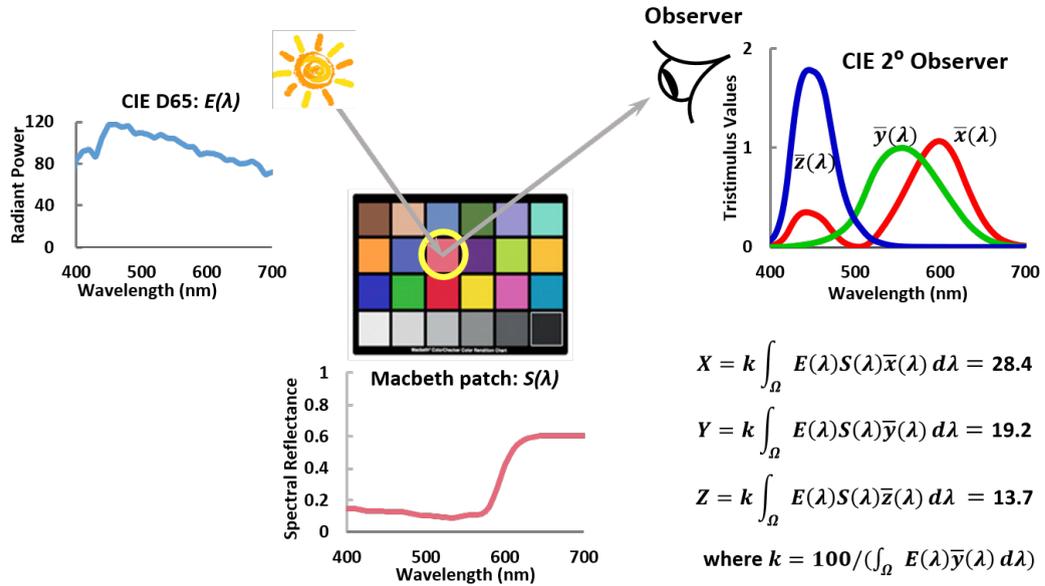


FIGURE 2.5: Calculation of XYZ tristimulus values given the spectral information of the illuminant, the surface and the observer.

experiments of enlarged 10° diameter viewing field. This leads to the CIE establishing a new system, the CIE 1964 Standard Colorimetric System, which provides 10° colour matching functions, shown as the dashed curves in Figure 2.4.

Calculation of the Tristimulus Values

With the help of the colour matching functions (CMFs), we are now able to compute the tristimulus values for any given light. Figure 2.5 visualises a colour measurement scenario with the spectral information available. The calculation of the XYZ tristimulus values from the spectra data – a CIE D65 illuminant [65], a Macbeth chart patch [59], and CIE 2° standard observer – are shown in Figure 2.5. In the following, we will show how the calculation of XYZ is performed.

The colour sensation process starts from the light emitted from the source illuminant, then reflected from the object surface before entering the observer's eye. Hence, the colour of a surface is mainly determined by three physical factors: the spectral power distribution of the illuminant, the spectral reflectance of the object, and the spectral sensitivities of the human visual sensors.

Let $E(\lambda)$ denote the spectral power distribution of an illuminant, $S(\lambda)$ the spectral reflectance of a surface, and $\bar{x}(\lambda)$, $\bar{y}(\lambda)$, and $\bar{z}(\lambda)$ the CIE CMFs. The XYZ tristimulus

values are defined as

$$\begin{aligned}
 X &= k \int_{\Omega} E(\lambda)S(\lambda)\bar{x}(\lambda)d\lambda, \\
 Y &= k \int_{\Omega} E(\lambda)S(\lambda)\bar{y}(\lambda)d\lambda, \\
 Z &= k \int_{\Omega} E(\lambda)S(\lambda)\bar{z}(\lambda)d\lambda
 \end{aligned} \tag{2.5}$$

where the normalising constant $k = 100 / \int_{\Omega} E(\lambda)\bar{y}(\lambda)d\lambda$ makes Y value equal to 100 for a perfect reflecting diffuser (that reflects 100% of the incoming light) [8]. The integral is made in the defined domain of the visible spectrum Ω .

Note that the integration can be alternatively expressed in the form of numerical summation with sampling intervals $\Delta\lambda$ as

$$\begin{aligned}
 X &= k \sum_{\lambda=\lambda_1}^{\lambda_2} E(\lambda)S(\lambda)\bar{x}(\lambda)\Delta\lambda, \\
 Y &= k \sum_{\lambda=\lambda_1}^{\lambda_2} E(\lambda)S(\lambda)\bar{y}(\lambda)\Delta\lambda, \\
 Z &= k \sum_{\lambda=\lambda_1}^{\lambda_2} E(\lambda)S(\lambda)\bar{z}(\lambda)\Delta\lambda.
 \end{aligned} \tag{2.6}$$

where the constant k is $100 / \sum_{\lambda=\lambda_1}^{\lambda_2} E(\lambda)\bar{y}(\lambda)\Delta\lambda$. The visible domain Ω is in the wavelength range of $[\lambda_1, \lambda_2]$. The functions with respect to the wavelength λ are sampled at every $\Delta\lambda$ across the visible spectrum.

Typically and justifiably from a colour accuracy point of view [76], a spectrum can be represented by 31 measurements made between 400 nm and 700 nm by 10 nm sampling interval. Note, the methods we set forth - since they are designed for the discrete domain - are agnostic about the sampling interval. If the data is given at a 5 nm sampling distance then each spectrum would be represented as a 61-component vector. Henceforth, we will talk about spectra being 31-dimensional vectors (and this corresponds to the format of most available measured spectral data).

The product of $E(\lambda)S(\lambda)$ is termed as the colour signal, i.e. $C(\lambda) = E(\lambda)S(\lambda)$. Henceforth, we will use $C(\lambda)$ to represent the spectral product of the light and reflectance for

convenience. By substituting into (2.6), we have

$$\begin{aligned} X &= k \sum_{\lambda=\lambda_1}^{\lambda_2} C(\lambda)\bar{x}(\lambda)\Delta\lambda, \\ Y &= k \sum_{\lambda=\lambda_1}^{\lambda_2} C(\lambda)\bar{y}(\lambda)\Delta\lambda, \\ Z &= k \sum_{\lambda=\lambda_1}^{\lambda_2} C(\lambda)\bar{z}(\lambda)\Delta\lambda. \end{aligned} \quad (2.7)$$

CIE 1976 Uniform Colour Spaces and Colour Difference Formulae

The CIE XYZ colour space in the previous section has a non-negligible drawback that the colour difference in the space is not uniform. That is, the equal distances in the colour space are not consistent with the equal perceptual differences. Hence, in 1976, the CIE technical committee made two uniform colour space recommendations, CIELAB and CIELUV [103]. We mainly use the CIELAB colour space in the study and therefore only the calculations for this colour space have been given below.

The CIELAB colour space is defined by the following equations:

$$\begin{aligned} L^* &= 116f(Y/Y_n) - 16 \\ a^* &= 500[f(X/X_n) - f(Y/Y_n)] \\ b^* &= 200[f(Y/Y_n) - f(Z/Z_n)] \end{aligned} \quad (2.8)$$

$$f(x) = \begin{cases} x^{1/3} & \text{if } x > (24/116)^3 \\ (841/108)x + 16/116 & \text{if } x \leq (24/116)^3 \end{cases}$$

where X_n, Y_n, Z_n are the values of X, Y, Z for a perfect reflective diffuser white.

CIELAB colour space is not only uniform but its axes correlate roughly to perceptual attributes. From the calculation of (2.8), it can be observed that L^* is a function of Y and Y_n suggesting a reflection of the luminance intensity of the stimulus. The horizontal axis a^* approximates the colour direction of green-red, while the vertical axis b^* gives a clue about blue-yellow direction.

As a uniform colour space, the Euclidean distances in CIELAB colour space can be calculated to represent approximately the perceived colour difference between two different stimuli. The CIELAB colour difference equation is defined as

$$\Delta E_{ab}^* = [(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2]^{1/2} \quad (2.9)$$

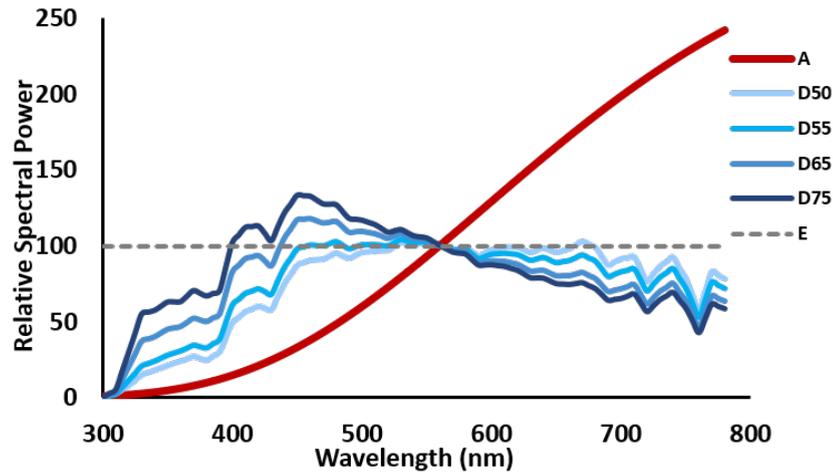


FIGURE 2.6: Relative spectral power distributions of CIE Standard Illuminants A, D50, D55, D65, D75, and E.

When applying this colour difference equation, note that it was originally designed for the colour stimuli of the same size, viewed under lights not too different from D65 in neutral surroundings.

CIE Standard Illuminants

Objects are always lit under various illuminants, e.g. changing day lights, artificial indoor lights, and outdoor lights. According to (2.5), stimulus values are calculated from the multiplication of illuminant and object reflectance. Therefore, any change in the illuminant spectral power distribution can result in different XYZ values. With the existence of countless light spectra, the reference colour appearance of the object under typical lighting conditions is of great importance. In order to simulate typical lighting conditions, CIE established several standard illuminants and artificial light sources [38]. We list the most widely used as follows:

- Standard illuminant A simulates the incandescent lamp light with a correlated colour temperature (CCT) of 2856 K.
- Standard illuminant D65 represents a typical daylight illuminant with a CCT of about 6504 K and is one of a series of CIE daylight illuminants designated at CCTs of about 5000 K, 5500 K, and 7500 K, known as D50, D55, and D75 respectively.
- Illuminant E has an even spectral power distribution representing the equal-energy spectrum.

Note that colour temperature is defined as the temperature of the blackbody radiator (or Planckian radiator) which had the same chromaticity as the given stimulus. The

term - correlated colour temperature - is used when some light sources have chromaticity values deviated from the Planckian locus.

The physical property of an illuminant is determined by its Spectral Power Distribution (SPD) of the radiation emitted from the light source. The SPD denotes the energy distribution at each wavelength throughout the visible spectrum. The relative spectral power distribution of the CIE standard illuminants are drawn in Figure 2.6. The relative SPD is normalised SPD at 560 nm where the value is set to 100.

2.2 Image Colour Formation

A physically accurate model of camera image formation for a Lambertian surface colour (the one that reflects light into all viewing directions equally) [8, 66] is written as

$$\rho_k = \int_{\Omega} E(\lambda)S(\lambda)Q_k(\lambda) d\lambda, \quad k \in \{R, G, B\}. \quad (2.10)$$

Similar to (2.5), the colour is formed by the interaction of the light $E(\lambda)$, the surface $S(\lambda)$, and the sensor Q_k at a pixel. The subscript k indicates the colour channel (usually, short-, medium- and long-wave sensitive mechanisms or more commonly known as R-, G- and B- channels). The function $Q_k(\lambda)$ is the spectral sensitivity of the k th camera colour channel.

Substituting the colour signal $C(\lambda) = E(\lambda)S(\lambda)$ into (2.10), we have

$$\rho_k = \int_{\Omega} C(\lambda)Q_k(\lambda) d\lambda, \quad k \in \{R, G, B\}. \quad (2.11)$$

We can recast the continuously integrated responses in (2.11) using discrete representation

$$\rho_k = \sum_{\lambda=\lambda_1}^{\lambda_2} C(\lambda)Q_k(\lambda) \Delta\lambda, \quad k \in \{R, G, B\}. \quad (2.12)$$

We use a constant sampling value $\Delta\lambda = 10 \text{ nm}$ in this thesis and henceforth we will assume the term $\Delta\lambda$ is incorporated into the camera response vectors.

Under the discrete representation, the image formation problem can be formulated using linear algebra. The tools of linear algebra are very useful in studying the colour imaging problem and are extensively used in this thesis.

Using vectors for sampled spectra, (2.12) can be compactly written as

$$\boldsymbol{\rho}^T = \mathbf{c}^T \mathbf{Q} \quad (2.13)$$

where \mathbf{c} is a 31×1 vector of the colour signal spectrum sampled at measuring intervals. \mathbf{Q} is a 31×3 matrix whose columns respectively contain the R, G, and B spectral sensitivities. The 3×1 vector $\boldsymbol{\rho}$ denotes the R, G, and B response values.

Let \mathbf{C} denote a $31 \times n$ matrix whose columns represent n colour signal spectra. The set of $n \times 3$ RGB responses can be written as a single concise expression:

$$\mathbf{P} = \mathbf{C}^T \mathbf{Q}. \quad (2.14)$$

2.3 The Data Sets

In this section, we introduce the data sets that we will use in the design of the optimal prefilters for digital cameras, and also in the evaluation of colour measurement performance.

2.3.1 Camera Spectral Sensitivity Functions

The camera data set - the spectral sensitivity functions of digital cameras ranged from 400 nm to 700 nm with an interval of 10 nm - that we will be using are the work of Jiang *et al.* [41]. The camera set comprises of 28 commercial cameras covering a variety of types, including professional DSLRs, point-and-shot, industrial and mobile cameras. This camera group has a mix of CMOS and CCD sensors. Most cameras are equipped with CMOS sensors, except Canon EOS 1D, Nikon D200, D40, D50, D80, and two Point Gray Grasshopper cameras which use CCD sensors.

A large component of the set comprises 9 Canon and 10 Nikon cameras. Generally, cameras from the same manufacturer show similar sensitivity features. We will pay close attention to these subsets while applying our methods.

The ground-truth spectral sensitivity functions of the cameras were measured using a monochromator and a spectrophotometer PR655. The measurement setup of how the camera sensitivities are derived are not the focus here but details can be found in the supplementary material of [41].

We will design colour filters for each individual camera given their spectral sensitivity functions. The validity of the filter design method (with respect to the colour accuracy improvement) will be analysed both individually and collectively.

2.3.2 Reference Colour Matching Functions

Unless otherwise stated, we will use the CIE 1931 XYZ colour matching functions of a 2-deg standard observer. The motivation of the 2-deg observer is that as for a digital camera, the pixel colour is viewed with a small viewing angle and 2-deg is more suitable for imaging applications.

2.3.3 Illuminant and Object Spectra

To evaluate the colour measuring performance, a data set of illuminant and reflectance spectra is required. Both illuminant and reflectance data set are measured by the Computational Vision Laboratory of Simon Fraser University [1], which is publicly accessible. The SFU data set was chosen because it provides high quality spectral data and contains sufficient variety of reflectances and common illuminants. We will refer to this data set as SFU data hereafter.

The surface reflectance set is consisted of data from several sources, including the 24 Macbeth colour checker patches, 1269 Munsell chips, 120 Dupont paint chips [93], 170 natural objects [93], the 350 surfaces in Krinov data set [46], and 57 additional surfaces measured by Simon Fraser University [6].

The illuminant set has 102 light spectra, including 81 natural daylight spectra measured at various times of the day and under various weather conditions, 11 artificial lights (e.g. 4 tungsten lights, 3 fluorescent lights and 4 simulated daylights) in the laboratory condition and the rest 10 illuminants are synthetic lights.

2.4 Colour Correction

2.4.1 Colour Correction Transformation

For digital cameras, a colour correction is usually performed on the RGB measurements to give estimate of the CIE tristimulus values of an object. Given the spectral data of the capturing illuminant, object reflectances and the spectral sensitivity functions, we can calculate the ground-truth XYZ values and the camera colour responses. A

correction transformation will be determined by minimising the colour difference between the measured and estimated colour values.

Let two $n \times 3$ matrices \mathbf{X} and \mathbf{P} represent the corresponding XYZ and RGB responses of n colour signals. The function Ψ mapping the RGB values \mathbf{P} to the ground-truth \mathbf{X} can be generally represented as:

$$\mathbf{X} = \Psi(\mathbf{P}) \quad (2.15)$$

Many different algorithms for solving the colour correction problem have been developed in the literature. Popular mapping methods includes linear [28, 98] and polynomial least-squares regressions (there are many variants of the linear, polynomial and root-polynomial methods) [26, 35, 48], look-up-tables [37], and neural networks [13, 104].

2.4.2 Linear Colour Correction

The linear colour correction method uses a 3×3 matrix. The best linear correction matrix between \mathbf{P} and \mathbf{X} can be calculated as a least-squares regression:

$$\min_{\mathbf{M}} \|\mathbf{PM} - \mathbf{X}\|_F^2 \Rightarrow \mathbf{M} = [\mathbf{P}^T \mathbf{P}]^{-1} \mathbf{P}^T \mathbf{X} = \mathbf{P}^+ \mathbf{X}. \quad (2.16)$$

where \mathbf{M} is a full rank 3×3 matrix. The ‘book formula’ shown in the right hand side of the equation \mathbf{P}^+ is the Moore-Penrose inverse, i.e. $\mathbf{P}^+ = [\mathbf{P}^T \mathbf{P}]^{-1} \mathbf{P}^T$.

This simple but useful method, even though it is not optimal in terms of fitting error, has two advantages compared to most non-linear methods. First, the transform scales linearly with exposure. If the scene is made twice as intense (e.g. by doubling the quantity of incoming light), the same matrix correctly maps the camera measurements to XYZs (because the magnitude of camera RGBs and XYZs both double). Typically, non-linear methods do not have this exposure-invariant property (except root-polynomial [29] and hue-plane preserving [2]).

The second advantage is that a linear transform is, well, linear. The human eye measures colour stimuli linearly: at the cone quantal catch level, the response to the sum of two spectral stimuli is equal to the sum of the responses to the two stimuli viewed individually [97]. This can be an important *physical* consideration. As an example, when we view a surface that has highlights, the recorded colour is a convex sum of the so-called *body* colour (the colour name we would assign to the object) and the colour of the highlight [43], sometimes called the *interface* colour. If, we are viewing a red shiny plastic surface the body colour is red and if the viewing light is white then the interface colour is also white (i.e. the same as the colour of the light). As we move from pure body to pure highlight colour, the measured XYZs lie on a 2D plane in the colour

space. Equally, the camera, which at the sensor level has a linear response, will also make measurements that lie on a 2D plane. But, a non-linear correction will distort the plane and the result will be an image that is not physically accurate or even physically plausible. This problem is discussed in detail in [2, 55].

2.4.3 Calculating Colour Error

Often, the perceptual colour difference metric is commonly used for the evaluation of the colorimetric performance for a colour measurement system. The smaller colour difference values, the better colour accuracy.

We will use the colour difference formula, ΔE_{ab}^* in the CIELAB colour space, among other options such as CMC (1:c) [51] (widely used in textile industry), more advanced formula CIEDE2000 [50], most recently ΔE_{ITP} for high dynamic range and wide gamut imagery [69]. The colour error metric ΔE_{ab}^* was chosen for the following reasons. First of all, ΔE_{ab}^* can be easily and intuitively computed as the Euclidean distance between two colour points in the CIELAB colour space while other metrics require considerable increase in complexity of calculation. Secondly, ΔE_{ab}^* is a widely used and accepted colour difference metric which makes it easier for comparison to other studies who also use ΔE_{ab}^* for evaluation. More importantly, it facilitates our study results better understood by the industry. Thirdly, there is a good correlation between ΔE_{ab}^* and CIEDE2000 but with the advantage of much less calculation cost. Lastly, we can design our filter for minimising the colour difference metric. It is possible for us to derive a vigorous formula (derivatives of the objective functions) based on the ΔE_{ab}^* under linearised colour space [74].

Practically, the colour performance of a camera can be assessed through image capturing of the measuring scene under a real illuminant. The measured RGBs are mapped to make an estimate of the XYZs in the colour correction process. later, the predicted and the ground-truth XYZs are converted into the CIELAB colour space and then the colour difference between them is evaluated in terms of ΔE_{ab}^* .

When there are multi-illuminants and multi-reflectances, we calculate the average colour error across the reflectance set under each illuminant condition. For example, if we have m illuminants and n reflectance spectra, the overall mean colour error is calculated as

$$\Delta E_{avg}^* = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n \Delta E_{i,j}^* \quad (2.17)$$

where the colour error of every reflectance under each illuminant, $\Delta E_{i,j}^*$, is calculated.

Other statistical indicators like median, 95-percentile, maximum colour errors can also be used to have a better sense of the colour performance across the data set.

2.5 Measures of the Goodness of a Set of Spectral Sensitivities

Given a set of spectral sensitivities, the questions that naturally comes is that how they perform if they are used for colour measurement. It is essential and desirable to have quality measures of filters that can be trusted to use in the evaluation of their performance in colour reproduction. In the following paragraphs, we review some widely recognised goodness measures for filter quality evaluation in the imaging systems in the literature.

To measure the quality of a spectral sensitivity for colour measuring, Neugebauer proposed a quality factor, namely q-factor [62]. Given a camera sensitivity, the q-factor measures its goodness by returning a number in the range of $[0, 1]$. We say a sensitivity is maximally or 100% good if there is a linear combination of the XYZ CMFs that exactly matches the camera sensitivity. The intuition here is reasonable: if we had three linearly independent camera sensitivities that each scored 1, then we can find a linear combination of the sensitivities that equals the XYZ CMFs.

Suppose \mathbf{X} represents the 31×3 matrix of the CMFs (X, Y and Z are in the columns) and, as a reminder, each column has 31 terms because we are sampling the visible spectrum from 400 nm to 700 nm at 10 nm sampling. Let \mathbf{f} denote the 31×1 camera sensitivity. Then the closest linear combination of \mathbf{X} that matches \mathbf{f} is written as $\mathbf{X}\mathbf{X}^+\mathbf{f}$ where \mathbf{X}^+ denotes the Moore-Penrose inverse [68]. The q-factor is defined as the following:

$$q = \frac{\|\mathbf{X}\mathbf{X}^+\mathbf{f}\|_2^2}{\|\mathbf{f}\|_2^2} \quad (2.18)$$

What does the q-factor tell us about the goodness of a set of camera sensitivity functions? Well, the answer is not straightforward. Colour response is three dimensional. Yet, the q-factor treats each camera sensitivity separately.

Unlike the q-factor targeting an individual sensitivity function, Vora and Trussell [88, 91] proposed a goodness measure for a set of sensitivity functions. They adopted a vector space approach for analysing the filter quality: the goodness measure, namely Vora-Value. It is defined by the projection of the space spanned by filters onto that spanned by the human visual space under a certain illuminant.

Let \mathbf{Q} represent a set of spectral sensitivities and \mathbf{X} represent the set of the XYZ CMFs. The Vora-Value is defined as

$$\nu(\mathbf{Q}, \mathbf{X}) = \frac{\text{trace}\left(\mathbf{Q}(\mathbf{Q}^T \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T\right)}{\alpha} \quad (2.19)$$

where the superscripts T and $^{-1}$ denote the matrix transpose and inverse, respectively, and $\text{trace}()$ returns the sum of the elements along the diagonal of a matrix. The constant α defines the number of visual channels (or mathematically, the dimension of the rank of matrix \mathbf{X}), three for the trichromatic human visual system.

While (2.19) looks unwieldy, it has an elegant geometric interpretation. We can think of the columns of \mathbf{X} and \mathbf{Q} as spanning three dimensional subspaces of the 31-dimensional space (as \mathbf{X} and \mathbf{Q} are 31×3 matrices). The Vora-Value can also be regarded as a percentage-type of measure of how similar the subspaces are. If the Vora-Value is 1, this means the subspaces are identical. And, the RGBs measured by the camera are a linear transform from the XYZ tristimuli. When the Vora-Value is 0, this means the subspaces are orthogonal to each other. As a heuristic, if a camera has a Vora-Value greater than 0.95, then, in general, RGBs can be colour corrected to XYZs with a small average error.

It is shown that Vora-Value is a generalisation of Neugebauer's q -factor and can be used to evaluate the performance of a set of sensitivity functions with any number. Vora and Trussell [87] also extended it to a family of data-dependent measures: given a data set with known statistical information, these data-dependent measures can assess the ability of the sensitivity set relate to the mean-square tristimulus errors.

A more comprehensive measure was later proposed by Sharma and Trussell [73], named 'Figure of Merit' (FOM). This measure evaluates the spectral sensitivity set related to a perceptually uniform colour error (the CIELAB colour error) and takes the measurement noise into account. In this measure, two important aspects are accounted for: the degree of perceptual relevance and the varying noise levels. Moreover, it provides a unified framework with flexibility to derive variant figures of merits that can suit the practical applications. Later, Quan *et al.* [71] extended the FOM to a more complex measure, namely Unified Measure of Goodness (UMG), by incorporating a more practical noise model (including both signal-dependent and signal-independent noise).

In our work, we will use the goodness measure, Vora-Value, to evaluate the performance of the optimal filters solved from our optimisations.

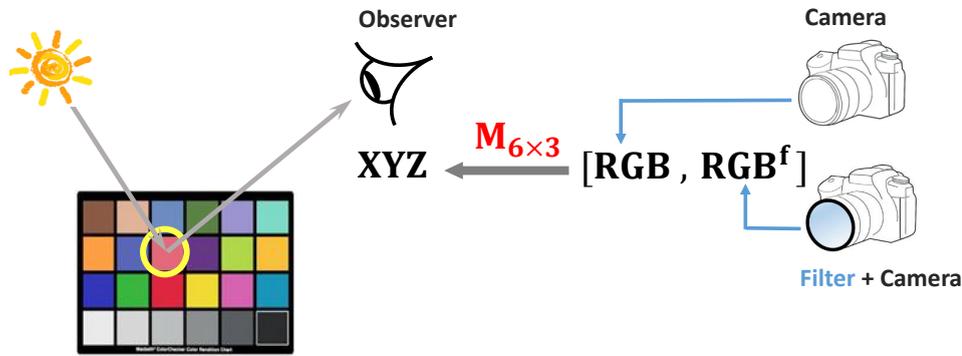


FIGURE 2.7: For measuring the object colours, two images are taken, once with and once without a colour filter placed in front of the camera. The colour values combined from two images are mapped to the reference XYZ tristimuli using a 6×3 correction matrix.

2.6 Filter Design in Colour Imaging

The problem of designing the optimal sensitivities for a colour imaging system has been intensively investigated [3, 34, 70, 73, 90, 96, 100, 105]. For every colour measuring device, the sensitivity characteristics of the imaging device are fundamental in determining how the system responds to the colour stimuli. The imaging system can be built for many purposes. Yet, here we focus on the objectives that are devoted for the colorimetric reproduction to achieve high colour fidelity, i.e. they interpret the physical data obtained from sensors to account for the colour properties of the human visual system.

In the literature, most studies focus on the optimal sensitivity design of the whole sensitivity set - a complete new sensitivity set consisting of three or more channels - for the colour measuring system. However, the approach we propose in this thesis is much simpler. Given an off-the-shelf camera, we simply place a specially designed filter in front of the camera which will reshape the effective sensitivity functions of the camera. Our objective is to seek optimal transmissive prefilter so that the filtered camera becomes more colorimetric (by significantly reducing the colour errors in linearly transforming the RGB values to the XYZ values).

The idea of placing a coloured filter in front of a camera to make it more colorimetric has been previously proposed [25]. In this work, Farrell and Wandell proposed placing a filter between the object and the sensors. The approach is illustrated in Figure 2.7. Two images are captured, once with and once without the filter. That is, the first is taken as normal with the native spectral sensitivities; the second is taken by placing a coloured filter in front of the imaging device. In this way, six measurements are made per pixel.

Let \mathbf{P}^f and \mathbf{P} denote the image RGB values of n colour samples taken with and without a prefilter. The superscript f denotes the image taken with the filter presented. Thus, both \mathbf{P}^f and \mathbf{P} are $n \times 3$ matrices. With two images taken, mathematically, we can compose a new matrix

$$\mathbf{P}_{n \times 6} = [\mathbf{P}, \mathbf{P}^f] \quad (2.20)$$

where the new matrix has the size of $n \times 6$.

For this case, when we perform the linear colour correction as in (2.16), the least-squares matrix \mathbf{M} becomes a 6×3 matrix:

$$\min_{\mathbf{M}} \|\mathbf{P}_{n \times 6} \mathbf{M} - \mathbf{X}\|_F^2 \Rightarrow \mathbf{M}_{6 \times 3} = [\mathbf{P}_{6 \times n}^T \mathbf{P}_{n \times 6}]^{-1} \mathbf{P}_{6 \times n}^T \mathbf{X}. \quad (2.21)$$

Note that the ground-truth XYZ tristimuli \mathbf{X} is still an $n \times 3$ matrix.

Farrell and Wandell's approach can offer significantly reduced colour errors (linearly from camera outputs to tristimulus values). The emphasis of the prior art was to increase the dimensionality of the capture. Since two captures are made which produce six measurements per pixel. The target XYZs is matched by applying a 6×3 correction matrix.

Although both our proposed and Farrell and Wandell's work consider the use of prefilters for the image, the logic behind is different. In [25], the filter is used to increase measuring data representation by capturing more images and the best filter was chosen empirically (from a set of commercially available choices that supports the best colour measurement). Of course, by doing so, the filters are not optimal. Yet, our proposal aims at reforming the spectral characteristic of the sensing system in order to change the effective camera sensitivities to meet either the Luther condition or to better predict the XYZ tristimulus values.

In the following, we review the methods that look for the optimal set of sensitivity functions in the colour measuring applications. Most colour imaging systems are assumed to have three RGB channels. Usually, the sensitivity set is designed under a well-defined objective. The evaluation metrics reviewed in Section 2.5 are good options for being the design measures.

As reviewed in the last section, Vora and Trussell [91] developed a simple 'goodness' measure, Vora-Value, to assess how closely a spectral sensitivity set measures colours compared with the human visual system. In [88, 90], Vora and Trussell used the Vora-Value as a design tool for finding the optimal sensitivity set to be used in a colour scanner. They also emphasised the importance of generating smooth sensitivity functions for better physical realisability. To achieve this goal, a group of single-Gaussian filters

and a sum-of-Gaussian functions were used. Moreover, by using the Gaussian functions reduces the number of degrees of freedom of the filter parameters and forces the resulting filters to be smooth.

Vrhel and Trussell investigated the design of optimal filters under multi-illuminants and as well as the impact of noise [75, 95, 96]. Interestingly but not surprisingly, the best sensors that solve these problems are not linearly related to the XYZ CMFs [73].

The spectral sensitivity functions can also be designed with the goal of minimising the perceptual colour error, such as ΔE_{ab}^* . Wolski *et al.* [100] formulated a non-linear optimisation for the filter design using the criterion of ΔE_{ab}^* colour error. An iterative searching technique was adopted for seeking the solution. Of course, this method inevitably boosts the computational complexity since as indicated by equations in (2.8), the transform from CIE XYZ to CIE LAB is nonlinear.

We argue that the CIELAB space is not a good objective to aim for in designing the best sensitivities, in the sense that it would not provide a tractable optimisation criterion because of the non-linearity nature. Thus it is not easy to use in optimisation for filter design problems. On the contrary, a simple formula (such as the Luther condition and Vora-Value) saves computational time and, more importantly, often provides an insight into the system behaviour.

When higher colour fidelity is required, another option we have is to increase the number of sensors/filters used for imaging. Vrhel and Trussell [95] showed that an overall colour error below the just noticeable difference (JND) threshold can be achieved if a set of four colour filters is used (with a linear colour correction). A more sophisticated case is to take the measuring noise into consideration. In [82], Trussell *et al.* claimed that a group of 12 or fewer Gaussian-modeling filters is sufficient to achieve an average colour error of less than ΔE_{ab}^* and maximum error of less than $3\Delta E_{ab}^*$ under a practical level of shot noise.

In some applications, such as the fine-art archiving, the colour accuracy is more critical and the restoration of the spectral information is sometimes required. For these applications, multi-spectral and hyper-spectral imaging systems are widely used [18, 34, 63, 84]. In the multi-spectral system, a camera is equipped with many more filters so that the colorimetric values can be determined with great accuracy and more importantly, much of the spectral information of the objects can also be retained. To build up a multi-spectral imaging system, one can apply a more complicated colour filter arrays to increase the number of colour bands [60] or use spectrally-tunable LED lights to make multiple images captured under various lights [39, 58, 105]. The multi-spectral approach can be

problematic for it is characterised by longer capture time and much more expensive imagers.

A great challenge for filter design approach is to fabricate the exact filters to fulfill the desired accuracy target. One way to take off the burden of the fabrication is to selecting and combining a set of known filters to achieve the optimal colour fidelity [19, 22, 44]. Given a selection criterion of an error metric, the best filter set can be picked using a heuristic search method. This strategy is widely adopted in the work of [3, 4, 39, 53]. It is a practical choice though not optimal.

Our filter design approach focuses on finding the best filter without worrying about the fabrication process. Yet, we also extend our filter design optimisation to restrain the shape and transmissivity of the filters to improve the physical feasibility of our filters (so plausibly could make the filters easier to manufacture).

We would like to make a final note on two types of approaches: data-independent and data-dependent. The former designs the colour filters to replicate, although most often to linearly relate to the CIE CMFs such that a unique linear transformation can bring the device colours to the tristimulus coordinates. As for the data-dependent approach, it takes advantage of a priori knowledge about the measuring illuminants and reflectances. However, there are pros and cons of the data-dependent approach. If the data of interest are known, the data-dependent manner benefits from the data information and results in better colorimetric performance. On the other hand, if the measured data are very different from the set that is used for deriving the optimal sensitivities, instead, it will fail to measure the new data set accurately. Both approaches are considered and used for designing the prefilter in this thesis.

2.7 Summary

The fundamentals of colour vision and colour imaging are described in this chapter. These fundamentals form the basis for the following chapters where we develop our approaches. In this thesis, we will aim to design a colour prefilter for a digital camera with known sensitivity information such that the new camera system - using the linear colour correction - is able to deliver much better colorimetric measurement. The filter will be parameterised as an optimisation problem using both data-independent and data-dependent criteria. The colour performance will be evaluated in terms of perceptually uniform colour error indicators.

Chapter 3

Luther Condition Filter Design

In this chapter, we propose to design a prefilter to be placed in front of the camera that results in the filtered camera best meets the Luther condition. That is, the effective spectral sensitivities of the ‘filter+camera’ system are linearly related to the colour matching functions of the human visual system.

3.1 Introduction

Digital cameras are designed to record triplet colour responses in analogy to the trichromatic human visual system which has three types of photosensitive cones related to colour perception [97]. Practically, the spectral sensitivities of a camera are designed to satisfy many objectives including manufacturability and the need to minimise the conspicuity of noise [85]. As a consequence, the camera coordinates it produces are device dependent and most cameras do not ‘see’ colours like the way we human do. That is, the RGBs a camera measure, are not a linear transform from colour matching triplets. More formally, a camera ‘sees’ like we do – with a simple linear colour correction – if and only if its spectral sensitivities are a linear combination of the corresponding human visual system matching curves (or equivalently a linear combination of the CIE XYZ colour matching functions) [36]. This condition is often termed as the Luther condition [40, 52]. If this condition is fully met, we say that the camera is *colorimetric* and it can reproduce the exact XYZ tristimulus values (within a linear transform).

In the literature, it has been proposed that to make a camera more colorimetric we could take several pictures under different coloured filters [25]. However, this prior art approach sought to increase the dimensionality of the capture – i.e. to take an image with and without a filter – with respect to which a linear combination can better predict

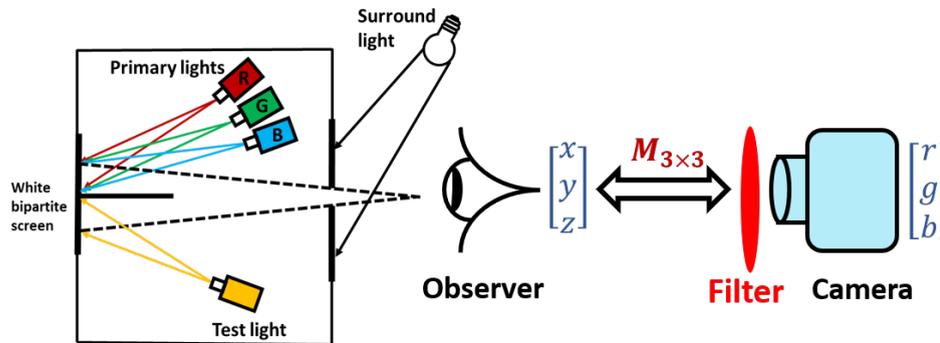


FIGURE 3.1: The use of a digital camera with a colour filter placed between the scene and the camera in a colour matching experiment. The goal is to find a prefilter such that the RGBs a camera measures (after a 3×3 correction matrix) are the same as the colour mixtures a human observer would give to make a match.

what we see (e.g. better predict XYZs). The multiple-image approach can be problematic because it necessitates a long capture process where the user has to change filters between image captures. If nothing else, the captured images are not in registration, which is a problem in itself.

Of course, a more pragmatic way to proceed — and the one that is implemented in all camera processing pipelines — is simply, given the RGBs for a given camera do, by mathematical means, find a 3×3 correction transformation matrix that brings them as close as possible to the desired XYZs (or display RGBs). Interestingly, when we linearly regress RGBs to XYZ tristimuli, we can interpret this as linearly transforming the sensors themselves. That is, a new camera whose sensitivities are modified by the linear regression transform approximately measures the desired XYZs. It follows that one strategy to improving the colour measurement capability of a camera would be to change the camera sensitivities. The closer the spectral sensitivities of a camera are to being linearly related to the XYZ colour matching functions, the better it will perform as a tool for colour measurement, i.e. the more *colorimetric* it will become.

Here we propose that finding the filter to make a camera more colorimetric should be chosen in tandem with the best colour correction transformation mapping RGBs to XYZs. Physically, the role of a filter, which absorbs light on a per wavelength basis, is multiplicative. If $f(\lambda)$ is a transmittance filter and $\underline{Q}(\lambda)$ denotes the camera sensitivities (red-, green- and blue- channels), then $f(\lambda)\underline{Q}(\lambda)$ is a physically accurate model of the effect of placing a filter in front of the camera sensor.

Our filter design method is built directly on top of the Luther condition [52]. That is, the filter multiplied by the camera spectral sensitivities is ‘almost’ a linear combination from the colour matching functions of the human visual system. If $\underline{X}(\lambda)$ denotes the XYZ colour matching functions, then an explicit Luther condition inspired formulation

of the filter design problem can be modeled as:

$$\min_{f(\lambda), \mathbf{M}} \| f(\lambda) \underline{Q}(\lambda) \mathbf{M} - \underline{X}(\lambda) \|_F^2 \quad (3.1)$$

where $f(\lambda)$ defines the spectral transmittance of the colour filter, and \mathbf{M} is a fixed 3×3 transform matrix. $\| \cdot \|_F^2$ denotes the square of the Frobenius norm [31].

Figure 3.1 illustrates the objective of the prefilter for the camera system – to match the human vision results with equal visual appearance. In the experiment of colour matching, the goal of the observer is to adjust the intensities of the three primary lights (R, G and B lights) to make a colour that looks identical to the bipartite of the test light [78, 102]. In this example, R, G and B are set to, respectively, the intensities r , g , and b in order to make a visual match. In Figure 3.1, we seek a filter, shown in red in front of the camera (though the filter colour is not a priori specified), so that the RGBs of the camera multiplied by matrix \mathbf{M} record the same triplets as the observer does under the same viewing geometry. That is the camera ‘sees’ the mixture of the light intensities a human observer would use to make a match.

The colour matching experiment is of direct practical importance. Indeed, suppose a display device can output three colours equal to the R, G and B stimuli used in the colour matching experiment. It follows that a camera that had sensitivities equal to CIE RGB colour matching functions would *see* the correct RGBs to drive the display that would result in a perceptual match to the test light. Equally, it would suffice that camera’s sensitivities were linearly related to the CMFs since we could linearly transform the camera measurements to the correct RGB to drive the display. Of course, we note that - unlike the RGB colour matching functions - we can only *drive* the putative display with positive numbers. Consequently, there are real world colours that cannot be reproduced on displays.

Later in this chapter, we will show that for a given sensor set, the best filter and best linear mapping in the minimisation of (3.1) can be found together by solving an alternating least-squares (ALS) regression problem. To quantify the improvement in colour measurement for a given camera afforded by a colour filter, we calculate the Vora-Value [88] before and after correction. Remarkably, we find that some cameras become ‘nearly’ colorimetric. We also evaluate the colorimetric performance via a colour correction experiment using a set of measured spectra in terms of the error metric ΔE_{ab}^* [103], with and without a filter.

3.2 The Luther Condition

Let us denote individual camera and XYZ responses as $\boldsymbol{\rho}$ and $\boldsymbol{\chi}$. We can use a camera to *exactly* measure colours if and only if there exists a function such that $g(\boldsymbol{\rho}) = \boldsymbol{\chi}$ for all spectra. It follows that if a pair of spectra, \mathbf{c}_1 and \mathbf{c}_2 , integrates to the same RGB, then this pair must also integrate to the same XYZ tristimuli:

$$\mathbf{c}_1^T \mathbf{Q} = \mathbf{c}_2^T \mathbf{Q} \Rightarrow \mathbf{c}_1^T \mathbf{X} = \mathbf{c}_2^T \mathbf{X}. \quad (3.2)$$

where \mathbf{Q} and \mathbf{X} denote respectively the discrete spectral sensitivities of the camera and the XYZ CMFs. The columns of matrices \mathbf{Q} and \mathbf{X} represent the spectral sensitivity for each sensor channel and the rows represent the sensor responses at a sampled wavelength.

Equation (3.2) implies that $\mathbf{c}_1^T - \mathbf{c}_2^T$ is simultaneously in the null space of \mathbf{Q} and \mathbf{X} . Since any spectrum in the null-space of \mathbf{Q} is a physically plausible spectral difference, this implies that the null-spaces of \mathbf{Q} and \mathbf{X} are the same and this in turn implies the Luther condition [52]:

$$\mathbf{X} = \mathbf{Q}\mathbf{M} \quad (3.3)$$

where \mathbf{M} is a 3×3 full rank matrix mapping the camera sensitivities to the human visual system. See [36] for the original proof of the Luther condition (which we precis above).

3.3 Luther-Condition Filter Optimisation

We propose a modified Luther condition where a camera is said to be colorimetric if there exists a physically realisable filter which, when placed in front of the camera, generates *effective* sensitivities which are a linear transform from the XYZ CMFs.

Let us rewrite (3.1) in the discrete domain. First, let us establish the notation. Let \mathbf{Q} and \mathbf{X} denote respectively the spectral sensitivities of a digital camera and the CIE XYZ colour matching curves and both are 31×3 matrices. Let a 31-vector \mathbf{f} denote the sampled equivalent of the continuous filter function $f(\lambda)$. Let $\text{diag}(\mathbf{f})\mathbf{Q}$ denote the discrete equivalence of $f(\lambda)Q(\lambda)$. The $\text{diag}()$ operation converts a vector into a diagonal matrix (the vector components appear along the diagonal). Mathematically, \mathbf{f}_i , the i th component in \mathbf{f} multiplies the i th row in \mathbf{Q} . That is, the three camera sensitivities at the i th wavelength are multiplied by the same value. The diagonal matrix operator allows us to express component-wise multiplication.

Specifically, using the notation, we propose that the filter that best matches the Luther condition can be found by minimising:

$$\min_{\mathbf{f}, \mathbf{M}} \| \text{diag}(\mathbf{f})\mathbf{Q}\mathbf{M} - \mathbf{X} \|_F^2 \quad \text{s.t. } \mathbf{f} > 0. \quad (3.4)$$

\mathbf{M} is a 3×3 correction matrix. We minimise the square of Frobenius norm $\| \cdot \|_F^2$ (i.e. the sum of squares error) subject to the constraint that the filter value is larger than 0 (physically, we cannot have a filter that has negative transmittance).

We do not have to constrain the maximum transmittance because we can only solve for \mathbf{f} and \mathbf{M} up to an unknown scaling factor. Indeed, suppose the filter \mathbf{f} is returned where the max transmittance is larger than 1. The fitting error in (3.4) is unchanged if we divide \mathbf{f} by its maximum value (resulting in a max transmittance of 100%) so long as we multiply the corresponding correction matrix \mathbf{M} by the same value.

3.3.1 Algorithm for Solving the Luther Optimisation

However, there is no closed form solution to (3.4). Rather we solve for \mathbf{f} and \mathbf{M} using a technique called Alternating Least-Squares (ALS) regression. Promisingly the alternating least-squares method is guaranteed to converge (although not necessarily to the global optimum) [106].

To ease notation, we define $\mathbf{F} = \text{diag}(\mathbf{f})$ and write $\mathbf{F}\mathbf{Q}$. When we wish to extract the vector \mathbf{f} from the diagonal matrix \mathbf{F} , we write $\mathbf{f} = \text{ediag}(\mathbf{F})$ ('e' signifies to 'extract' the diagonal). The algorithm is shown in Algorithm 3.1.

Algorithm 3.1 ALS algorithm for Luther-condition optimisation

- 1: $i = 0, \mathbf{F}^0 = \mathbf{F}^{initial}, \mathbf{M}^0 = I_3$
 - 2: **repeat**
 - 3: $i = i + 1$
 - 4: $\min_{\mathbf{M}^i} \| \mathbf{F}^{i-1}\mathbf{Q}\mathbf{M}^i - \mathbf{X} \|_F^2$
 - 5: $\min_{\mathbf{F}^i} \| \mathbf{F}^i\mathbf{Q}\mathbf{M}^i - \mathbf{X} \|_F^2$
 - 6: **until** $\| \mathbf{F}^i\mathbf{Q}\mathbf{M}^i - \mathbf{F}^{i-1}\mathbf{Q}\mathbf{M}^{i-1} \|_F^2 < \epsilon$
 - 7: $\mathbf{f}^{Luther} = \text{ediag}(\mathbf{F}^i)$ and $\mathbf{M} = \mathbf{M}^i$
-

The operator \mathbb{I} denotes matrix multiplication. In ALS algorithm, we make an initial guess of the filter $\mathbf{F}^{initial}$ and solve for \mathbf{M} (Step 4). Then we hold \mathbf{M} fixed and solve for \mathbf{F} (Step 5). Taken together we estimate a new 'corrected' (partial solution) spectral sensitivities. We repeat this process until convergence (the method is guaranteed to converge). The filter and then the linear transform - are solved using simple, closed-form

least-squares estimation. For completeness we provide details of how these calculations are made in the Appendix A.1.

At each iteration, the filter and linear transform - \mathbf{F}^i and \mathbf{M}^i - are calculated relative to the previous $i - 1$ filters and matrices. It follows in Step 7 that the final solution is the multiplication of all the per-iteration solutions: $\mathbf{f}^{Luther} = eddiag(\prod_{s=1}^i \mathbf{F}^s)$ (where $eddiag()$ means to extract the matrix diagonal and forms a vector) and $\mathbf{M}_j = \prod_{s=1}^i \mathbf{M}^s$.

3.3.2 Filter Positiveness

Notice nowhere in the above procedure do we constrain the filter transmittance to be greater than 0 (even though this constraint is in the optimisation statement). Empirically, we found that the optimised filter is always positive for all the cameras we tested (see the experimental results in Section 3.4). Moreover, we prove in Theorem 3.1 presented below that when there exists a filter which makes the camera sensors perfectly colorimetric that the filter has to be everywhere positive.

The theorem is presented for continuous spectral sensitivity functions. As such, we write the XYZ CMFs and camera sensitivities as vector functions: $\underline{X}(\lambda)$ and $\underline{Q}(\lambda)$.

Theorem 3.1. *If there exists an exact solution $f(\lambda)$ for $\mathbf{M}^T \underline{Q}(\lambda) f(\lambda) = \underline{X}(\lambda)$, and the variable λ is defined over the domain where $X(\lambda) > 0$ and $\underline{Q}(\lambda)$ are continuous (and full rank), then $f(\lambda) > 0$.*

Proof. First we remark on the continuity of the camera and XYZ functions. Both are the result of physical processes which are continuous in nature. To our knowledge, it is not possible to make a physical sensor system that captures light which has discontinuous sensitivities. And, in terms of physiological systems, biological sensor response functions are always continuous.

Next, if $f(\lambda) < 0$ across all wavelengths and $\mathbf{M}^T \underline{Q}(\lambda) f(\lambda) = \underline{X}(\lambda)$, then $-\mathbf{M}^T \underline{Q}(\lambda) * (-f(\lambda)) = \underline{X}(\lambda)$. In this case $-f(\lambda)$ must be all positive and so an all-positive filter can be found. The interesting case to consider is when the filter has both negative and positive values.

Clearly the 3×3 matrix \mathbf{M}^T must be full rank otherwise the mapped camera sensitivities would be rank deficient and therefore could not model the CMFs. Equally, multiplying by a filter does not change the rank of the sensor set. Because, by assumption $\underline{Q}(\lambda)$ are continuous it follows that $f(\lambda)$ must also be a continuous function since otherwise

$\mathbf{M}^T \underline{Q}(\lambda) f(\lambda)$ would be discontinuous (multiplying a continuous and discontinuous functions together, save for the case where one of the functions is everywhere 0, results in a discontinuous function).

As $f(\lambda)$ is continuous if the function has both negative and positive values there must be at least one wavelength λ_v where $f(\lambda_v) = 0$ and so $\mathbf{M}^T \underline{Q}(\lambda_v) f(\lambda_v) = \underline{0}$. But, this cannot be the case since the XYZ colour matching functions are not all zero at any given wavelength within the defined domain. \square

Of course, the theorem is written in the continuous domain. How does this theorem work given discrete data, e.g. 10 nm samples across the visible spectrum? Pragmatically, we have found that the recovered filters are always all positive, for the discrete case. And, of course – by linear interpolation, for example, we can turn non-continuous data into continuous functions. Then, by the theorem if there is an exact filter correction then this filter must be all positive. And, the filter must also *work* at the original sample points (be all positive here too). Remember, the theorem does not teach how to find the all-positive filter only that if exact colour correction is possible the filter must be all positive. For linearly interpolated discrete data the filter must be all positive.

3.4 Results

3.4.1 Luther-filters

The optimal Luther-filters for a group of 28 commercial cameras (the camera data set described in Section 2.3) are derived. The camera spectral sensitivities are known a priori. These cameras have a clear deviation from the Luther condition, see [41] for further detail. For each camera, a specific filter and a linear transformation were calculated solved using Algorithm 3.1 developed in the last section.

Figure 3.2c displays an example of colour filter calculated for a Canon 40D camera whose spectral sensitivities are plotted in Figure 3.2a. The relative spectral transmittance values are positive in the range between 0 and 100%. Multiplying the Canon 40D sensitivities by this filter (and linearly fitting to the XYZ colour matching functions), we arrive at the comparison shown in Figure 3.2d (solid for the XYZ colour matching functions and dotted for the approximation). The filter corrected sensitivities are almost the same as the XYZ colour matching functions. In contrast, Figure 3.2b shows the native camera spectral sensitivities fitted directly to the XYZs. Visually, we can see that with the addition of our derived filter, the closeness of the spectral sensitivity fit is remarkable and thus makes the camera much more colorimetric.

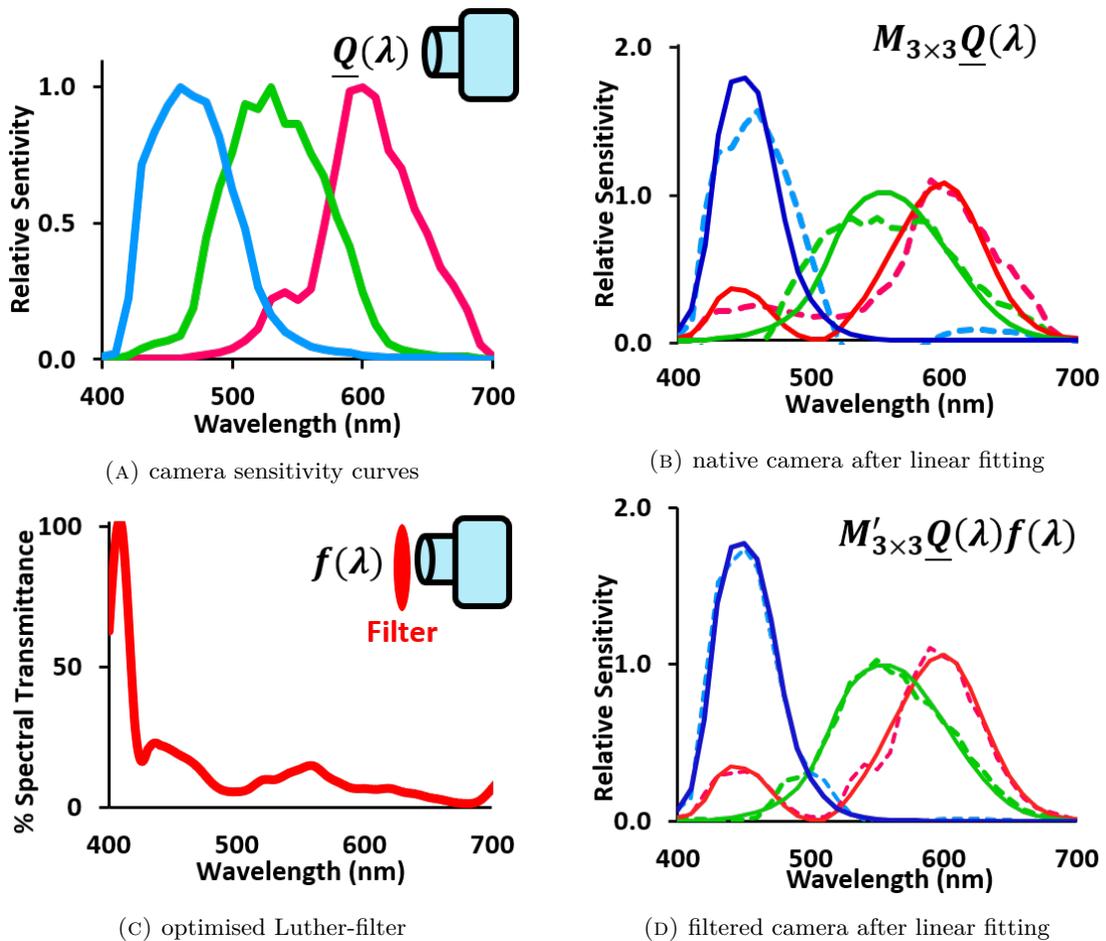


FIGURE 3.2: (A) shows the spectral sensitivities of a Canon 40D camera. (B) shows the least-squares fit of the camera curves (in dashed lines) to XYZ colour matching functions (in solid lines). With a specially designed colour filter shown in (C), the least-squares fit of the camera spectral sensitivity curves multiplied by the filter - shown in (D) - is much closer to the XYZ colour matching functions.

Readers will notice that the filter shape has a steep gradient at the short wavelengths and so likely not easy to manufacture. Further, across most of the wavelengths the filter absorbs most of the light. This need not be a problem for colour measurement as we can increase exposure time, for example. Though, it does mean that the camera with and without the filter would, for the same recording conditions, capture significantly less light. And, this could result in an increase in the noise in the final image output by a camera reproduction pipeline. Indeed, if we deploy this filter - and keep the capture conditions the same - we would need to ‘scale up’ the recorded values and this operation also scales up the noise. Effectively, we capture an image at a higher ISO number. The role of noise and the shape of the filter will be addressed in Chapter 6.

TABLE 3.1: Performance of cameras with **LUTHer**-condition optimised filters versus the **NAT**ive sensitivities with no filter applied in terms of Vora-Value and NRMSE (note that higher Vora-Value but lower NRMSE value indicate better performance)

Individual Camera	Vora-Value		NRMSE	
	NAT	LUTH	NAT	LUTH
Canon 40D	0.932	0.986	0.298	0.062
Nikon D90	0.888	0.945	0.374	0.104
Canon group	0.938	0.987	0.285	0.062
Nikon group	0.921	0.944	0.316	0.108
Whole camera set	0.918	0.961	0.317	0.092

3.4.2 Evaluation of Spectral Fitting

To quantitatively measure the *spectral* match between the filtered and linear transformed camera sensors and the desired XYZ colour matching functions, we adopt the measures of Vora-Value [88] and the Normalised Root-Mean-Square Error (NRMSE).

The Vora-Value measures the closeness between the spaces of two imaging systems. Given a set of filter sensitivities set \mathbf{Q} and the colour matching functions \mathbf{X} , it is defined as

$$\nu = \frac{1}{3} \text{trace}(\mathbf{Q}\mathbf{Q}^+\mathbf{X}\mathbf{X}^+) \quad (3.5)$$

where $\text{trace}()$ is a matrix operator that sums up the terms along the diagonal of a matrix and $^+$ is the Moore-Penrose inverse, explicitly $\mathbf{Q}^+ = (\mathbf{Q}^T\mathbf{Q})^{-1}\mathbf{Q}$. We refer the reader to [88] where the derivation and significance of Vora-Value as a useful tool for quantifying the colour measurement potential of a set of colour filters are justified.

The Vora-Value returns a number between 0 and 1 meaning respectively not colorimetric at all and 100% colorimetric (that the Luther condition is fully satisfied). While there is not a guide on what different Vora-Values *mean*, generally a high Vora-Value indicates a better colour performance, see Figure 3.3b where the correlation between the Vora-Values and mean colour errors is shown. Empirically we found that a sensor system with a Vora-Value above 0.9 captures colours that a 3×3 matrix can correct to XYZs to a tolerable perceptual error (e.g. the colours will look mostly correct and the visual error will be acceptable). A Vora-Value above 0.99 is indicative of a camera that is almost colorimetric, i.e. to all practical purposes will sense the real-scene colours like we do.

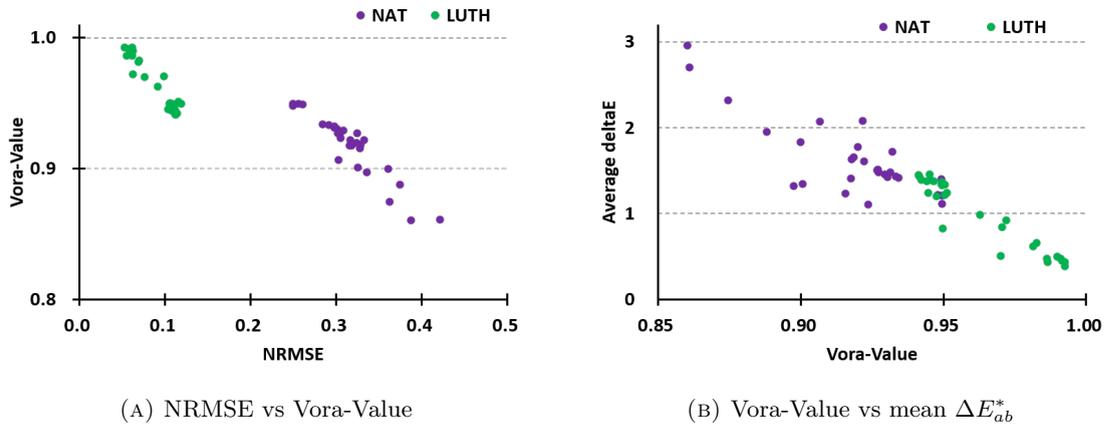


FIGURE 3.3: Correlations between (A) NRMSE and Vora-Value and (B) Vora-Value and mean ΔE_{ab}^* .

We also evaluate spectral fitness in terms of the normalised spectral fitness error and is defined as:

$$NRMSE = \frac{\| \mathbf{X} - \mathbf{Q}\mathbf{Q}^+\mathbf{X} \|_F}{\| \mathbf{X} \|_F}. \quad (3.6)$$

where we calculate the the Frobenius norm of the spectral difference between the corrected camera sensitivity functions and the desired CMFs and normalised by that of the CMFs.

Let us first look at some camera examples before reviewing the results collectively. In Table 3.1, we look at the Vora-Value and NRMSE performance - with and without their optimised Luther-filters solved using Algorithm 3.1 - after the linear correction. Note that Vora-Value and NRMSE have an inverse correlation, see Figure 3.3a. In our study, we aim for higher number for Vora-Value but lower number for NRMSE.

In the table, **NAT** denotes the native camera without a filter and **LUTH** denotes the camera with a Luther-optimised filter. The first two rows of the table show the performance of 2 typical DSLR cameras each from a well-known brand. Then we look at the average performance of the Canon subgroup (with 10 Canon cameras) and Nikon subgroup (with 9 Canon cameras). Finally, the average performance over the whole data set of 28 cameras is summarised. We repeat this methodology for NRMSE. In terms of Vora-Value, the Canon 40D camera which was originally 0.932 becomes 0.986. The NRMSE reduces from 0.2980 to 0.062, reduced by nearly 80%. By applying this colour filter and linear mapping, Canon 40D camera becomes almost colorimetric. As for Nikon D90, good improvement is made as the NRMSE is reduced by two thirds, yet the Vora-Value is not improved as remarkably as Canon 40D. Taken as a subgroup, Canon cameras can be filter-corrected to become colorimetric more readily than Nikon cameras.

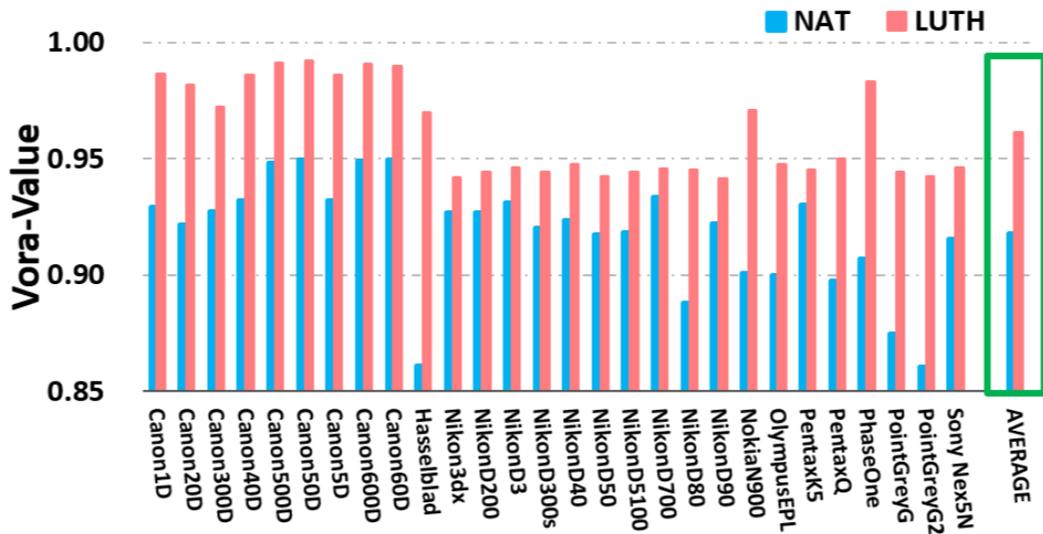


FIGURE 3.4: Spectral match to the Luther condition by unfiltered **NAT**ive camera sensitivities and filtered **LUTH**er-condition optimised sensitivities for a group of 28 cameras in terms of Vora-Value. The overall average Vora-Values of the camera group for both conditions are shown in the green box.

The corrected Canon cameras have Vora-Values of 0.972 at least, with an average value as high as 0.987 for the whole subset compared to 0.944 of the Nikon subgroup.

Collectively, the Vora-Value performance for a set of 28 digital cameras [41] - with and without their optimised Luther-filters - are shown in Figure 3.4. The Vora-Value for the unfiltered, **NAT**ive sensitivities are shown in blue and for the **LUTH**er-condition optimised sensitivities in red. The average results across the camera set is listed in the last bar within the green box. On average, the native Vora-Value is 0.918 but when the optimised filter is added it increases to 0.961. We can conclude that spectrally all cameras are improved to some extent in terms of NRMSE and Vora-Value.

3.4.3 Colour Measurement Experiment

Now let us evaluate the derived Luther-condition filters (solved using Algorithm 3.1) in terms of a perceptually relevant colour measurement metric. The error metric ΔE_{ab}^* [65] calculates the distance between two colour vectors in the CIELAB colour space and returns a single number that roughly correlates with the perceived difference between two colours. One ΔE_{ab}^* corresponds approximately to a ‘Just Noticeable Difference’ to a human observer [103]. When ΔE_{ab}^* is less than 1, we say that the difference is indiscernible to the visual system.

Here we wish to evaluate how well the RGBs measured by a camera can be corrected to match XYZs and how much this colour correction performance is improved when

TABLE 3.2: ΔE_{ab}^* statistics of the colour corrected **NAT**ive camera and the colour corrected camera with the **LUTH**er-condition optimised filter for Canon 40D camera under different lighting cases

	Mean	Median	90%	95%	99%	Max
CIE D65						
NAT	1.65	1.03	3.55	4.94	11.23	19.29
LUTH	0.46	0.25	1.09	1.45	3.49	5.90
CIE A						
NAT	2.30	1.44	4.65	6.17	16.96	26.41
LUTH	0.64	0.40	1.33	1.84	4.75	8.19
102 illuminants						
NAT	1.72	1.02	3.68	5.12	12.94	28.39
LUTH	0.53	0.30	1.15	1.65	4.11	6.83

a coloured filter is placed in front of the camera. For each camera (native and filter corrected), we calculate the RGBs under a single light of CIE D65 and CIE A standard lights [65], as well as a collection of 102 lights, for the SFU reflectance set of 1995 spectra [6]. We then find the optimal least-squares linear correction to best map to ground-truth XYZs. We then calculate the mean, median, 90 percentile, 95 percentile, 99 percentile, and maximum of ΔE_{ab}^* over the test data set.

3.4.3.1 Single-light Case

Let us use the Canon 40D camera spectral sensitivities as a putative measurement device and quantify how well it can measure colours - with and without a filter. In this experiment the measurement light is either a CIE D65 (bluish) or a CIE A (yellowish) illuminant. For reflectances, we use the SFU set of 1995 spectra [6] (itself a composite of many widely used sets). The 1995 XYZs for these reflectances under the specific light are the *ground-truth* with respect to which we measure colour error.

Using the Canon camera sensitivities, the reflectance spectra and either the CIE D65 and A lights we numerically calculate two sets of 1995 RGBs. Now, we linearly regress the RGBs for each light to their corresponding ground-truths (we map the native RGBs for CIE D65 and A to respectively the XYZs under the same lights). We call these colour corrected RGBs the **NAT**ive camera predictions (and we adopt this notation in the results shown in Table 3.2). Row 1 and Row 3 of Table 3.2 report the mean, median,

90, 95 and 99 percentile and maximum CIELAB ΔE_{ab}^* errors for CIE D65 and CIE A lights.

Now let us place a filter in front of the camera. Again, we calculate two sets of RGBs (one for each light) for the camera spectral sensitivities multiplied by the filter found using the Luther-condition optimisation (Algorithm 3.1). The recorded filtered RGBs are mapped best to corresponding XYZs using linear regression. The **LUTHER** ΔE_{ab}^* error statistics are shown in Row 2 and Row 4. It is clear that placing a Luther-condition filter substantially increases the ability of the camera to measure colours accurately. Across all metrics the ΔE_{ab}^* errors reported are about a third of those found when a filter is not used.

3.4.3.2 Multiple-light Case

We now repeat the experiment for a set of 102 measured lights [6]. The results of this second experiment are shown in the last 2 rows of Table 3.2. Here, the reported error statistics are averages. For each illuminant - as described in the single light case above - we calculate the mean, median, 90 percentile, 95 percentile, 99 percentile and maximum ΔE_{ab}^* . That is, we calculate six error measures for 102 lights. We then take the mean of each error statistic over all the lights.

In terms of the reported errors of the raw RGBs compared to the filtered RGBs for the Luther-filters, we see the same data trend for the multiple lights case as we saw previously for single lights. A Luther-condition derived filter reduces the measurement error by 2/3, on average.

3.4.3.3 Multiple Cameras

Now, we calculate the mean ΔE_{ab}^* error (for the 102 lights and 1995 reflectances) for each of 28 cameras. For each camera, we calculate the optimal Luther-filters. Per camera, Figure 3.5 summarises the mean and 95 percentile ΔE_{ab}^* performance. Grey bars show, respectively, the mean and 95 percentile error performance of **NATive** (unfiltered) colour corrected RGBs for the 28 cameras. Respectively, the green lines record the performance of the best Luther-filters.

It is evident that, whichever statistic is employed, the optimised filters support more than half of the cameras in the testing group with significant improvements in their colour measurement ability. This is an interesting result since it shows that within the range of manufacturable sensors there are sets which can be made much more colorimetric with

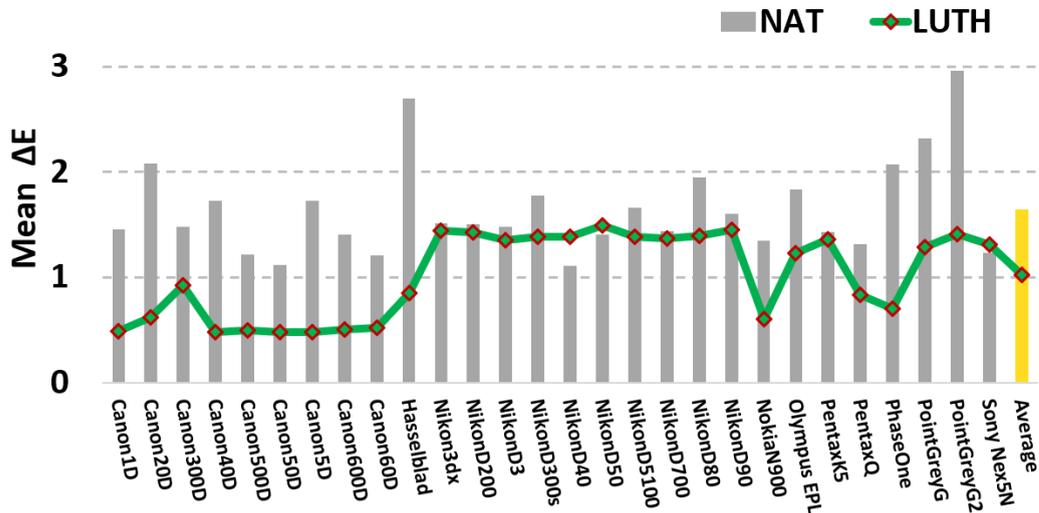
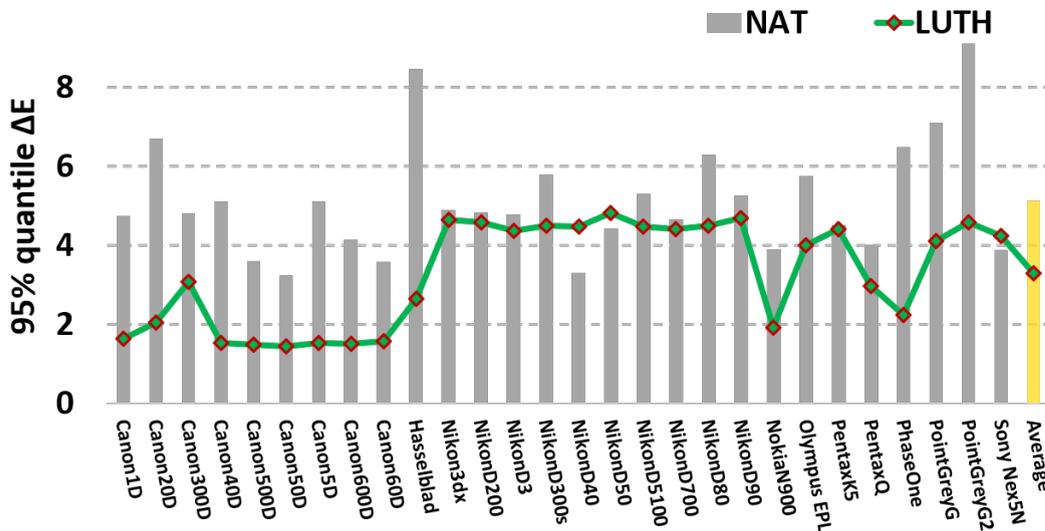
(A) mean ΔE_{ab}^* colour error(B) 95-percentile ΔE_{ab}^* colour error

FIGURE 3.5: (A) Mean and (B) 95-percentile ΔE_{ab}^* errors for 28 cameras. The grey-bars show the colour errors for **NATIVE** colour correction. The green lines with red diamonds show the results by using the **LUTHER**-condition optimised filters.

the addition of a coloured filter. And these cameras could then be used in applications where accurate colour measurement is needed.

Additionally, the Luther-filter method works particularly well on some of the cameras. However, when reviewing the cameras individually, a few cameras like Nikon D40 and Sony Nex5N, are not improved by a prefilter in terms of the colour errors, despite the fact that all cameras better approximate the Luther condition (with smaller NRMSE and higher Vora-Value) with optimised filters. There are two possible reasons. Firstly, our filters are not straightly designed to minimise the colour errors in the CIELAB colour space. A better approximation to the Luther condition does not necessary improves the

performance in terms of colour errors. Furthermore, in the Luther-condition method, we focus on the sensitivity functions of the cameras and assume that the colour spectra that the camera measures are equally distributed (or mathematically, they are identical and independent distributed). However, the real-world colour signals are not arbitrary which will result in the failure in terms of the colour performance. We will show in Chapter 5 that when the real spectra are taken into account, these cameras become significantly more colorimetric.

Discussion

Note, we make a distinction between using a camera for colour measurement and for making visually pleasing images. Here, we are interested in using a camera to measure XYZ tristimuli (or measures like CIE Lab values that are derived from tristimuli [65]). From a measurement perspective, we need higher tolerances and a higher Vora-Value. Clearly, from the point of view of making attractive images cameras that have Vora-Values less than 0.95 can work very well. Indeed, many of the 28 cameras with Vora-Values less than 0.95 can still take images that look appealing from a photographic perspective. But, commercial cameras are not suitable vehicles for accurate colour measurement.

3.5 Conclusion

In this chapter, we develop a method to find the best coloured filter such that when placed in front of a camera, its new effective spectral sensitivities best match the Luther condition (and become most colorimetric). Experiments demonstrate that applying our optimised filters to the testing cameras provides a step change in how well a camera can measure colour and often makes a camera ‘almost’ colorimetric. By adding such a filter for a digital camera can result in a dramatic improvement in its ability to see the world colorimetrically.

The following chapter will design the colour filter using the Vora-Value as the optimisation criterion, a more general measure that exploits the geometric perspective of the sensitivity ‘closeness’ between a camera and the human visual system. We will also show the mathematical connection between the Luther condition and the Vora-Value.

Chapter 4

Vora-Value Filter Design and its Relation to Luther Condition Filter Optimisation

In Chapter 3, we solve for a colour filter that when placed in front of the camera would make the camera best satisfy the Luther condition. Curiously, the Luther-filter optimisation solves for a filter using the XYZ colour matching functions (CMFs) of the human visual system as the reference sensitivity set. However, depending on the target sensitivity set (any linear combination of the CMFs can be an eligible option, such as the cone fundamentals), a different optimal filter is found. The fact that there are multiple optimal answers, perhaps, points to a weakness in the optimisation statement? In this chapter, we will seek a filter by using the Vora-Value [88], under which best matches all possible target spectral sensitivity sets (e.g. any linear combination of the XYZ CMFs).

4.1 Introduction

The Vora-Value is a goodness measure for a set of sensitivity functions for colorimetric measurement. For the camera application, it measures the similarity between the vector spaces spanned by the spectral sensitivities of a camera and the XYZ colour matching functions underpinning the human visual system.

We visualise the concept in Figure 4.1. In blue, we show the vector space spanned by the sensitivity functions of the Human Visual Space (HVS). These sensitivities might be the cone fundamentals or equivalently the XYZ CMFs (we say ‘equivalently’ because these

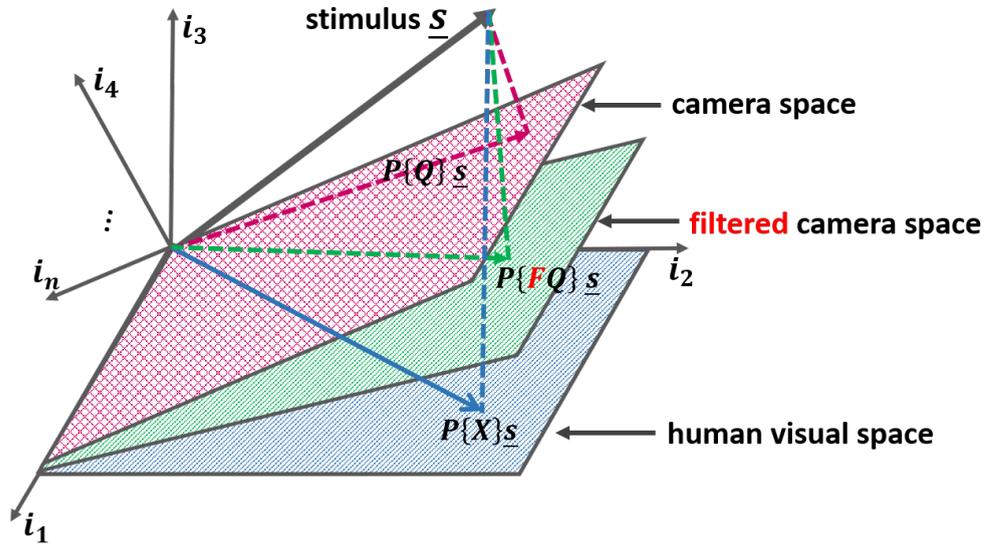


FIGURE 4.1: A visualisation of the vector spaces of camera (in red), filtered camera (in green), and the human visual system (in blue). A colour stimulus is projected to each of the space. We aim for the filtered camera space to be as closely as possible to the human visual space such that the colour it records can better predict that of the human visual response.

sensitivity sets are linearly related [20, 80] and both sets span the same vector space). Similarly, the camera spectral sensitivities - shown in red - span a different vector space.

When we place a filter in front of a camera, the new effective spectral sensitivities span a new vector space, shown in green in Figure 4.1. Our aim is to set forth a method for finding a filter that makes the filtered sensitivities span a subspace (the green vector space) as close as possible to the human visual space (the blue vector space). The Vora-Value optimisation has the advantage that the best filter is related to the target sensitivity basis and not fixed coordinates (e.g. the XYZ CMFs) with respect to that basis.

Let's start to make some of this intuition more concrete. First, let us introduce a little notation. We denote respectively the spectral sensitivities of the camera and the human visual sensors as matrices \mathbf{Q} and \mathbf{X} . Again, these two matrices are 31×3 matrices with each row of sensitivity responses at a given wavelength. Note the columns of \mathbf{X} could be the CIE XYZ colour matching functions or any function set thereof.

The blue and red planes in Figure 4.1 - for the human visual system and camera sensitivities, respectively - visualise the vector spaces spanned by \mathbf{Q} and \mathbf{X} . By span we mean the set of all spectra we can generate by taking linear combinations of the columns of \mathbf{Q} (or \mathbf{X}). The matrix \mathbf{Q} (or \mathbf{X}) spans a 3-dimensional vector space contained in the 31-dimensional space. So, how do we evaluate the closeness (or otherwise) of two vector spaces spanned by the column vectors of \mathbf{Q} and \mathbf{X} ?

Usually, when we compare quantities, they are in *correspondence*. So far the idea of vectors spanning a subspace has been quite abstract. The key to comparing vector spaces is defining a method of placing spectra generated in one vector space in correspondence to matching spectra in another. Let us generate a 31-dimensional spectrum \mathbf{s} . Returning to Figure 4.1, for the stimulus \mathbf{s} , we find the corresponding (projected) spectra, \mathbf{s}_Q and \mathbf{s}_X in the vectors spaces spanned by \mathbf{Q} and \mathbf{X} that are closest to \mathbf{s} . We denote the projection of \mathbf{s} onto the camera vector space \mathbf{Q} as $\mathbf{s}_Q = P\{\mathbf{Q}\}\mathbf{s}$ [16]. Similarly, we denote the spectral sensitivities of a camera with a filter, \mathbf{F} , placed in front as \mathbf{FQ} and the corresponding projected spectrum as $\mathbf{s}_{FQ} = P\{\mathbf{FQ}\}\mathbf{s}$. Finally, the corresponding projection onto the human visual space \mathbf{X} is denoted $\mathbf{s}_X = P\{\mathbf{X}\}\mathbf{s}$.

And, given these corresponding spectra, we can calculate their distance: $\|\mathbf{s}_Q - \mathbf{s}_X\|_2$. By generating lots of random spectra - the values of which are independent and identically distributed - drawn from 31-dimensional space, we can calculate the mean difference between their *projections* to the spaces spanned by \mathbf{Q} and \mathbf{X} . This projected spectral error (after normalisation) - calculated over all possible spectra - is our definition of subspace similarity.

This reasoning leads to a simple, closed-form, vector space similarity formula that calculates a number in $[0,1]$. Intuitively, it is kind of a percentage measure of how similar two vector spaces are. Actually, to this point, our reason is (perhaps with a slightly different emphasis) a retelling of the derivation of the Vora-Value [88]. The Vora-Value is a measure of how well a camera's RGBs can be (linearly) corrected to colorimetric counterparts (under the so-called Maximum Ignorance Assumption [27] which posits that all spectra are equally likely).

In colour science, the Vora-Value is a single number in the interval of $[0,1]$ which reports how close a camera's spectral sensitivities are to the human visual subspace (HVS) [87]. We will utilise the Vora-Value as an optimisation criterion to design the optimal filter for a given camera and is solved by using a gradient ascent algorithm. We also recast the gradient-ascent Vora-Value approach so that we can use Newton's method [11] (which includes 2nd derivatives) to speed up the optimisation.

Experiments demonstrate that the Vora-Value optimised colour filter can dramatically increase the colorimetric accuracy across a large set of commercial cameras.

From an algorithmic viewpoint, the Luther and Vora-Value optimisations are quite different from one another. The Luther condition optimisation is solved using a simple alternating least-squares procedure which converges quickly. The Vora-Value optimisation, however, has a more complex derivation for the gradient ascent approach. The

Vora-Value optimisation converges less quickly, though arguably to a better overall answer (because of the basis independence property of the Vora-Value).

Apart from the Vora-value filter optimisation, another important result we will present in this chapter is to prove that if we choose to optimise the Luther condition using not the XYZ CMFs but a linear combination which is orthonormal, then we are in fact optimising the Vora-Value. This means we can use the quicker-to-converge Alternating Least-Squares (ALS) optimisation to find the best Vora-Value filter. An additional advantage of the ALS method is that it is easier to add constraints into the problem formulation than to do so for the Vora-Value approach, which will be further explained in Chapter 6.

The rest of the chapter is structured as follows. In Section 4.2, we review the definition of the Luther condition and Vora-Value. There, we also introduce the concept of orthonormal basis which will underpin our optimisation methods. In Section 4.3, we show how we design our filter that the new effective camera system maximises the measure of Vora-Value and also the algorithm used to solve for the optimal filter. In Section 4.4, we show the close relation between the Luther- and Vora-Value optimisations based on which a unified filter design method is proposed. After a simple modification in the optimisation, the discovered Luther-filter is also the filter that maximises the Vora-Value. Experimental results are reported in Section 4.5. Section 4.6 concludes the chapter.

4.2 Background

4.2.1 The Luther Condition under Linear Transform

The Luther condition states that the spectral sensitivity curves of the camera sensors are a linear combination of the CIE XYZ CMFs. Let $\mathbf{Q} = [\mathbf{r}, \mathbf{g}, \mathbf{b}]$ and $\mathbf{X} = [\mathbf{x}, \mathbf{y}, \mathbf{z}]$ denote respectively the spectral sensitivities of the camera and the CMFs of the human visual sensors. Mathematically, the Luther condition is written as

$$\mathbf{X} = \mathbf{QM} \tag{4.1}$$

where \mathbf{M} is a 3×3 matrix. Equivalently we write:

$$\mathbf{XT}_1 = \mathbf{QT}_2\mathbf{M}' \tag{4.2}$$

where \mathbf{T}_1 , \mathbf{T}_2 and \mathbf{M}' are 3×3 full rank matrices. Clearly, given (4.1), then $\mathbf{M}' = (\mathbf{T}_2)^{-1}\mathbf{MT}_1$. That is, the Luther condition does not depend on the particular basis used; we can map

camera sensitivities to XYZs, cone functions or any linear combination thereof. This is a key property that we will return to in Section 4.4.

4.2.2 The Vora-Value

The Vora-Value is often used to measure how similarly a camera samples the spectral world compared to the human visual system. Mathematically, it measures the geometric closeness between the vector spaces spanned by the spectral sensitivity functions of a camera and a human observer. The Vora-Value is a number between 0 and 1. When the Vora-Value is 1, the camera subspace shares the same vector space with the human visual subspace and this means the camera is colorimetric. Therefore the camera RGBs can be corrected to the XYZ tristimulus values within a linear transform. Contrastingly, a Vora-Value of 0 means the two vector spaces are perpendicular to each other. It means the camera is completely unsuited for the colour measurement task.

Given a camera sensor set \mathbf{Q} and the trichromatic human visual sensors \mathbf{X} , the Vora-Value is defined [90] as

$$\nu(\mathbf{Q}, \mathbf{X}) = \frac{1}{3} \text{trace}(\mathbf{Q}(\mathbf{Q}^T \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T), \quad (4.3)$$

Again the superscripts T and $^{-1}$ denote the matrix transpose and inverse, respectively, and $\text{trace}()$ returns the sum of the elements along the diagonal of a matrix

This representation can be written in a more compact way when we used the notation of matrix projector. The projector of a matrix, such as \mathbf{Q} , is defined as

$$\mathbf{P}\{\mathbf{Q}\} = \mathbf{Q}(\mathbf{Q}^T \mathbf{Q})^{-1} \mathbf{Q}^T, \quad (4.4)$$

thus we can rewrite the Vora-Value in a more compact way as

$$\nu(\mathbf{Q}, \mathbf{X}) = \frac{1}{3} \text{trace}(\mathbf{P}\{\mathbf{Q}\} \mathbf{P}\{\mathbf{X}\}) \quad (4.5)$$

where $\mathbf{P}\{\mathbf{Q}\}$ and $\mathbf{P}\{\mathbf{X}\}$ denote the projection matrices respectively of the camera spectral responses and the human visual sensitivities, respectively.

Projector matrices often occur in least-squares regression. For example, finding \mathbf{y} in the classic regression problem $\mathbf{A}\mathbf{y} \approx \mathbf{b}$ has a closed form solution. The linear combination of the columns of \mathbf{A} that solves the least-squares regression is written as: $\mathbf{y} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$, which makes the least-squares approximation $\mathbf{A}\mathbf{y} = \mathbf{A}(\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} = \mathbf{P}\{\mathbf{A}\} \mathbf{b}$ (the projector of \mathbf{A} multiplied by \mathbf{b}).

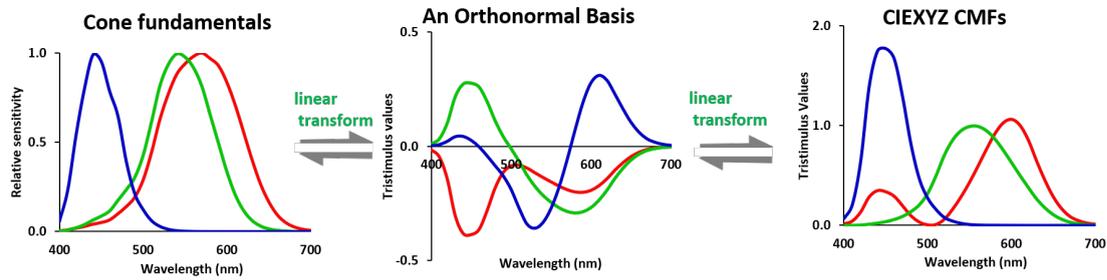


FIGURE 4.2: Three sensitivity sets: cone fundamentals, a set of orthonormal basis, and the CIE 1931 XYZ colour matching functions. These sensitivity sets are all linearly related and span the same Human Visual Space.

4.2.3 Orthonormal Basis

In an n -dimensional vector space, a basis is a set of n linearly independent vectors. Linear independence of the basis ensures that any vector in the vector space can be expressed as a linear combination of the basis vectors (see Figure 4.2). We say the basis spans the vector space. There are many sensitivity sets in a vector space and, by definition, they are all linear transform apart and share the same basis.

Let $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]$ denote a special linear combination of \mathbf{X} that is orthonormal, i.e. $\mathbf{V} = \mathbf{X}\mathbf{T}$ where \mathbf{T} is the linear mapping matrix. Orthonormality implies that its columns are unit vectors and also perpendicular to each other. Mathematically, we write $\mathbf{V}^T\mathbf{V} = \mathbf{I}_3$ (\mathbf{I}_3 is the 3×3 identity matrix) [54]. The orthonormal matrix \mathbf{V} can be obtained by many methods, e.g. the Gram-Schmidt process and QR factorisation [31].

By simple substitution into the matrix projector in (4.4), we can express projector matrices (and the Vora-Value) in a simpler algebraic form:

$$P\{\mathbf{X}\} = P\{\mathbf{V}\} = \mathbf{V}\mathbf{V}^T. \quad (4.6)$$

This suggests that any linearly related matrix shares the same projector matrix. Hence, the Vora-Value formula is, by construction, independent of the basis that describe the camera and the HVS.

Orthogonal basis of the colour matching functions is a useful tool for studying problems in colour science, such as the design of the optimal spectral sensitivities [62, 87, 101], error propagation [54, 87, 107], and metamer colours [16, 17].

4.3 Vora-Value Filter Optimisation

Here we formulate the filter design problem by using the criterion of Vora-Value. The filter-modified Vora-Value optimisation for the effective ‘filter+camera’ system can be written as

$$\arg \max_{\mathbf{F}} \nu(\mathbf{F}\mathbf{Q}, \mathbf{X}) = \frac{1}{3} \text{trace}(\mathbf{P}\{\mathbf{F}\mathbf{Q}\} \mathbf{P}\{\mathbf{X}\}). \quad (4.7)$$

Incorporating the explicit equation for the projector matrix as (4.4), we have

$$\arg \max_{\mathbf{F}} \nu(\mathbf{F}\mathbf{Q}, \mathbf{X}) = \frac{1}{3} \text{trace}\left(\mathbf{F}\mathbf{Q}(\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} \mathbf{X} \mathbf{X}^T\right). \quad (4.8)$$

Note in writing this equation, we made use of the fact that $\mathbf{F}^T = \mathbf{F}$ (diagonal matrices are symmetric).

4.3.1 Gradient Ascent for Solving the Optimisation

So far we have concerned ourselves with formulating the filter design problem, that is, writing down what it is we wish to solve for. Now, we wish to present the mechanics of how we optimise for the filter. Henceforth we will concern ourselves solving for the filter with maximising the Vora-Value in this section. Moreover, presenting the gradient method is a necessary step for us to present a much faster algorithm: the Newton’s method [11].

For a given filter, we can calculate its Vora-Value. The set of all Vora-Values can be thought of as a hilly landscape. By calculating the derivative (how the Vora-Value changes) when the filter is modified in certain directions, we can choose to take a small step in a direction that increases the Vora-Value.

We maximise the objective function in the usual calculus way by first finding the derivative of the Vora-Value function and then updating \mathbf{F} by moving in the direction of the gradient (and in so doing we increase ν). It can be shown - step-by-step derivation details given in Appendix A.2.1 - that the derivative of ν calculated with respect to \mathbf{F} is calculated as:

$$\frac{\partial \nu}{\partial \mathbf{F}_{jj}} = \frac{2}{3} [(\mathbf{I} - \mathbf{P}\{\mathbf{F}\mathbf{Q}\})\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{F}\mathbf{Q}\}\mathbf{F}^{-1}]_{jj}. \quad (4.9)$$

Equivalently, the gradient of the objective function in terms of the underlying filter vector - remember $\mathbf{f} = \text{diag}(\mathbf{F})$ - can be written as

$$\nabla \nu(\mathbf{f}) = \frac{2}{3} \text{diag}\left((\mathbf{I} - \mathbf{P}\{\mathbf{F}\mathbf{Q}\})\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{F}\mathbf{Q}\}\mathbf{F}^{-1}\right) \quad (4.10)$$

where the gradient with respect to the filter vector, $\nabla\nu(\mathbf{f})$, is a 31×1 vector. It is evident that the gradient function has a very interesting structure: it is the diagonal of the product of three projection matrices multiplied by the inverse of the filter at hand. We will come back to this interesting feature later.

Gradient Ascent Algorithm

The simple Gradient Ascent algorithm for finding the filter solution of the optimisation is shown below (see Algorithm 4.1). Note that we use the term ‘gradient ascent’ (but not the more familiar term ‘gradient descent’) because we are maximising our objective function.

Algorithm 4.1 Gradient Ascent for solving the optimisation

- 1: $k = 0$, $\mathbf{f}^0 = \mathbf{f}^{initial}$
 - 2: **repeat**
 - 3: Compute the gradient: $\nabla\nu(\mathbf{f}^k)$
 - 4: Choose a step size: $t^k > 0$
 - 5: Update $\mathbf{f}^{k+1} := \mathbf{f}^k + t^k \nabla\nu(\mathbf{f}^k)$
 - 6: $k \leftarrow k + 1$
 - 7: **until** $\max(|\nabla\nu(\mathbf{f}^k)|) \leq \eta$
 - 8: **return** \mathbf{f}^k .
-

The algorithm works by updating the current filter at step k to a new filter at step $k + 1$. At step k we calculate the gradient and then update according to: $\mathbf{f}^{k+1} = \mathbf{f}^k + t^k \nabla\nu(\mathbf{f}^k)$. The term t^k is the step size (a small positive value) which controls how far the filter should move towards the direction of the gradient. However, it does not teach directly how far we should move that leads to the fastest ascent. If we add too much of the gradient we can step over the maximum and never reach the maximum point. The simplest approach would be to set t^k to a small fixed number. But, in this case, the optimisation will take a long time to converge. To make convergence quicker, we adaptively choose the stepsize. Here the step size changes per iteration and is chosen by the backtracking line search method [11]. We refer the reader to [64] for more details.

By choosing to modify the filter step-wise – at each step seeking to go further up the hill – we modify the filter until we are on the hill top. Of course, there is nothing in the gradient ascent method that guarantees that we will find an optimal solution to the filter design problem. At minimum, we will certainly, arrive at a filter that is better than the initial guess (e.g., the 100% do-nothing filter $\mathbf{f} = [1, 1, \dots, 1]^T$ when the filter is fully transmissive). And, as we present later, we find a very good filter to our problem.

4.3.2 Newton's Method for Solving the Optimisation

Often simple gradient ascent is slow to converge. A better way to tackle the optimisation is to adopt the Newton's method when the second derivative (i.e. the Hessian matrix) of the objective function can be computed. Newton's method uses the curvature information (second derivative) of the objective function and generally is faster to converge, typically it has a quadratic convergence speed compared to linearly for gradient approach. When a neighborhood of the optimal solution is reached, often in just a few iterations a solution with adequately high accuracy can be achieved. Describing the details of how the Newton's method works is beyond the scope of this thesis - but see [64] for a review - we will rather describe how it is used and report the Hessian we calculate for the filter design problem.

The Hessian for the filter design problem is a 31×31 matrix (assuming 31 sample wavelengths) and is denoted $\nabla^2 \mu$. The term ∇_{ij}^2 denotes the partial derivative $\frac{\delta^2 \mu}{\delta \mathbf{f}_i \delta \mathbf{f}_j}$. The Hessian for our problem is calculated as:

$$\begin{aligned} \nabla^2 \nu = & -\frac{4}{3} \left(\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) \circ \left((\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\})\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) + \frac{2}{3} \left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) \circ \left((\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\})\mathbf{P}\{\mathbf{V}\} \right) \\ & + \frac{2}{3} \left((\mathbf{I} - 2\mathbf{P}\{\mathbf{FQ}\})\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) \circ \left(\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) + \frac{2}{3} \left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) \circ \mathbf{I} \end{aligned} \quad (4.11)$$

where \circ denotes the Hadamard product (or elementwise product) of two matrices. The step-by-step derivation is given in Appendix A.2.3.

Newton's Method Algorithm

Newton's method for the problem of filter design is encapsulated in Algorithm 4.2. Here, given an initial guess $\mathbf{f}^{initial}$, we choose a stepsize after calculating the gradient and Hessian. Again step-size is tackled using the backtracking line search procedure [11]. Later we update the filter solution by using its gradient and Hessian: $\mathbf{f}^{k+1} := \mathbf{f}^k - t^k [\nabla^2 \mu(\mathbf{f}^k)]^{-1} \nabla \mu(\mathbf{f}^k)$. The algorithm will keep refine the filter until a stopping criterion is met. Here we simply use an empirically determined maximum iteration number (e.g. 1000) when the algorithm is well converged.

As we will see in the experiment section, the Newton's method converges much more quickly than the gradient ascent. However, the cost of implementation is significant. Here the 31×1 gradient and 31×31 Hessian must be calculated at each step in the iteration. More onerously the inverse of the Hessian must also be calculated (a computationally non-trivial operation).

In fact, the Newton's method we present above has a potential issue. In order to use the Newton's method, the Hessian must be positive definite. But, our Hessian is in fact only positive semi definite. We can update our optimisation to make the Hessian

Algorithm 4.2 Newton's method for solving the optimisation

- 1: $k = 0$, $\mathbf{f}^0 = \mathbf{f}^{initial}$
 - 2: **repeat**
 - 3: Compute the gradient and Hessian: $\nabla\mu(\mathbf{f}^k)$ and $\nabla^2\mu(\mathbf{f}^k)$
 - 4: Choose a stepsize: $t^k > 0$
 - 5: Update $\mathbf{f}^{k+1} := \mathbf{f}^k - t^k[\nabla^2\mu(\mathbf{f}^k)]^{-1}\nabla\mu(\mathbf{f}^k)$
 - 6: $k \leftarrow k + 1$
 - 7: **until** stopping criterion is satisfied
 - 8: **return** \mathbf{f}^k
-

positive definite and this modification is discussed indirectly next (when we look at the relationship between the Vora- and Luther-optimisations). Ultimately, we get to (4.24) and this defines the Hessian that we actually use.

In the next section, we propose a slightly different formalism of the Luther condition optimisation (proposed in Chapter 3). And this has a penalty term on the norm of the filter (we seek a filter the transmittance values of which do not vary too sharply across the visible spectrum). This will also be the key to making the Newton's method work in practice.

4.4 The Modified-Luther Condition Optimisation

Below we reformulate the problem of Luther condition optimisation with three important modifications and will later show how the Luther condition can be connected to the Vora-Value optimisation. The first change we make is to use an orthonormal basis set of the CMFs as the target (rather than the CMFs themselves). This step will help elucidate the similarity between optimising a filter to best meet the Luther condition (The equation in (4.2) assures the Luther condition still holds within linear transform) and to maximise the Vora-Value. Second, we recast the optimisation as a gradient ascent optimisation (and study the Newton formulations). Third, to ensure the well foundedness of the formulation - and to bound the shape of the filter - we add an additional regularisation term that constrains the filter.

4.4.1 Modified Luther Condition Filter Design

As we commented earlier, the Vora-Value and Luther optimisations return different optimal filters. Here we will present a modified Luther condition optimisation which, ultimately, maximises the Vora-Value and so serves to unify the two filter design methodologies.

Mathematically, we formulate the modified Luther-condition optimisation as

$$\arg \min_{\mathbf{F}, \mathbf{M}} \mu(\mathbf{F}, \mathbf{M}) = \|\mathbf{FQM} - \mathbf{V}\|_F^2 + \alpha \|\mathbf{F}\|_F^2 \quad (4.12)$$

where \mathbf{V} denotes an orthonormal basis of the CMFs and the subscript $\|\cdot\|_F$ denotes the Frobenius norm. Note that here we do not constrain the filter to be positive although a physical filter should have positive transmittance values. We remove this constraint because we found that, empirically, our method always returns an all-positive filter solution. Here we use the symbol μ to denote the objective function with the regularisation term. Clearly, $\|\mathbf{FQM} - \mathbf{V}\|_F^2$ is the core optimisation: we wish to find \mathbf{F} (the filter) such that the filtered sensitivities are approximately a 3×3 linear transform \mathbf{M} from the orthonormalised CMFs, here \mathbf{V} . The second term in the optimisation is the squared norm of the filter where $\alpha \geq 0$ is a penalty term defined by the user. As α becomes large, the optimisation must effectively choose a filter whose norm is smaller. Small norm filters tend to be smoother and less jaggy. We say that the scalar α regularises the optimisation.

The regularisation term in the optimisation makes it tilt towards a filter with lower transmittance values over those with higher transmittance values. Indeed, a filter that has a value of 1% at every wavelength has a lower norm than one which transmits 100% of the light at every wavelength. However, we argue that this is not a problem. Remember that in the core optimisation, \mathbf{F} and \mathbf{M} are recovered up to a scaling factor, i.e. $\alpha\mathbf{F}$ and $\frac{1}{\alpha}\mathbf{M}$ is an equally good solution. Viewed through the lens of this scaling indeterminacy, the filter that reflects 1% of light is equally efficacious as the one that transmits 100%. Pragmatically, because we know this scaling indeterminacy exists, we always scale the optimised filter so that its maximum transmittance value is 100%.

Importantly, the regularisation term usefully pushes the optimisation towards recovering smoother filters. A very jaggy filter with an average transmittance of (say) 50% has a much higher norm than one that has a uniform 50% transmittance across the spectrum. Jaggy filters are harder to manufacture so penalizing their discovery in the optimisation makes sense. In Chapter 6, we show how we can add more explicit constraints on the shape and transmittance properties of the filters that we design.

4.4.2 Equivalence of the Modified-Luther and Vora-Value Optimisations

We begin by considering the special case of $\alpha = 0$ in (4.12):

$$\arg \min_{\mathbf{F}, \mathbf{M}} \|\mathbf{FQM} - \mathbf{V}\|_F^2. \quad (4.13)$$

Suppose we are given the filter matrix \mathbf{F} , then, employing the closed-form Moore-Penrose inverse solution to the least-squares [31], the best \mathbf{M} is obtained

$$\mathbf{M} = ((\mathbf{FQ})^T \mathbf{FQ})^{-1} (\mathbf{FQ})^T \mathbf{V}. \quad (4.14)$$

Here we see that the best linear mapping \mathbf{M} in the Luther-condition optimisation of (4.12) is essentially a function of the unknown variable matrix \mathbf{F} . By multiplying this newly solved \mathbf{M} with \mathbf{FQ} , we have

$$\mathbf{FQM} = \mathbf{FQ}((\mathbf{FQ})^T \mathbf{FQ})^{-1} (\mathbf{FQ})^T \mathbf{V} = \mathbf{P}\{\mathbf{FQ}\} \mathbf{V} \quad (4.15)$$

That is, we pre-multiply the orthonormal basis for the XYZs, \mathbf{V} , by the projector for \mathbf{FQ} (see (4.4) for the definition of a projector matrix).

Substituting into (4.13), the optimisation can be rewritten as

$$\arg \min_{\mathbf{F}} \|\mathbf{P}\{\mathbf{FQ}\} - \mathbf{I}\|_F^2 \mathbf{V} \quad (4.16)$$

where \mathbf{I} is the identity matrix. Equation (4.16) has the advantage that the optimisation depends only on the filter \mathbf{F} .

Theorem 4.1. *By minimising $\|\mathbf{P}\{\mathbf{FQ}\} - \mathbf{I}\|_F^2$, we maximise $v(\mathbf{FQ}, \mathbf{X})$.*

Proof. We will use the following identities derived from some basic properties of the matrix trace operator and matrix projection:

1. $\text{trace}(\mathbf{A}^T \mathbf{A}) = \|\mathbf{A}\|_F^2$
2. $\text{trace}(\mathbf{AB}) = \text{trace}(\mathbf{BA})$
3. $\text{trace}(\mathbf{A} + \mathbf{B}) = \text{trace}(\mathbf{A}) + \text{trace}(\mathbf{B})$
4. $\mathbf{P}\{\mathbf{A}\} \mathbf{P}\{\mathbf{A}\} = \mathbf{P}\{\mathbf{A}\}$, idempotency of projectors
5. $\mathbf{P}\{\mathbf{A}\} = (\mathbf{P}\{\mathbf{A}\})^T$, the projector is symmetric.
6. $\text{trace}(\mathbf{P}\{\mathbf{A}\}) = \text{rank}(\mathbf{A})$

Using axiom 1, we rewrite the formula in (4.16) as

$$\arg \min_{\mathbf{F}} \|(P\{\mathbf{FQ}\} - \mathbf{I})\mathbf{V}\|_F^2 = \arg \min_{\mathbf{F}} \text{trace}\left(\mathbf{V}^T(P\{\mathbf{FQ}\} - \mathbf{I})^T(P\{\mathbf{FQ}\} - \mathbf{I})\mathbf{V}\right) \quad (4.17)$$

Now, we carry out some algebraic manipulation of the argument of (4.17). Using axiom 2 for moving \mathbf{V}^T to the other side and known from (4.6), $P\{\mathbf{X}\} = \mathbf{V}\mathbf{V}^T$, we rewrite

$$\|(P\{\mathbf{FQ}\} - \mathbf{I})\mathbf{V}\|_F^2 = \text{trace}\left((P\{\mathbf{FQ}\} - \mathbf{I})^T(P\{\mathbf{FQ}\} - \mathbf{I})P\{\mathbf{X}\}\right) \quad (4.18)$$

Manipulating the above expression using the properties of projector matrix (axioms 3 through 5), we obtain

$$\|(P\{\mathbf{FQ}\} - \mathbf{I})\mathbf{V}\|_F^2 = -\text{trace}(P\{\mathbf{FQ}\}P\{\mathbf{X}\}) + \text{trace}(P\{\mathbf{X}\}) \quad (4.19)$$

Using axiom 6, clearly in (4.19), $\text{trace}(P\{\mathbf{X}\})$ is a positive constant given the known matrix \mathbf{X} . Therefore, we minimise the derived expression by maximising $\text{trace}(P\{\mathbf{FQ}\}P\{\mathbf{X}\})$. Thus, we can write:

$$\arg \min_{\mathbf{F}} \|(P\{\mathbf{FQ}\} - \mathbf{I})\mathbf{V}\|_F^2 \equiv \arg \max_{\mathbf{F}} \text{trace}(P\{\mathbf{FQ}\}P\{\mathbf{X}\}) \quad (4.20)$$

From the definition of Vora-Value, we know $\nu(\mathbf{FQ}, \mathbf{V}) = \frac{1}{3}\text{trace}(P\{\mathbf{FQ}\}P\{\mathbf{X}\})$. We conclude:

$$\arg \min_{\mathbf{F}} \|(P\{\mathbf{FQ}\} - \mathbf{I})\mathbf{V}\|_F^2 \equiv \arg \max_{\mathbf{F}} \nu(\mathbf{FQ}, \mathbf{V}) \quad (4.21)$$

□

In summary, we have shown that a least-squares procedure that finds a filter that - in combination with a linear least-squares mapping (see (4.13) through (4.16)) - best fits an orthogonal basis of the colour matching functions (i.e. minimises the fitting error) must simultaneously maximise the Vora-Value.

Here we have proved the optimisation equivalence; however, the filter solutions found by the ALS and gradient algorithms can be different as we will show in the experimental results. The paths these two algorithms taking to a converged solution are different. The mathematical details of how the filter solution is updated in each iteration in both algorithms are given in Appendix A.2.2. We show both their similarity and difference that these two algorithms converge in a similar way but have different convergence rate.

4.4.3 Gradient and Hessian for the Regularised Optimisation

In (4.22), we reformulate the filter design optimisation that we aim to minimise (instead of to maximise as for the Vora-Value optimisation):

$$\mu(\mathbf{FQ}, \mathbf{X}) = -\text{trace}(\mathbf{P}\{\mathbf{FQ}\}\mathbf{P}\{\mathbf{X}\}) + \alpha \|\mathbf{F}\|_2^2. \quad (4.22)$$

In contrast to the Vora-Value optimisation, we reverse to the negative and cancel the fractional scalar in the equation. Here we use the symbol μ to denote the new objective function with a regularisation term to distinguish from ν for the Vora-Value. The penalty term that we introduce here is the squared norm of the filter where $\alpha > 0$.

Clearly, we have $\nabla\mu(\mathbf{f}) = -3\nabla\nu(\mathbf{f}) + 2\alpha\mathbf{f}$. The gradient of the regularised optimisation is calculated as

$$\nabla\mu(\mathbf{f}) = -2 \text{eddiag}\left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{P}\{\mathbf{X}\}(\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\})\right) + 2\alpha\mathbf{f}. \quad (4.23)$$

Similarly, we can derive the Hessian matrix for (4.22) and its explicit expansion can be obtained as

$$\begin{aligned} \nabla^2\mu = & -4\left(\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) \circ \left(\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\}\right)\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} + 2\left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) \circ \left(\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\}\right)\mathbf{P}\{\mathbf{V}\} \\ & + 2\left(\mathbf{I} - 2\mathbf{P}\{\mathbf{FQ}\}\right)\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \circ \left(\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) + 2\left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) \circ \mathbf{I} + 2\alpha\mathbf{I} \end{aligned} \quad (4.24)$$

where the last term relates to the regularisation term. Because the inverse of Hessian drives the Newton's method, it is essential the Hessian has full rank. We prove in Appendix A.2.4 that the Hessian is positive definite when a non-zero multiple, α , is added to the identity matrix and thus the Hessian matrix is invertible.

4.5 Results

4.5.1 Vora-filters

Actually, as we prove the equivalence of the modified Luther optimisation and Vora-Value optimisation, we have only one optimisation in hand. We name the derived optimal filters as **Vora-filters** (in contrast to the **Luther-filters** in Chapter 3).

The optimal **Vora-filters** for a Canon 40D and a Nikon D80 DSLR cameras are shown in Figures 4.3 and 4.4 respectively. In the figure, the maximum transmittance of the filters

are normalised to 100%. The spectral sensitivities of the testing cameras are plotted in Figures 4.3a and 4.4a for reference. When using the algorithms for solving for the filters, we use a regularisation value of $\alpha = 0.01$ in the Newton’s method and $\alpha = 0$ in the ALS and gradient algorithms. On the left, we show the filter solutions derived from three solvers. Specifically, the optimal filters obtained using the ALS (Algorithm 3.1) in green, the gradient algorithm (Algorithm 4.1) in blue and the Newton’s method (Algorithm 4.2) in red. We can see that these three algorithms find very similar (almost the same) filter solutions for both cameras. So much so that you can only see the difference of the transmittance values at the ends of the visible spectrum. Otherwise the graphs are on top of one another.

As readers will see in Figures 4.3 and 4.4, the optimised filters have sharp ‘spiked’ peaks in the short wavelength range (and also the long wavelength range for Nikon D80 camera). This result at first glance may seem to be surprising but is actually not unexpected. Both cameras, Figures 4.3a and 4.4a, have little sensitivity at the ends of the visible spectrum and as consequence to locally change the shape of the effective sensitivities (filter multiplied by camera) necessitates spikes of the kind shown. Surely it is with difficulty to manufacture of filters with spikes as shown (if they can be manufactured at all). The problem will be addressed in Chapter 6.

4.5.2 Comparison of Algorithm Performance

Although these algorithms find highly similar filter solutions, there are significant differences in how they converge. In Figure 4.5, we show how many iterations it takes these three algorithms to converge (here we show the results of the Canon 40D camera). We evaluate their performance at each iteration in terms of two different error metrics: Vora-Value in Figure 4.5a and mean ΔE_{ab}^* in Figure 4.5b.

We use the same data set and calculation process as described in Section 2.4 for calculating the colour error metric ΔE_{ab}^* . The mean ΔE_{ab}^* used is calculated by averaging the colour error results over the reflectance set under all 102 illuminants.

The performance of the ALS, gradient, Newton’s method algorithms are respectively shown in green, blue and red lines in Figure 4.5. We can see that the Newton’s method (red) converges slower in the first few iterations but quickly speeds up to arriving at the optimal solution. It takes less 10 iterations for a well-converged solution. Comparatively, for the same Vora-Value target, the numbers of iteration that ALS (green) and gradient (blue) algorithms need are, respectively, in the magnitude of 10^2 and 10^3 iterations. These two algorithms converge much slower, especially when approaching the optimal. In terms of running time for finding the optimal filter for a given camera, it takes about

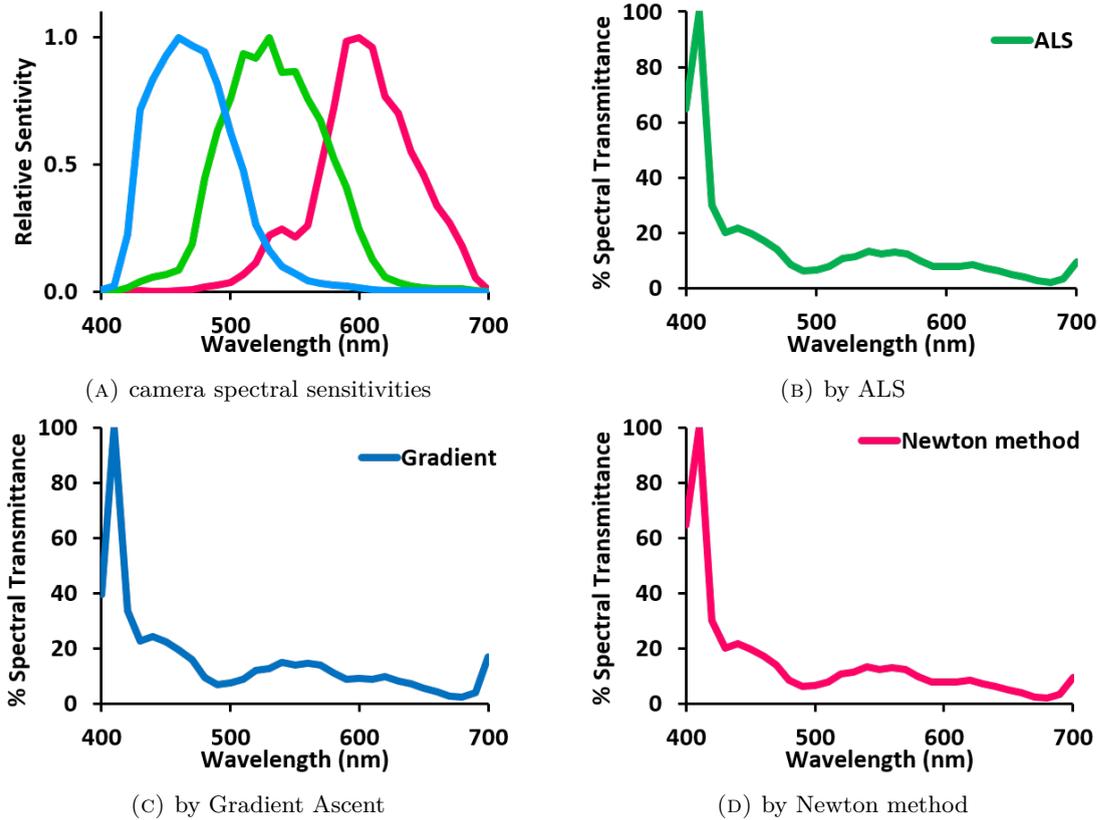


FIGURE 4.3: (A) Spectral sensitivities of Canon 40D camera and the optimised filters solved from the algorithms: (B) Alternating Least-Squares (in green), (C) Gradient Ascent (in blue) and (D) Newton’s method (in red).

0.1 second for the ALS algorithm and 0.85 second for the gradient algorithm compared to 0.009 second for Newton’s method in Matlab R2018b on a desktop PC (Intel Core i5 64-bit operating system). We would like our readers recall that Newton’s method generally costs much more computationally, particularly for calculating the second derivative and its inverse. In contrast, the ALS method has a much simpler form which uses least-squares regression solved simply by the closed-form Moore-Penrose inverse [31].

4.5.3 Vora-Values and Colour Measurement Experiment

Now we evaluate the performance of our optimised filters based on Vora-Value optimisation and compare to the prior method of Luther-condition based optimisation in Chapter 3. For convenience, the two filter design methods are noted as **Vora-filter** and **Luther-filter** hereafter, respectively. To be crystal clear, the experimental results we show for the Luther method is from the algorithm developed in Chapter 3 where we find the best filter making a camera best match the XYZ CMFs (under a linear transform). That is, the results do not report the modified Luther optimisation (which, of course, returns the same result as the Vora-Value modification).

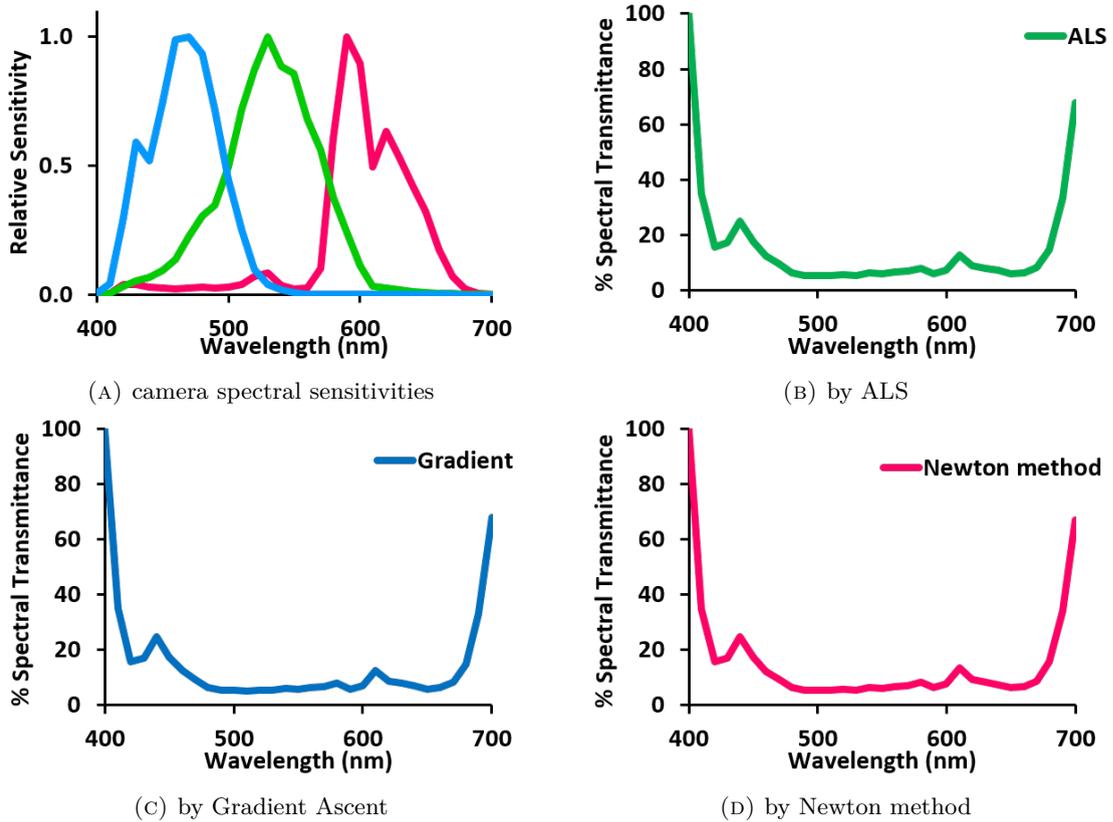
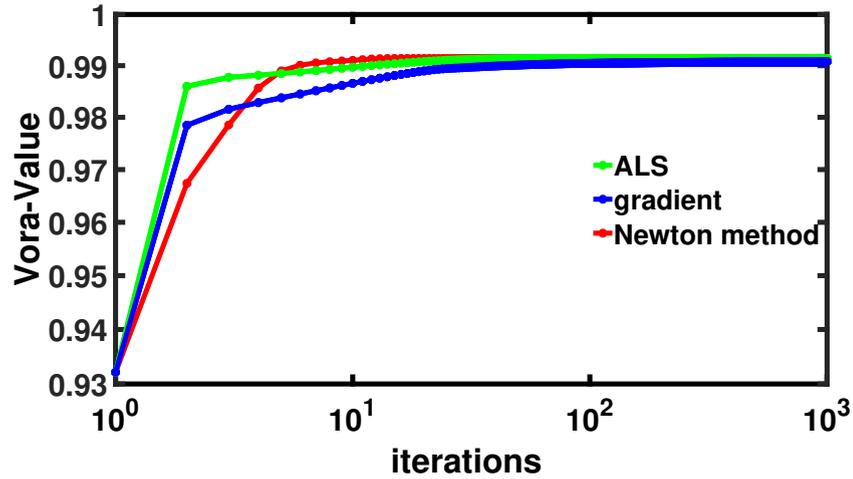


FIGURE 4.4: (A) Spectral sensitivities of Nikon D80 camera, the optimised filters solved from the algorithms: (B) Alternating Least-Squares (in green), (C) Gradient Ascent (in blue) and (D) Newton’s method (in red).

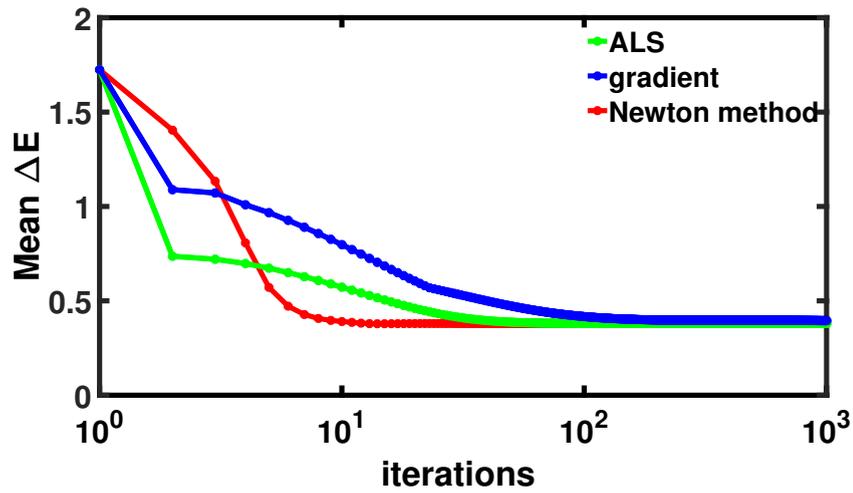
TABLE 4.1: ΔE_{ab}^* statistics of the colour corrected **NAT**ive camera without filter, the colour corrected camera with the optimised **LUTH**er-filter and **VORA**-filter for Canon 40D and Nikon D80 cameras.

Method	Canon 40D						Nikon 80D					
	Vora-Value	mean	median	95%	99%	max	Vora-Value	mean	median	95%	99%	max
NAT	0.932	1.72	1.03	5.12	12.94	28.39	0.888	1.95	1.14	6.30	13.14	38.43
LUTH	0.986	0.44	0.22	1.48	3.19	8.77	0.945	1.39	0.87	4.49	7.35	11.43
VORA	0.991	0.38	0.20	1.23	2.97	9.89	0.967	0.81	0.50	2.72	4.66	8.40

Note that the results of **Vora-filter** reported here relate to the filter solved by using the ALS algorithm as it has simpler forms (than the gradient ascent or Newton algorithm for the Vora-Value formulation) for implementation and obtains a filter solution with almost no difference to the other two algorithms. We also include the results of the native camera without a filter (denoted **NAT**ive) to establish the baseline results. The Vora-Value results are presented for both cameras Table 4.1. It is clear that with filters optimising for the Luther condition, both deliver a higher Vora-Value (unsurprisingly, but a good test that our new Vora-Value method works).



(A) maximising the Vora-Value metric



(B) minimising mean colour error

FIGURE 4.5: Algorithm convergence in terms of (A) Vora-Value and (B) mean ΔE_{ab}^* .

From Table 4.1, we can see that the current **Vora-filter** for Canon 40D improves the Vora-Value to 0.991 from 0.932 for its unfiltered **NATive** sensor sensitivities and higher than 0.986 by the prior art of the Luther-filter. Similar gain can also be seen for Nikon 80D camera on the right part of the table. Generally, a higher Vora-Value indicates greater similarity of the vector spaces spanned by the sensitivities of a camera and human visual system and therefore relates to substantially higher accuracy in colour measurement [88].

Now we examine the performance of our derived filters in the colour measurement experiment. We use the same data set and calculation process as described in Section 2.4. The colour error statistics are shown in columns 3-7 for Canon and columns 9-13 for Nikon in Table 4.1. We can see that the filters from the current optimisation performs the best (when no constraint is added). Compared to the baseline results, for both testing cameras, we can conclude that by using such optimised filters, we can effectively reduce

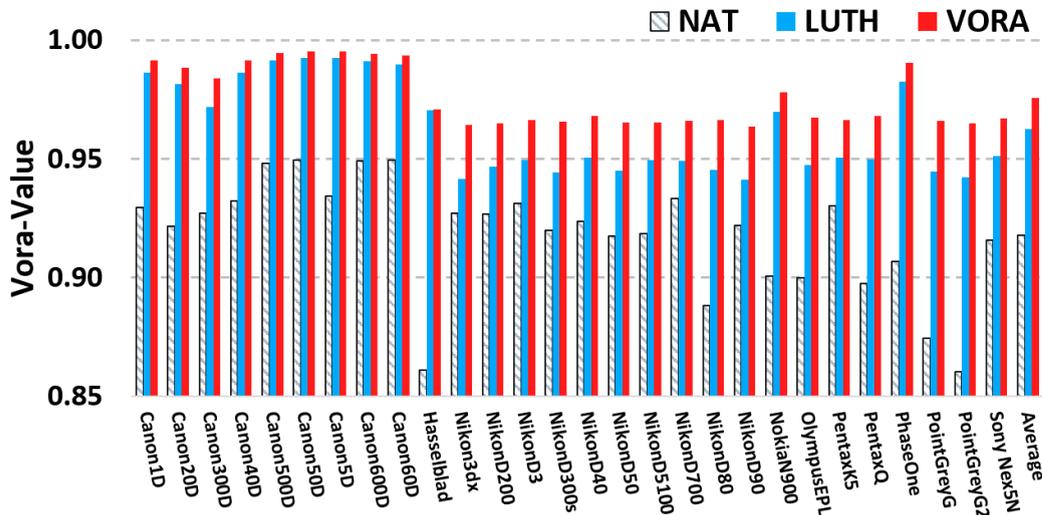
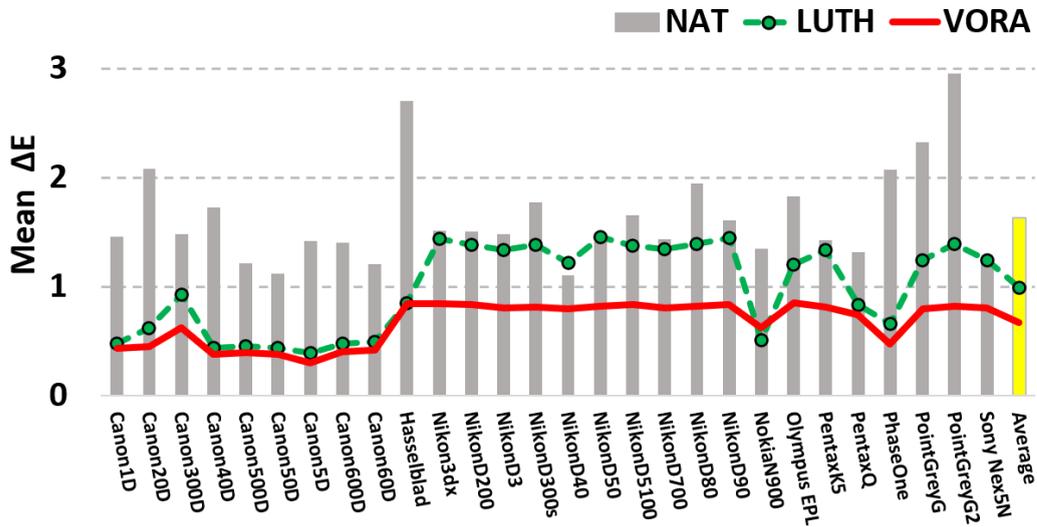


FIGURE 4.6: Results of Vora-Values for a group of 28 cameras: **NAT**ive sensitivities without a filter (in neutral), the **LUTH**er-filters (in blue) and the **VORA**-filters (in red). The overall averaged Vora-Values of the camera group are given in the last column.

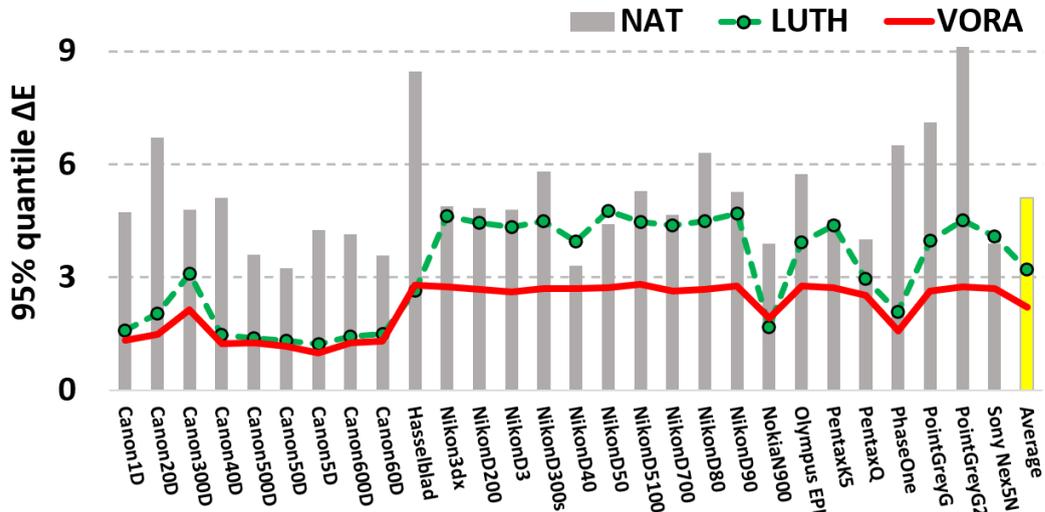
the colour errors by two thirds to three quarters. Compared to the **Luther-filters**, we see that a marginal improvement in Vora-Value score leads to a notable increase in colour performance: a further 10% reduction.

Now, we extend the experiments to a much broader group of 28 digital cameras with known spectral sensitivity data set. For each camera, we solve for the optimised filters given the corresponding spectral sensitivities. The Vora-Value performance for 28 cameras - with and without their optimised filters - are shown in Figure 4.6. The Vora-Value for the unfiltered **NAT**ive camera sensitivities are shown in chevroned grey bars and for the **Luther-filters** in blue and the new **Vora-filters** in red. It can be seen that the current optimised filters have the highest Vora-Value scores. The overall results, averaged across the whole camera group, are depicted in the rightmost bars. We see that, on average, the native Vora-Value is increased from 0.918 to 0.963 (**Luther-filters**) and 0.976 (**Vora-filters**) when the optimised filters are placed. Unsurprisingly, the current filter method **Vora-filters** outperforms the prior art **Luther-filters** for most cameras.

Now we calculate the mean ΔE_{ab}^* error (using the same illuminant and reflectance data sets) for 28 cameras. For each camera, we calculate its colour performance of how well it can predict the ground-truth values using the linear correction - both with and without a filter - in terms of ΔE_{ab}^* error metrics. We calculate the optimal **Vora-filters** and **Luther-filters** for each individual camera. Figure 4.7 summarises the mean and 95-percentile ΔE_{ab}^* performance for each camera in the data set. Grey bars show the mean and 95 percentile error performance of unfiltered colour corrected RGBs for the



(A) mean ΔE_{ab}^* colour error



(B) 95-percentile ΔE_{ab}^* colour error

FIGURE 4.7: (A) Mean and (B) 95-percentile ΔE_{ab}^* errors for 28 cameras. The grey-bars show the colour errors for **NATIVE** cameras simply use colour correction without a filter. The results by using the **LUTHER**-filters and **VORA**-filters are in dashed green lines and solid red lines respectively.

28 cameras. The dashed green and solid red lines record the performance of the best **Luther-filters** and **Vora-filters**, respectively.

Clearly, by using optimised filters we have greatly improved colour measurement for all 28 cameras and on average the performance increment is significant. Our current optimised **Vora-filters** deliver significantly better colour measurement performance compared with using the prior art of the Luther-condition optimised filters.

4.6 Conclusion

In this chapter, we have presented a unifying filter design method based on the underlying relation between the Luther condition and the Vora-Value. We show the optimal filter targeting at these two objectives can be simultaneously achieved. Hence, we can find an optimised Vora-filter that in the least-squares sense, also optimises the Luther condition. We have developed three algorithms to solve for the best filter, i.e. alternating least-squares(ALS), gradient ascent and Newton's method. We also find that the ALS approach has competitive performance compared to Newton's method and has a much simpler formulation. Experiments validate our method.

Both filter design methods presented in the last two chapters are data-independent with no a priori knowledge of the measuring spectra data. In the next chapter, we will present a data-dependent filter design method driven by the real-world measured spectra.

Chapter 5

Data-Driven Filter Design

In this chapter, we reformulate our filter design optimisation for making the camera responses as close to the CIE XYZ tristimulus values as possible given the knowledge of real measured surfaces and illuminants data. The problem is formulated as a bilinear least-squares estimation problem (linear both in the filter and the colour correction) and can be solved using an Alternating Least-Squares (ALS) regression technique (in which the two unknown matrices are iteratively solved).

5.1 Introduction

Mapping the device RGBs measured by a camera to either display coordinates (such as sRGB) or the XYZ tristimuli - a human vision system referenced colour space - is called colour correction. Colour correction is an essential procedure in the camera pipeline since cameras do not ‘see’ the world as humans do. Fundamentally, this is because the relationship between the spectral sensitivities of a camera and human visual matching functions is not a linear mapping. Explicitly, the Luther condition is not satisfied [52].

Many different algorithms for solving the colour correction problem have been developed in the literature. The most common method is to apply a linear correction transform mapping RGBs to XYZs (or display RGBs). While linear correction generally works well, it can still fail in a large number of cases, especially for saturated colours. In order to address this issue, polynomial regression methods were introduced by additionally considering the correlation and auto-correlation between sensor channels [35]. Polynomial and local mapping methods can target more complex colour problem or optimisation aspects, e.g. to correct spatial [48] and intensity non-uniformity [26], to preserve hue attribute [3, 55], to map one or two essential surfaces correctly [28], or to achieve better perceptual performance [85]. However, Finlayson *et al.* [29] pointed out that most

polynomial methods present the exposure issue, i.e. results can be affected by exposure changes. Indeed, given two input RGBs, $\boldsymbol{\rho}$ and $k\boldsymbol{\rho}$, the outputs of colour correction are \mathbf{q} and $k\mathbf{q}$ if it is exposure invariant. For this reason, they proposed the root-polynomial regression method to retain the exposure-invariant feature and their proposal outperformed most polynomial-type methods. Apart from regression approaches, methods like look-up-tables [37] and recently artificial neural networks [13, 104] have also been proposed for colour correction.

Inspired by the idea of using a colour filter for improving colorimetric ability [25], in Chapter 3, we propose a filter design method for a given camera system in the colour matching sense such that the whole system is almost colorimetric. The results of Luther condition based filter design provide impressive spectral match, yet did not show any striking improvement in colour correction for some Nikon cameras (as seen in Figure 3.5). We hypothesise the reason is that the optimisations presented so far in the thesis do not take account of the spectral characteristics of actual reflectance and illuminant spectra.

In this chapter, we follow the philosophy of altering the camera sensitivities using a prefilter, but to exploit the colorimetric capability for a given camera system in a colour correction way. That is, rather than spectral sensitivity matching, this method builds upon the system outputs by considering the colour formation process. A camera still holds the colorimetric characteristic if its outputs regarding to the scene under certain lighting condition are equal to the human visual responses, even somehow the camera sensitivity flouts the Luther condition.

Suppose $\underline{Q}(\lambda)$ denotes the spectral sensitivity functions of a camera system and $\underline{X}(\lambda)$ denotes the observer colour matching curves respectively. Physically, the role of a filter, which absorbs light on a per wavelength basis, is multiplicative. If $f(\lambda)$ is a transmittance filter, then $f(\lambda)\underline{Q}(\lambda)$ is a physically accurate model of the effect of placing a filter in front of the camera sensor. Then an explicit colorimetric error by applying a filter in a least-squares meaning can be modelled as:

$$\min_{f(\lambda), \mathbf{M}} \left\| \left[\int_{\Omega} E(\lambda)S(\lambda)f(\lambda)\underline{Q}(\lambda)d\lambda \right] \mathbf{M} - \int_{\Omega} E(\lambda)S(\lambda)\underline{X}(\lambda)d\lambda \right\|_F^2 \quad (5.1)$$

The spectral power distribution of a light source coming into the object is represented by $E(\lambda)$ while the object reflectance as $S(\lambda)$. The subscript $\| \cdot \|_F$ denotes the Frobenius norm. The two integration operations over the visible range Ω (from 400 nm to 700 nm) in the above equation refer to the system responses of a camera and a human observer respectively. Detailed explanation of colour formation can be found in Section 2.2. In the above optimisation equation, $f(\lambda)$ defines the physical feature of the designed filter, i.e. the spectral transmittance, and \mathbf{M} is a 3×3 linear transform matrix.

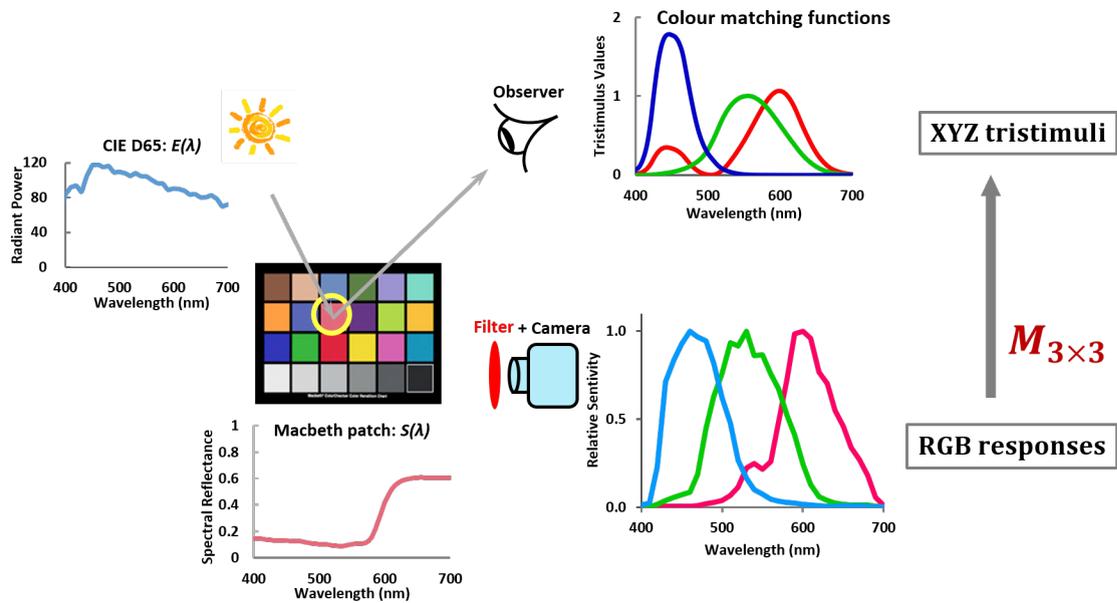


FIGURE 5.1: Schematic diagram of colour measurement for an object viewed under an illuminant given their spectral representations. Given the spectral data of the measured objects, we seek a filter for a camera such that the RGB outputs after a linear mapping become the same as the XYZ tristimulus values.

Figure 5.1 visualises the formulation problem of a camera system with a prefilter (shown as a red ellipse for illustration) and an observer under a given viewing condition over a standard colour chart (the two sensing systems should be placed under the same viewing geometry in reality). Our aim is to seek a filter for a corresponding camera, so that the device output RGB values after a linear correction \mathbf{M} predict the closest triplets as possible as a CIE standard observer.

In this chapter, we formulate a filter design driven by the actual measured spectra data that best approximates a colorimetric system, and develop an algorithm to solve this optimal filter and mapping matrix. In order to quantify the colorimetric performance, we calculate the CIELAB colour difference – ΔE_{ab}^* – between the new system outputs and reference XYZ values [103]. A comparison of linear colour correction, Luther-filter method and the current method will be made. You will see a dramatic improvement in terms of perceptual colour error results using this new method.

Experiments demonstrate that we can find **Data-driven** filters that can dramatically increase the colorimetric accuracy across a large set of commercial cameras.

5.2 The Data-driven Luther Condition

The Luther condition for spectral sensitivities presupposes that any and all physical spectra are plausible. However, we know that reflectance spectra are smooth [14, 46, 93].

Lights while more arbitrary are designed to integrate to fall near or close to the Planckian locus [65] and to score well on measures such as the Colour Rendering Index [10].

Let \mathbf{Q} and \mathbf{X} denote the sensitivity functions of the camera and XYZ responses in discrete manner, the columns of which are the spectral sensitivity of each sensor channel. Also, let \mathbf{C} denote a collection of n colour spectra in the $n \times 31$ colour signal matrix. Pragmatically, a camera is colorimetric if its responses are a linear transformation from the XYZ tristimulus values

$$\mathbf{C}^T \mathbf{X} = \mathbf{C}^T \mathbf{Q} \mathbf{M} \quad (5.2)$$

where \mathbf{M} is a 3×3 matrix. Equation (5.2) is a Data-driven formulation of the Luther condition.

5.3 Data-driven Filter Optimisation

Simple Case: in the simple Data-driven approach, we look for a colour filter and the 3×3 colour correction matrix that, in a least-squares sense, best maps camera measurements for a training colour signal data set to the corresponding ground-truth XYZ tristimulus values. Denoting an ensemble of n colour signals in the matrix \mathbf{C} (in the size of $n \times 31$), the Data-driven optimisation in (5.1) is rewritten in the discrete representation as:

$$\min_{\mathbf{f}, \mathbf{M}} \|\mathbf{C}^T \text{diag}(\mathbf{f}) \mathbf{Q} \mathbf{M} - \mathbf{C}^T \mathbf{X}\|_F^2 \quad \text{s.t. } \mathbf{f} > 0 \quad (5.3)$$

Solving (5.3) depends on the structure of the colour signal matrix. If we choose $\mathbf{C} = \mathcal{I}_{31}$ (the 31×31 identity matrix), it simplifies into the Luther-condition optimisation. This assumption is related to the Maximum Ignorance assumption [72] where all possible spectra are considered equally likely.

General Case: Let us develop a more general optimisation statement. One, where we have cnt colour signal matrices - denoted \mathbf{C}_j ($j \in \{1, 2, \dots, cnt\}$) and the corresponding cnt colour correction matrices \mathbf{M}_j . Each colour signal matrix typically corresponds to a training set of surface reflectances illuminated by a single spectrum of light \mathbf{E}_j , thus the colour signal matrix is $\mathbf{C}_j = \text{diag}(\mathbf{E}_j) \mathbf{S}$, where \mathbf{S} is a $31 \times n$ matrix of reflectances, one reflectance spectrum per column. But, the different light assumption is not a necessary assumption. As an example, we might mix colour signals for the Maximum Ignorance assumption with measured data (i.e. two colour signal matrices) where both measurement sets contain multiple lights.

The general Data-driven filter optimisation problem is written as:

$$\min_{\mathbf{f}, \mathbf{M}_j} \sum_{j=1}^{cnt} \|\mathbf{C}_j^T \text{diag}(\mathbf{f}) \mathbf{Q} \mathbf{M}_j - \mathbf{C}_k^T \mathbf{X}\|_F^2 \quad \text{s.t. } \mathbf{f} > 0 \quad (5.4)$$

Finally, k could denote some other privileged standard reference viewing condition (where the reference viewing illuminant is not in the set of training lights). For example, in colour measurement, we are often interested in the XYZ tristimuli for a daylight illuminant D65 which has a prescribed but not easily physically replicable spectrum.

In most cases, $k = j$; that is, per illuminant, we seek the filter such that real camera responses can be mapped linearly to corresponding XYZ tristimuli (for that light).

5.3.1 ALS Algorithm for Solving the Optimisation

We are going to solve (5.4) for the filter \mathbf{f} and the correction matrices using an alternating least-squares procedure (although we make some necessary adjustments) as shown in Algorithm 5.1.

Again to ease notation, we use $\mathbf{F} = \text{diag}(\mathbf{f})$ and write $\mathbf{F}\mathbf{Q}$. When we wish to extract \mathbf{f} from \mathbf{F} , we write $\mathbf{f} = \text{ediag}(\mathbf{F})$ ('e' signifies to 'extract' the diagonal). Notice that the input to the optimisation is an initial filter guess denoted by \mathbf{f}^{seed} . The seeding filter can be a uniform filter with $[1, 1, \dots, 1]$, the Luther-filter $\mathbf{f}^{\text{Luther}}$, the Vora-filter \mathbf{f}^{Vora} , or any random filter. We will investigate the initialisation effect later in the Results section.

Algorithm 5.1 ALS algorithm for the Data-driven optimisation

- 1: $i = 0, \mathbf{F}^0 = \text{diag}(\mathbf{f}^{\text{seed}}), \mathbf{Q}^0 = \mathbf{F}^0 \mathbf{Q}$
 - 2: **repeat**
 - 3: $i = i + 1$
 - 4: $\min_{\mathbf{M}_j^i} \|\mathbf{C}_j^T \mathbf{Q}_j^{i-1} \mathbf{M}_j^i - \mathbf{C}_k^T \mathbf{X}\|_F^2, j = 1, 2, \dots, cnt$
 - 5: $\min_{\mathbf{F}^i} \sum_{j=1}^{cnt} \|\mathbf{C}_j^T \mathbf{F}^i \mathbf{Q}_j^{i-1} \mathbf{M}_j^i - \mathbf{C}_k^T \mathbf{X}\|_F^2, \mathbf{f}^i > \mathbf{0}$
 - 6: $\mathbf{Q}_j^i = \mathbf{C}_j^T \mathbf{F}^i \mathbf{Q}_j^{i-1} \mathbf{M}_j^i, j = 1, 2, \dots, cnt$
 - 7: **until** $\forall(j) \|\mathbf{Q}_j^i - \mathbf{Q}_j^{i-1}\|_F^2 < \epsilon$
 - 8: $\mathbf{f}^{\text{Data}} = \text{ediag}(\prod_{s=0}^i \mathbf{F}^s)$ and $\mathbf{M}_j = \prod_{s=1}^i \mathbf{M}_j^s, j = 1, 2, \dots, cnt$
-

Let us consider three candidate minimisation conditions corresponding to three common scenarios for colour measurement each of which can be solved using Algorithm 5.1.

1) *Multiple Lights*: Here we assume that j indexes over cnt illuminants and $k = j$ (per illuminant we make the target XYZs using the same colour signals). We find a single

filter which given per-illuminant optimal least-square 3×3 correction matrices will best fit camera data to the corresponding multi-illuminant XYZs.

2) *Multiple measurement lights, single target light*: Again j indexes over cnt illuminants. But, the target is a single illuminant, for example CIE D65 [65].

3) *Single Light*. This case is, in effect, the simple restriction of the general case, $cnt = 1$. We have one measurement light and one target light. Like the Luther-condition optimisation, we solve for a single correction matrix.

In Algorithm 5.1, it is straightforward to solve for the j th colour correction matrix at i th iteration, \mathbf{M}_j^i , in Step 4 using the Moore-Penrose inverse. In Step 5, we find the filter with positive values. It is important to note that from a physical perspective, the transmittance of the filter must be within the range $[0, 100\%]$. Here, to ensure that the filter is all positive, we can solve for the filter by solving the optimisation subject to the positivity constraint, we solve a quadratic programming problem [31] (unlike the Luther-condition case there is no a prior mathematical argument as to why the best filter should be all positive). Quadratic programming allows linear least-squares problem subject to linear constraints to be solved rapidly and, crucially, a global optimum is found. Details of the least-squares computation and its conversion into the quadratic problem formulation are given in the Appendix A.3.

5.4 Results

5.4.1 Data-filters

Simple Case: We show the **Data-filters** optimised for a single illuminant - respectively, a CIE D65 (bluish) or a CIE A (yellowish) illuminant - with a collection of 1995 surfaces for a Canon 40D camera in Figures 5.2a and 5.2b. These filters are obtained by using the optimised **Luther-filters** obtained from Chapter 3 to seed the Algorithm 5.1, $\mathbf{f}^{feed} = \mathbf{f}^{Luther}$).

General Case: We also show the optimised **Data-filters** optimised for general case in Figure 5.2c: under 102 illuminants and 1995 object reflectances (again we use the **Luther-filters** to seed the Algorithm 5.1, $\mathbf{f}^{seed} = \mathbf{f}^{Luther}$).

Note that in order to simulate a physically reliable filter, we constrain its parameters in the range of $[0, 100\%]$ and for the current method, the experimental results presented here (in Table 5.1) are based on this constraint.

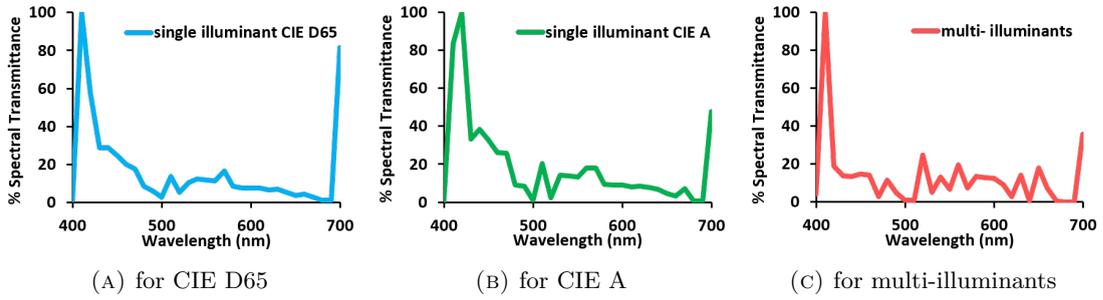


FIGURE 5.2: Spectral transmittance of the optimised Data-drive filters for single illuminant case (A) CIE D65 and (B) CIE A, and (C) multi-illuminant case of 102 illuminants for Canon 40D camera.

5.4.2 Colour Measurement Experiments

Now let us evaluate the derived **Data-filters** in terms of a perceptually relevant colour measurement reproduction metric.

TABLE 5.1: ΔE_{ab}^* statistics of the colour corrected **NAT**ive camera, the colour corrected camera with the **LUTH**er-condition optimised filter, and the colour corrected camera with the **DATA**-driven optimised filter for Canon 40D camera under different lighting cases

	Mean	Median	90%	95%	99%	Max
CIE D65						
NAT	1.65	1.03	3.55	4.94	11.23	19.29
LUTH	0.46	0.25	1.09	1.45	3.49	5.90
DATA	0.38	0.20	0.93	1.25	2.45	4.62
CIE A						
NAT	2.30	1.44	4.65	6.17	16.96	26.41
LUTH	0.64	0.40	1.33	1.84	4.75	8.19
DATA	0.43	0.25	0.99	1.44	2.63	3.84
102 illuminants						
NAT	1.72	1.02	3.68	5.12	12.94	28.39
LUTH	0.53	0.30	1.15	1.65	4.11	6.83
DATA	0.41	0.21	0.96	1.32	2.78	6.78

5.4.2.1 Single Light Case

Let us again use the Canon 40D camera spectral sensitivities as a putative measurement device and quantify how well it can measure colours - with and without a filter. In this experiment the measurement light is either a CIE D65 (bluish) or a CIE A (yellowish) illuminant. For reflectances we use the SFU set of 1995 spectra [6] (itself a composite of many widely used sets). The 1995 XYZs for these reflectance and lights are the *ground-truth* with respect to which we measure colour measurement error.

Using the Canon camera sensitivities, the reflectance spectra, and either the CIE D65 and A lights, we numerically calculate two sets of 1995 RGBs. Now, we linearly regress the RGBs for each light to their corresponding ground-truths (we map the native RGBs for CIE D65 and A to respectively the XYZs under the same lights). We call these colour corrected RGBs the **NAT**ive camera predictions (and we adopt this notation in the results shown in Table 5.1). Rows 1 and 4 of Table 5.1 report the mean, median, 90, 95 and 99 percentile and the maximum CIELAB ΔE error for CIE D65 and CIE A lights.

Now let us place a filter in front of the camera. Again, we calculate two sets of RGBs (one for each light) for the camera spectral sensitivities multiplied by the filter found a **Data-driven** colour filter. The recorded filtered RGBs are mapped best to corresponding XYZs using linear regression. Results for the corrected **DATA**-driven filtered RGBs for the two lights are reported in rows 3 and 6 of Table 5.1. For comparison, we also show the results of optimised **Luther-filters**. The **LUTHER** ΔE_{ab}^* error statistics are shown in rows 2 and 5.

Clearly, the ‘camera+filter’ can measure colours more accurately compared to the case where a filter is not used. Across all metrics the ΔE_{ab}^* errors reported are about a quarter of those found when a filter is not used.

Significantly, the best **Data-driven** filter also delivers improved performance compared with the results reported for the Luther-condition optimised filter. The errors are further reduced by about a quarter. Incorporating knowledge of typical lights and surfaces into the optimisation leads to improved colour measurement performance.

5.4.2.2 Multiple Lights Case

We now repeat the experiment for a set of 102 measured lights [6] using the optimal filter shown in Figure 5.2c. The results of this second experiment are shown in the last 3 rows of Table 5.1. Here, the reported error statistics are averages. For each illuminant - as

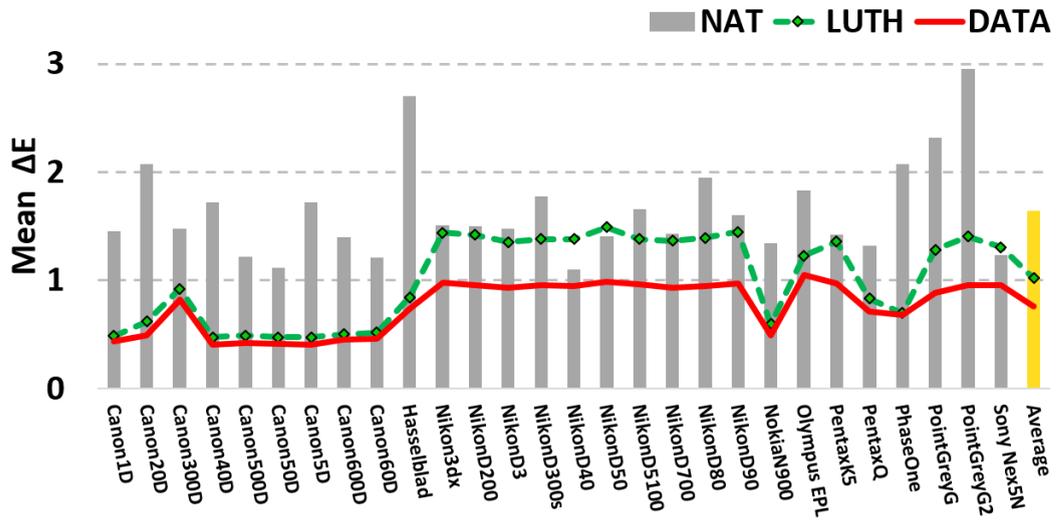
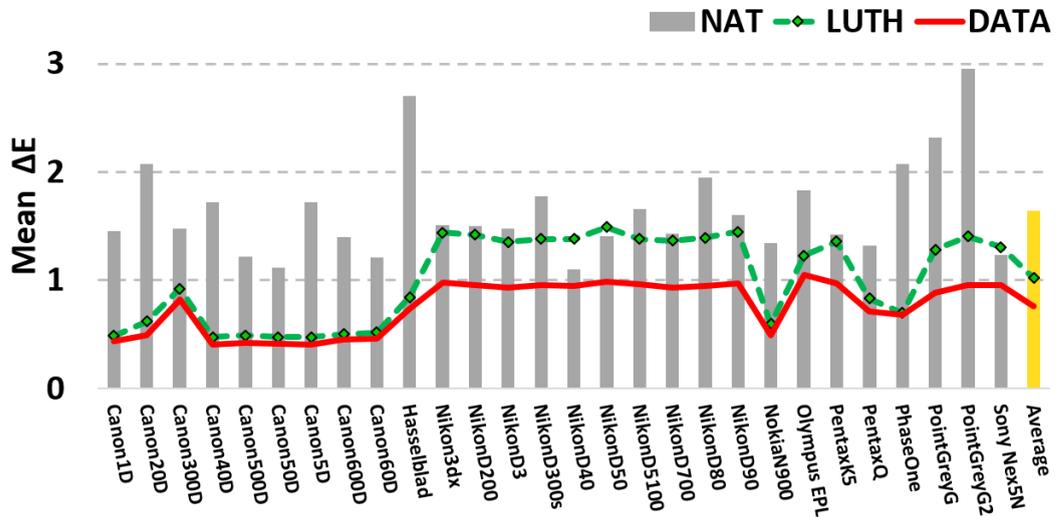
(A) mean ΔE_{ab}^* colour error(B) 95-percentile ΔE_{ab}^* colour error

FIGURE 5.3: Mean (A) and 95-percentile (B) ΔE_{ab}^* errors for 28 cameras. The grey-bars show the colour errors for **NAT**ive cameras simply use colour correction without a filter. The dashed green lines with black diamonds show the results by using the **LUTH**er-condition optimised filters. The results of the **DATA**-driven optimisation are plotted in solid red lines.

described in the single light case above - we calculate the mean, median, 90 percentile, 95 percentile, 99 percentile and maximum ΔE_{ab}^* . We then take the mean of each error statistic over all the lights. The aggregate illuminant set performance is reported in rows 7, 8 and 9 of Table 5.1 for respectively native RGBs and RGBs measured with respect to Luther-condition and Data-driven optimised filters.

In terms of the reported errors of the native RGBs compared to the filtered RGBs for the Luther- and Data-driven filters, we see the same data trend for the multiple lights case

as we saw previously for single lights. The Data-driven filter reduces the measurement error by 3/4, on average compared the native camera results.

5.4.2.3 Multiple Cameras

Now, we calculate the mean ΔE_{ab}^* error (for the 102 lights and 1995 reflectances) for each of 28 cameras [41]. For each camera we calculate the optimal Luther-condition and Data-driven optimal filters (where as before the Luther-condition filter seeds the Data-driven optimisation). Per camera, Figure 5.3 summarises the per camera mean and 95 percentile ΔE_{ab}^* performance.

Grey bars in Figures 5.3a and 5.3b show respectively the mean and 95-percentile error performance of native (un-filtered) colour corrected RGBs for the 28 cameras. Respectively, the dashed green and solid red lines record the performance of the best **Luther-filters** and **Data-driven filters**.

It is evident that the optimised filters support improved colour measurement for all 28 cameras and on average the performance increment is significant. For many cameras the Data-driven optimised filter delivers significantly better colour measurement performance compared with using the Luther-condition optimised filter.

5.5 Conclusion

In this chapter, we developed a data-driven method that designs transmittance filters which, when placed in front of a camera, make the camera more colorimetric. The method tackles colour correction for a given set of measured lights and surfaces, which we call **Data-driven** filter optimisation. Experiment results demonstrate that our Data-driven optimised filters to the testing cameras provides further improvement compared to the previous data-independent optimisations (i.e. the Luther-condition based optimization method).

Our default optimisation - though compellingly simple to formulate - deliver filters which are not smooth (difficult to make) and may also transmit very little light. In the next chapter, we will reformulate our filter optimisations to incorporate both smoothness and a lower bound on how much light must be transmitted so as to improve the feasibility of our filters.

Chapter 6

Filter Constraints

In the last three chapters, we proposed three different filter design methods — named **Luther-filter**, **Vora-filter**, and **Data-filter** respectively — for determining the optimal spectral transmittance of a prefilter that makes the camera more colorimetric. However, previous optimisation methods make no constraint on how should we shape the filters across the spectrum (besides the positiveness for the physical feasibility).

In this chapter, we extend the filter optimisations by providing a method for incorporating smoothness and transmittance bounds into the same optimisation framework. While finding the smooth filters, we also develop a multi-initialisation optimisation where we use multiple initial filters to seed the algorithm. We will show that the solved filters which are smooth and reasonably highly transmissive - and so plausibly manufacturable - provide similar performance to the prior art of unconstrained minimisation.

6.1 Introduction

When we look at the optimised filters, **Luther-filter** in Figure 3.2c, **Vora-filter** Figures 4.3 and 4.4, and **Data-filter** Figure 5.2, clearly, the filters obtained in the previous chapters are not desirable. For instance, the derived Luther-filter for Canon 40D shown in Figure 3.2c, has a sharp change in transmittance in the short wavelengths, and as a whole the filter is not smooth. Moreover, for most of the wavelength range, the filter transmits little light (<20%). By default, the filters found by using the algorithms proposed in the previous chapters can be arbitrarily non-smooth (even with the regularisation term introduced in Section 4.4.1 for the modified-Luther filter optimisation) and it might also be very non-transmissive. Non-smoothness can limit the feasibility of filter fabrication.

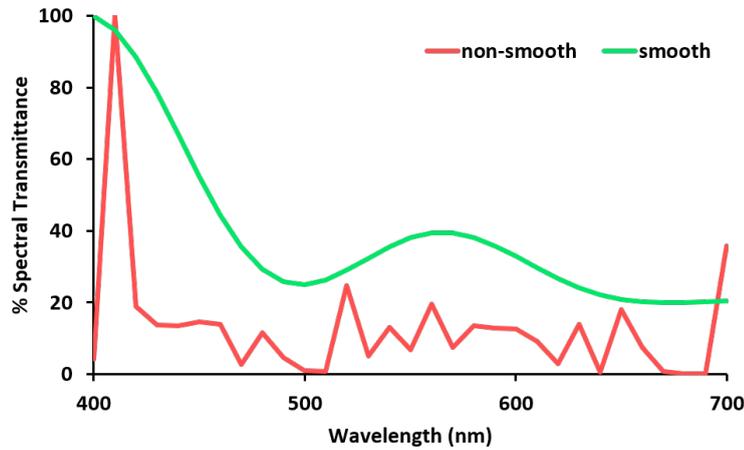


FIGURE 6.1: An example of smooth filters constructed linearly by using 6-cosine basis and also limited to transmit at least 20% of light at any wavelength, compared to the optimised Data-filter without smoothness constraint.

In this chapter, we restrain the spectral transmittance shape of the colour filter regards to its physical property. First, we extend the filter design to find a filter which is shaped in a smoothing manner with plausible transmissivity. We take inspiration from the work of Vora *et al.* [90] who emphasised the importance of the smoothness property of the filter transmittance curves for the fabrication process. They proposed to construct a smooth filter set by using linear combination of a set of smooth basis filters (e.g. Gaussian functions). In our work, smoothness will be enforced by specifying filters as a linear combination of the first few terms of the cosine basis expansion functions.

As well as being smooth, we may also wish a colour filter to transmit at least a lower bound percentage of the light that passes through. Indeed, a filter with too low transmittance values would perform, have limited practical utility. When a dark filter (that absorbs most of the light) is placed in front of a camera, we need to either increase the exposure time (or widen the aperture) or choose a higher ISO to obtain an image with the same light level as when no filter is applied. So, we will bound on the transmittance values with enforced maximum and minimum values to increase the overall amount of light that passes through the filters and enters into the camera. As a preview, in Figure 6.1, we present a smooth filter (in green) solved from the Data-driven optimisation after adding the constraints.

We reformulate our optimisation framework to allow us to incorporate both the smoothness and the minimum and maximum transmittance bounds of the recovered filter which are key practical concerns. We will show that the reformulation can be seen as a linearly constrained quadratic optimisation problem [49].

All optimisation methods presented in this thesis and the extension on smooth and bounded transmissive filters in this chapter are search based. While our empirical results are encouraging (the filters we design make cameras much more colorimetric), we cannot *prove* that the discovered filters are optimal. Indeed, they are not. If we initialise the Alternating Least-Squares procedure (e.g. see Algorithms 3.1 and 5.1) with a different starting point, we arrive at a different optimal filter. One of the advantages of our approach to bounding the smoothness of the colorimetric filters is that we can enumerate a coarse set of all smooth filters. And, with respect to this set, we can find the overall optimal filter.

Experiments demonstrate that we can find physically realisable filters which are smooth and reasonably transmissive which can greatly improve the colorimetric performance of a camera and thus we can use the ‘camera+filter’ setting for applications where high accuracy of colour measurement is needed.

6.2 Adding Filter Constraints

6.2.1 Smoothness Constraint

We would like to control the shape of the transmittance filter. Let us assume that a filter is a linear combination of a set of m basis filters:

$$\mathbf{f} = c_1 \mathbf{b}_1 + c_2 \mathbf{b}_2 + \cdots + c_m \mathbf{b}_m \quad (6.1)$$

where $\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_m$ are smooth basis functions and c_1, c_2, \dots, c_m are the coefficients related to each basis vector. It is useful to represent the equation in the vector-matrix representation as

$$\mathbf{f} = \mathbf{B}\mathbf{c} \quad (6.2)$$

Here $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_m]$ is a $31 \times m$ matrix with each column representing a basis function and the vector $\mathbf{c} = [c_1, c_2, \dots, c_m]$ contains m -component coefficients.

By judicious choice of basis, we can effectively bound the smoothness of the filter solution. In this work, we adopt the discrete cosine series expansion [81] and take linear combinations of the first m orthonormal basis - each basis vector has unit length and is perpendicular to the other basis - to make smooth filters. In Figure 6.2 we show a smooth filter example constructed by a linear combination of the the cosine basis functions . Because the individual terms of the cosine basis are smooth, any linear combination of this basis is also smooth.

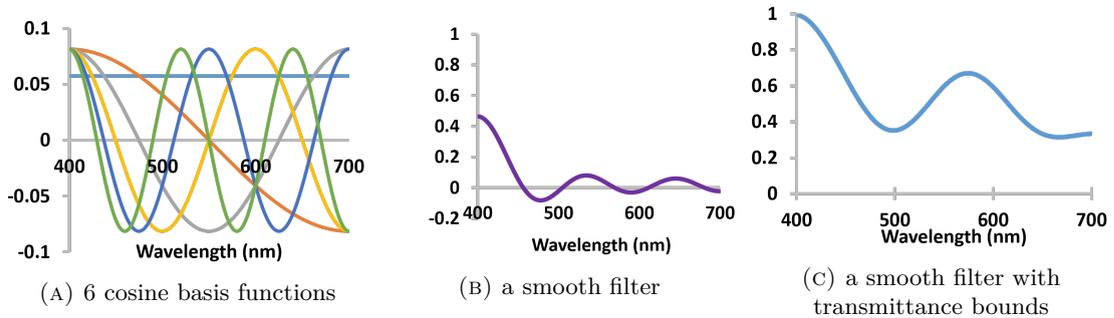


FIGURE 6.2: (A) A set of 6-cosine basis functions, (B) a smooth filter example made from linear combinations the basis functions, and (C) a smooth filter made from linear combinations the basis functions and also bounded with min and max transmittances.

6.2.2 Transmittance Constraint

Let us denote the minimum and maximum filter transmittance bounds as:

$$f_{min} \preceq \mathbf{f} \preceq f_{max}. \quad (6.3)$$

The scalars f_{min} and f_{max} denote the lower and upper thresholds on the desired transmittance of the filter; specifically, f_{max} is set to 100% by default as fully transmissive and f_{min} is a positive value between 0 and 100%. The symbol \preceq denotes the component-wise inequality, i.e. every element in the filter vector follows $f_{min} \leq f_i \leq f_{max}, i = 1, 2, \dots, 31$. Under these constraints, the filter transmittance should fall within the range of the upper threshold f_{max} and the lower threshold f_{min} over the whole spectrum. Our desire is to let the filter overall transmit as much light as possible. We can raise up the lower boundary f_{min} while keeping $f_{max} = 100\%$ (as fully transmissive).

We can also raise the overall transmittance by

$$\frac{1}{31} \sum_{i=1}^{31} f_i \geq f_{avg} \quad (6.4)$$

where the scalar f_{avg} denotes the desired average transmittance value. That is, if we set $f_{avg} = 50\%$, when there comes the equal-energy light, then the filter will transmit at least 50% amount of the light. Equivalently, we can rewrite (6.4) in the vector representation as

$$\frac{1}{31} \mathbf{1}^T \mathbf{f} \geq f_{avg} \quad (6.5)$$

where the uniform vector is $\mathbf{1}^T = [1, 1, \dots, 1]$.

The constraints introduced above can be added to the filter individually or combined. In Results Section, we will show optimised filters under various combinations and levels of controls.

6.3 Filter Optimisation with Constraints

We mainly deploy the following combination where the target filter \mathbf{f} can be represented according to:

$$\mathbf{f} = \mathbf{B}\mathbf{c}, \quad \text{s.t.} \quad f_{min} \preceq \mathbf{f} \preceq f_{max} \quad (6.6)$$

where we enforce the optimised filter to be in the span of the column vectors of \mathbf{B} and also to meet the minimum and maximum transmittance constraints. The average constraint in (6.4) is not added since when we raise up the minimum transmittance f_{min} , we usually have a good average transmittance.

With respect to the constrained filter representation in (6.6), we can, respectively, rewrite the filter design methods given in the Luther-condition based, Vora-Value, and Data-driven optimisations.

6.3.1 Constrained Luther-Condition Optimisation

Let us rewrite the Luther-condition based filter optimisation in (3.4) under the following constraints as

$$\min_{\mathbf{c}, \mathbf{M}} \|\text{diag}(\mathbf{B}\mathbf{c})\mathbf{Q}\mathbf{M} - \mathbf{X}\|_F^2 \quad \text{s.t.} \quad f_{min} \preceq \mathbf{B}\mathbf{c} \preceq f_{max} \quad (6.7)$$

where the \mathbf{X} denote the CIEXYZ colour matching functions.

6.3.1.1 Solving for Constrained Luther-filters

The optimisation in (6.7) has no closed-form solution. Although we can no longer use the least-regressions to solve the filter directly, after some necessary changes being made, the coefficient vector \mathbf{c} and matrix \mathbf{M} can still be solved following the iterative ALS scheme given in Algorithm 3.1 which is used for solving the unconstrained filter optimisation. Specifically, we solve for the filter coefficient vector \mathbf{c} by holding the matrix \mathbf{M} fixed and alternatively using the newly solved filter to solve for the matrix \mathbf{M} and the process will continue updating both matrices in turn until it converges to a predefined error threshold.

In the following, we will demonstrate how to in turn solve the filter and the correction matrix for the new optimisation under constraints. Actually as pointed out in Section 5.3, the Luther optimisation can be seen as a particular case of the Data-driven method (when the colour signal matrix is set to an identity matrix $\mathbf{C} = \mathcal{I}_{31}$). We present the detailed procedures here. The derivation given below is similar to that given in Appendix A.3 but much simpler.

Given a known coefficient vector \mathbf{c} for the corresponding basis matrix \mathbf{B} , the correction matrix \mathbf{M} can be solved by:

$$\mathbf{M} = [\mathbf{Q}^T \text{diag}(\mathbf{B}\mathbf{c}) \text{diag}(\mathbf{B}\mathbf{c}) \mathbf{Q}]^{-1} \mathbf{Q}^T \text{diag}(\mathbf{B}\mathbf{c}) \mathbf{X}. \quad (6.8)$$

In order to solve the filter given in (6.8), we need make changes to the optimisation formulation. It is useful to *vectorise* the minimisation. The vectorisation, denoted $\text{vec}()$, transforms a matrix to a vector by stacking its columns on top of one another. By using the vectorisation, we can rewrite the minimisation as:

$$\| \text{vec}(\text{diag}(\mathbf{f})\mathbf{Q}\mathbf{M}) - \text{vec}(\mathbf{X}) \|_F^2 \quad (6.9)$$

Now let us rewrite the diagonal filter matrix as a summation of each value in the diagonal, f_i , with a single entry matrix \mathbf{D}_i as $\text{diag}(\mathbf{f}) = \sum_{i=1}^{31} f_i \mathbf{D}_i$. \mathbf{D}_i is a 31×31 matrix with a single non-zero entry $\mathbf{D}(i, i) = 1$. By substituting the filter matrix using this new representation into (6.9), we obtain

$$\text{vec}(\text{diag}(\mathbf{f})\mathbf{Q}\mathbf{M}) = \sum_{i=1}^{31} f_i \text{vec}(\mathbf{D}_i \mathbf{Q}\mathbf{M}) \quad (6.10)$$

as the vectorisation operator and summation are commutative.

Denoting $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{31}]$ where each column is a vector of $\mathbf{v}_i = \text{vec}(\mathbf{D}_i \mathbf{Q}\mathbf{M})$. Equation (6.10) can be expressed more compactly using matrix-vector multiplication as

$$\text{vec}(\text{diag}(\mathbf{f})\mathbf{Q}\mathbf{M}) = \mathbf{V}\mathbf{f} \quad (6.11)$$

Note that $\mathbf{D}_i \mathbf{Q}\mathbf{M}$ has the size of 31×3 , so after vectorisation, each vector \mathbf{v}_i is a 93×1 vector which makes the matrix \mathbf{V} in size of 93×31 .

Denoting \mathbf{w} for $\text{vec}(\mathbf{X})$ and $\mathbf{f} = \mathbf{B}\mathbf{c}$, the Luther-condition filter optimisation under constraints can be rewritten as

$$\min_{\mathbf{c}} \| \mathbf{V}\mathbf{B}\mathbf{c} - \mathbf{w} \|_F^2 \quad \text{s.t.} \quad f_{min} \preceq \mathbf{B}\mathbf{c} \preceq f_{max} \quad (6.12)$$

As the Frobenius norm has the property of $\| \mathbf{A} \|_F^2 = \mathbf{A}^T \mathbf{A}$, the above minimisation can be expanded and results in the form of a quadratic problem, subject to linear inequality constraints:

$$\min_{\mathbf{c}} \mathbf{c}^T \mathbf{B}^T \mathbf{V}^T \mathbf{V} \mathbf{B} \mathbf{c} - 2\mathbf{w}^T \mathbf{V} \mathbf{B} \mathbf{c} \quad \text{s.t.} \quad f_{min} \preceq \mathbf{B}\mathbf{c} \preceq f_{max} \quad (6.13)$$

where the scalar term $\mathbf{w}^T \mathbf{w}$ is omitted.

In (6.13), we actually have two inequalities, i.e. $\mathbf{B}\mathbf{c} \succeq f_{min}$ and $\mathbf{B}\mathbf{c} \preceq f_{max}$. In order to convert (6.13) into the standard quadratic problem formulation [49] (which only allows one inequality constraint), we make a simple modification:

$$\begin{bmatrix} \mathbf{B} \\ -\mathbf{B} \end{bmatrix} \mathbf{c} \preceq \begin{bmatrix} \mathbf{f}_{max} \\ -\mathbf{f}_{min} \end{bmatrix} \quad (6.14)$$

Now we can use the Quadratic Programming function to solve for the the filter coefficient vector \mathbf{c} . Finally, the filter solution can be calculated as $\mathbf{f} = \mathbf{B}\mathbf{c}$.

6.3.2 Constrained Vora-Value Optimisation

By substituting the new filter representation in (6.6) into the Vora-Value filter optimisation given in (4.7) – where the objective function $\nu(diag(\mathbf{f})\mathbf{Q}, \mathbf{X})$ is maximised, we have the explicit formulation as

$$\begin{aligned} \arg \max_{\mathbf{c}} \text{trace} \left(diag(\mathbf{B}\mathbf{c})\mathbf{Q}[\mathbf{Q}^T diag(\mathbf{B}\mathbf{c})diag(\mathbf{B}\mathbf{c})\mathbf{Q}]^{-1}\mathbf{Q}^T diag(\mathbf{B}\mathbf{c})\mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T \right) \\ \text{s.t. } f_{min} \preceq \mathbf{B}\mathbf{c} \preceq f_{max}. \end{aligned} \quad (6.15)$$

6.3.2.1 Solving for Constrained Vora-filters

The gradient technique can still be used for solving (6.15) with the optimal filter solution under desired constraints.

Using the chain rule for $\mathbf{f} = \mathbf{B}\mathbf{c}$, the derivative of $\nu(diag(\mathbf{B}\mathbf{c})\mathbf{Q}, \mathbf{X})$ with respect to \mathbf{c} is captured by:

$$\frac{\partial \nu(diag(\mathbf{B}\mathbf{c})\mathbf{Q}, \mathbf{X})}{\partial \mathbf{c}} = \mathbf{B}^T \frac{\partial \nu(diag(\mathbf{f})\mathbf{Q}, \mathbf{X})}{\partial \mathbf{f}} \quad (6.16)$$

or, equivalently in its explicit form of the gradient with respect to the filter \mathbf{f} as

$$\frac{\partial \nu(diag(\mathbf{B}\mathbf{c})\mathbf{Q}, \mathbf{X})}{\partial \mathbf{c}} = \mathbf{B}^T \text{ediag} \left((\mathbf{I} - \mathbf{P}\{\mathbf{F}\mathbf{Q}\})\mathbf{P}\{\mathbf{X}\}\mathbf{P}\{\mathbf{F}\mathbf{Q}\}diag(\mathbf{B}\mathbf{c})^{-1} \right) \quad (6.17)$$

where $\mathbf{P}\{\}$ denotes the projector of the matrix as defined in (4.6) and $\text{ediag}()$ denotes the extraction of the diagonal to form a vector. Of course, the filter also needs to satisfy the transmittance bounds in the range of $[f_{min}, f_{max}]$. The gradient method with inequality constraints can be solved by using the Projected Gradient Methods [9, 12].

In Section 4.4.2, we have shown the equivalence of the Vora-Value optimisation and the modified Luther condition (when the orthonormal basis functions of the XYZ CMFs is

used as the target set). The constrained Vora-Value optimisation can be alternatively formulated as

$$\arg \min_{\mathbf{c}, \mathbf{M}} \| \text{diag}(\mathbf{Bc})\mathbf{QM} - \mathbf{V} \|_F^2, \text{ s.t. } f_{min} \preceq \mathbf{Bc} \preceq f_{max}. \quad (6.18)$$

This equation has the same formulation as the constrained Luther-condition filter optimisation in (6.7) only where \mathbf{V} is used rather than the XYZ CMFs \mathbf{X} . Therefore, we could employ the same ALS procedures as introduced in the last section.

Compared to the gradient algorithm, the ALS algorithm is much simpler to implement and converges more quickly to the optimal answer.

6.3.3 Constrained Data-driven Optimisation

With respect to the new filter representation, we rewrite the Data-driven filter design optimisation in (5.4) as

$$\min_{\mathbf{c}, \mathbf{M}_j} \sum_{j=1}^{cnt} \| \mathbf{C}_j^T \text{diag}(\mathbf{Bc})\mathbf{QM}_j - \mathbf{C}_k^T \mathbf{X} \|_F^2, \quad f_{min} \preceq \mathbf{Bc} \preceq f_{max} \quad (6.19)$$

6.3.3.1 Solving for Constrained Data-filters

The current minimisation can be solved using the same alternating least-squares paradigm of Algorithm 5.1. But, in Step 5, we need to substitute the positive constraint $\mathbf{f}^i > \mathbf{0}$ with

$$f_{min} \preceq \text{diag}\left(\prod_{s=0}^{i-1} \mathbf{f}^s\right) \mathbf{f}^i = \mathbf{Bc}^i \preceq f_{max} \quad (6.20)$$

That is, the filter we find at the i th iteration when multiplied by all the filters from the previous iterations is constrained to be in the basis \mathbf{B} .

Looking at (6.20), we see that

$$\mathbf{f}^i = [\text{diag}\left(\prod_{s=0}^{i-1} \mathbf{f}^s\right)]^{-1} \mathbf{Bc}^i \quad (6.21)$$

That is, effectively the basis for the i th filter changes at each iteration. Again we can solve Step 5 subject to the constraints of (6.21) using Quadratic Programming.

Finally, we note that we could, of course, rewrite Algorithm 5.1 so that at each iteration, we solve for a filter defined by a coefficient weight directly (we could solve for an m -term coefficient vector rather than a 31-component filter). The two formalisms are equivalent.

Here, we chose to solve for the per iteration filter for notational convenience: we can use the general Data-driven algorithm and simply change how we calculate the per iteration filter optimisation.

6.3.4 Multi-initialisation Conditions

Alternating least-squares is guaranteed to converge but it will not necessarily return the global optimal result. But, it is deterministic. So, given the state of the correction matrices and filter at the i th iteration, we will ultimately arrive at the same solution.

Equally, if we change the initialisation condition, \mathbf{f}^{seed} , in Algorithm 5.1, we may end up solving for a different filter. Empirically, we observed that the Data-filters returned by the data-driven algorithm depends strongly on the initial filter that seeds the optimisation, though there is little change for Luther-filters and Vora-filters. We will investigate the effect of initialisation conditions for data-driven algorithm in details.

Let us consider 3 different ways to seed the Data-driven optimisation:

- 1) *Default*: $\mathbf{f}^{seed} = \mathbf{1}$. This uniform unit vector – $[1, 1, \dots, 1]^T$ in size of 31×1 – denotes a fully transmissive filter over the spectrum.
- 2) *Optimised-filter*: $\mathbf{f}^{seed} = \mathbf{f}^{Luther}$ or $\mathbf{f}^{seed} = \mathbf{f}^{Vora}$. We seed the Data-driven optimisation with optimised filter, e.g. the Luther-filter found using Algorithm 3.1 or the Vora-filter found using Algorithm 4.1.
- 3) *By sampling*: Here we find a set of sample filters \mathcal{F} (which meet our smoothness and transmittance boundedness constraints) and for each filter in the sample set, $\mathbf{f}^{seed} \in \mathcal{F}$, we will run Data-driven in Algorithm 5.1.

Algorithm 6.1 generates $\#filters$ (number of initial filters)—subject to bounded smoothness constraints—by uniformly and randomly sampling the filter coefficient space. Before sampling, the algorithm first finds the min and max values of the coefficients (which are calculated in each of the m dimensions individually). Explicitly, for the i th component in vector \mathbf{c} , we denote its minimum and maximum values as c_i^{min} and c_i^{max} .

In Algorithm 6.1, for the minimum value of the i th coefficient, we write:

$$c_i^{min} = \min c_i, \text{ s.t. } f_{min} \preceq \mathbf{Bc} \preceq f_{max}, i = 1, 2, \dots, m \quad (6.22)$$

That is, over all possible coefficient vectors \mathbf{c} , which satisfy the transmittance constraints, we take note of the minimum value of the i th component. That is, we find the minimum value that c_i can be over the set of all possible solutions. The maximum of the i th

Algorithm 6.1 Algorithm for generating an initial filter set

```

1:  $\mathcal{F} = \{\}$ 
2:  $c_i^{min} = \min c_i$ , s.t.  $f_{min} \preceq \mathbf{Bc} \preceq f_{max}$ ,  $i = 1, 2, \dots, m$ 
3:  $c_i^{max} = \max c_i$ , s.t.  $f_{min} \preceq \mathbf{Bc} \preceq f_{max}$ ,  $i = 1, 2, \dots, m$ 
4: while  $cardinality(\mathcal{F}) < \#filters$  do
5:    $c_i \sim U(c_i^{min}, c_i^{max})$ ,  $i = 1, 2, \dots, m$ 
6:    $\mathbf{f} = \mathbf{Bc}$ , ( $\mathbf{c} = [c_1 \ c_2 \ \dots \ c_m]^T$ )
7:   if  $f_{min} \preceq \mathbf{f} \preceq f_{max}$  &  $\{\forall \mathbf{q} \in \mathcal{F} : angle(\mathbf{f}, \mathbf{q}) > \theta\}$  then
8:      $\mathcal{F} \leftarrow \mathcal{F} \cup \{\mathbf{f}\}$ 
9:   end if
10: end while

```

coefficient term is written similarly (see Step 3, Algorithm 6.1). The minimum and maximum values of c_i can be solved using Linear Programming [31].

All m min and max components, taken together, make the two vectors \mathbf{c}^{min} and \mathbf{c}^{max} . These vectors together define the extremal values in each dimension of an m -dimensional hypercube. A vector that lies outside the hypercube is guaranteed not to satisfy the boundedness and transmittance constraints we have placed on our filters. This hypercube usefully delimits our search space (of the sample set of solutions).

To generate a set of filters for initialising the optimisation (solved in the Data-driven Algorithm), we will sample uniformly and randomly this hypercube. We use the notation $c_i \sim U(c_i^{min}, c_i^{max})$ to denote sampling a number in the interval $[c_i^{min}, c_i^{max}]$ uniformly and randomly. A filter constructed from the corresponding vector $\mathbf{c} = [c_1 \ c_2 \ \dots \ c_m]^T$ ($\mathbf{f} = \mathbf{Bc}$), will be added into the initial filter set \mathcal{F} only if it lies within the transmittance bounds **and** it is sufficiently far from those filters already in the set, see Step 7. In Algorithm 6.1, *sufficiently far* means at least θ degrees from the other set members. The function $cardinality()$ returns the number of members in a set.

6.4 Results

In this section, we will solve for filters that are smooth and transmissive. We enforce smoothness indirectly by assuming that our filters lie within the span of either a 6-, 8-, or 10-dimensional cosine basis. At the same time, the filter should transmit, per wavelength, a minimum amount of incident light (e.g. 20%, 30% and more).

6.4.1 Constrained Luther-filters

For the constrained Luther-filters, we will carry out two experiments with different filter constraining conditions. In the first experiment, we are solving for the optimal

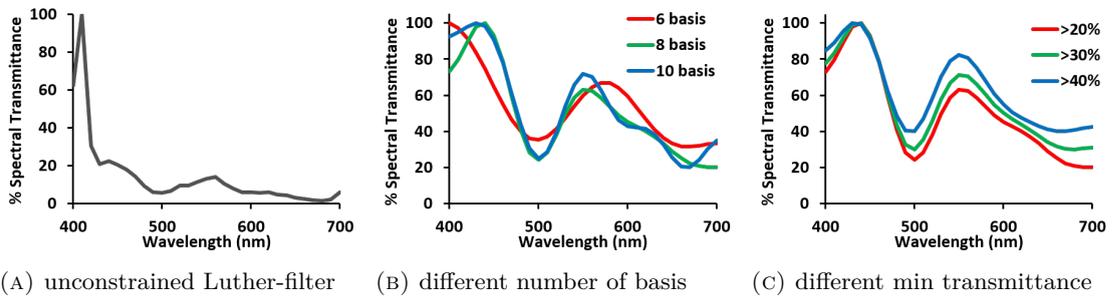


FIGURE 6.3: Spectral transmittance of Luther-optimised filters. (A) filter with no constraint (in black), filters constructed by 6-, 8-, and 10-term cosine basis functions with a fixed minimum transmittance of 20% (coloured solid lines). (B) filters with varying minimum transmittance of 20%, 30% and 40% composed by 8-cosine basis functions.

filter solutions of the testing camera using 6-, 8-, and 10-cosine basis functions from the constrained Luther-condition optimisation. In these cases, we seek the filters with a fixed minimum transmittance that pass at least 20% of the incident light. For each experimental condition (the number of basis functions), we solve for the optimal filter solution (equivalently the coefficients for the cosine basis) and the best correction matrix.

The spectral transmittance distribution of the optimal filter solutions under these conditions are shown in Figure 6.3b. For reference, we also plot the best Luther filter with no constraint, i.e. we run the original optimisation (see Figure 6.3a). Evidently, the fewer cosine basis functions we use, the smoother the filter becomes.

In the second experiment, the best filters with a varying minimum transmittances of 20%, 30% and 40% composed by 8-cosine basis are also calculated. The spectral distribution of the solved-for filters are shown in Figure 6.3c.

We evaluate how good the filtered camera performs in the colour measurement experiments in terms of the perceptual colour errors – CIELAB colour difference metrics. The colour correction experiments are performed for a set of 102 illuminants and 1995 reflectance spectra as described in Section 2.3. We calculate the RGBs of all reflectance spectra under each illuminant, and find the best 3×3 correction matrices mapping RGBs to the ground-truth XYZs before converted into CIELAB colour space. Then the overall mean, median, 90-percentile, 95-percentile, 99-percentile and max of ΔE_{ab}^* are averaged over the all test lights.

The colour measurement results are given in Table 6.1. Under the 20% transmittance bound, we can see that the optimal filter linearly composed by 8 cosine basis (see the red line in Figure 6.3b denoted ‘LUTH_8cos’) outperforms the other two smoothness conditions: it reduces nearly two-thirds of the colour errors across all statistical metrics comparing to those by the linear colour correction. Although constrained filters perform

TABLE 6.1: ΔE_{ab}^* statistics of the colour corrected **NAT**ive camera, the colour corrected camera with the **LUTHer**-condition optimised filter (without constraint), and the colour corrected camera with the **LUTHer**-condition optimised filters under various constrained conditions for Canon 40D camera.

	Mean	median	90%	95%	99%	max
NAT	1.72	1.03	3.68	5.12	12.94	28.39
LUTH	0.44	0.22	1.07	1.48	3.19	8.77
Filter transmittance $f \succeq 20\%$						
LUTH_6cos	0.94	0.54	2.03	2.84	7.00	21.14
LUTH_8cos	0.62	0.38	1.41	2.01	3.47	9.53
LUTH_10cos	0.69	0.42	1.60	2.27	3.89	10.06
Using an 8-term cosine basis						
$f \succeq 20\%$	0.62	0.38	1.41	2.01	3.47	9.53
$f \succeq 30\%$	0.69	0.41	1.59	2.22	3.99	12.69
$f \succeq 40\%$	0.83	0.46	1.89	2.63	5.39	16.62

less well than the non-constrained Luther optimal filter, it gains greatly on the filter transmissivity and smoothness.

The best filter for minimum transmittances of 20%, 30% and 40% are also calculated. The spectral distribution of the solved-for filters are shown in Figure 6.3c and their colour correction results are given in the bottom rows in Table 6.1. As expected, greater minimum threshold will lead to less effective colour reduction. Yet a filter having a minimum transmittance value of 40% (an overall of 63% transmittance which is reasonably transmissive) can still make a camera significantly more colorimetric by reducing nearly half of the colour errors.

6.4.2 Constrained Vora-filters

The Vora-Value optimised filters for a Canon 40D digital camera are shown in Figure 6.4. In black, we show the derived optimal filter found when no constraint (smoothness nor min/max transmittances) is applied. In red, filters are described by the first 8 terms in a cosine basis expansion and constrained to be between 20% and 100% transmissive. Finally in dashed green we increase the minimum transmission to 30%.

In Table 6.2, we evaluate the derived filters with respect to a colour measurement experiment in terms of colour error metric ΔE_{ab}^* statistics. We include the results of the unfiltered **NAT**ive camera sensor as baseline results. The prior art of **LUTHer**-filters

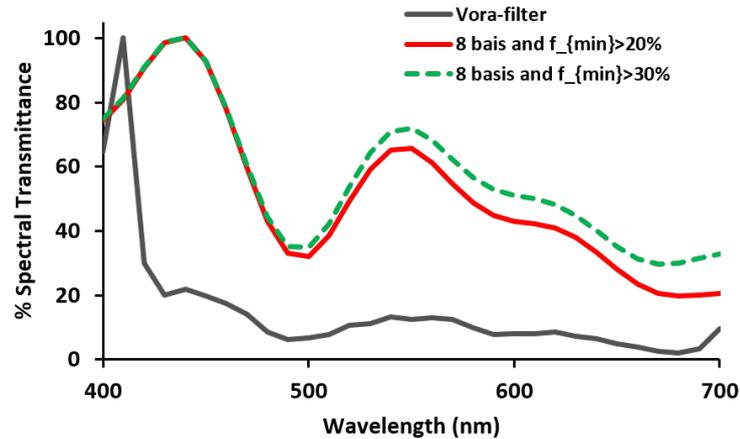


FIGURE 6.4: Spectral transmittance of optimised Vora-filters for Canon 40D camera. The filter in solid black line is solved with no constraint applied. The filters in red and dashed green lines are modelled under 8-cosine basis functions and constrained by the minimum transmittance of 20% and 30% respectively.

under the same constraint conditions is also given for comparison. We repeat this experiment where we optimise for smooth (linearly constructed by the first 8-cosine basis) and transmittance bounded **VORA**-filters (at 20% and 30% minimum transmittance limits).

From the table, we can see that although the performance is not as good as for the unconstrained filters, the results are still quite good and significantly better than using no filter. Moreover, the **VORA**-filters offer slightly better results (about 10% in colour error reduction) than the **LUTHER**-filters both for constrained and unconstrained conditions.

TABLE 6.2: ΔE_{ab}^* statistics of the colour corrected **NATIVE** camera, the colour corrected camera with **LUTHER**-condition and **VORA**-Value optimised filters (without constraint), and the colour corrected camera with the **LUTHER**- and **VORA**-filters under various constraining conditions for Canon 40D camera

	mean	median	95%	99%	max
NAT	1.72	1.03	5.12	12.94	28.39
filter without constraint					
LUTH	0.44	0.22	1.48	3.19	8.77
VORA	0.38	0.20	1.23	2.97	9.89
constrained filters $\mathbf{f} = \mathbf{Bc}$					
condition I: $0.2 \preceq \mathbf{f} \preceq 1$					
LUTH	0.62	0.38	2.01	3.47	9.52
VORA	0.51	0.29	1.61	3.48	12.05
condition II: $0.3 \preceq \mathbf{f} \preceq 1$					
LUTH	0.69	0.41	2.22	3.99	12.69
VORA	0.62	0.36	1.98	3.93	13.60

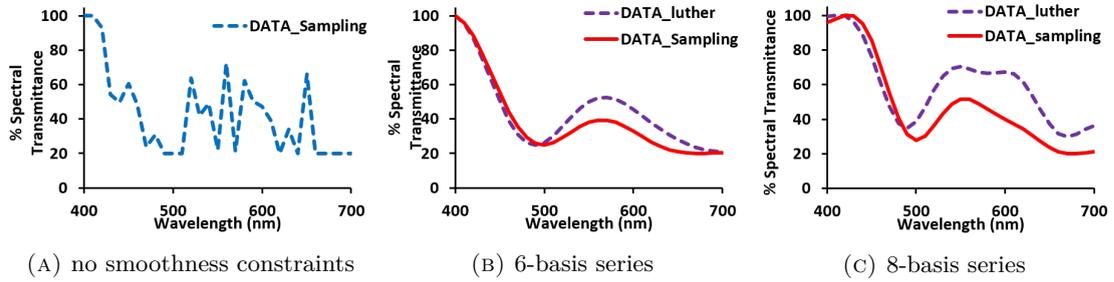


FIGURE 6.5: **Data-driven** optimised filters for Canon 40D camera: (A) filters solved with no smoothness but the min-and-max transmittance constraint, (B) and (C) filters constructed by 6- and 8-cosine basis functions respectively. Dotted purple lines (**DATA_luther**) show the filters initialised with the Luther-optimised filter, solid red lines (**DATA_sampling**) show the filters initialised with a sampling filter set. All filters shown are constrained with a fixed minimum transmittance of 20%.

6.4.3 Constrained Data-filters

Let us visualise the 20% bounded transmittance constraint using the Canon 40D camera sensitivities and our Data-driven optimisation. First, we initialise the optimisation (Algorithm 5.1) using the Luther-condition optimised filter as the *seed* filter. The optimisation returns the filter, denoted **DATA_luther** and is plotted in the blue-dashed line in Figure 6.5a.

We repeat this experiment where we both require the derived filters to transmit at least 20% of the light and also that they belong to the span the first 6- or first 8-terms in a cosine basis expansion. When the recovered filter is constrained to lie in the span of a 6-dimensional cosine basis, the recovered **DATA_luther** filters are shown in Figure 6.5b (respectively blue and purple dashed lines). See Figure 6.5c for the filters calculated using an 8-dimensional cosine basis.

The red lines shown in Figures 6.5b and 6.5c are the filters optimised by our sampling algorithm (using Algorithm 6.1 to find a set of filters to seed Algorithm 5.1 and then choosing the best one that has the best overall performance). The experiment for deriving these sampled filters are described in the next subsection.

By examining Figures 6.5a, 6.5b and 6.5c, it is evident that - when no basis, a 6- or an 8-dimensional cosine basis are used - that changing the initialisation condition results in a different filter being found.

6.4.4 Sampled-based Optimisation

Using Algorithm 6.1, let us run a sampled optimisation for **Data-driven** filters. That is, for a given cosine basis, we calculate a set of candidate solutions, the sample set \mathcal{F} .

TABLE 6.3: ΔE_{ab}^* statistics of the colour corrected **NAT**ive camera, the colour corrected camera with the **DATA**-driven optimised filter solutions obtained when initialised with **Luther**-condition filter and **Sampling** filter set respectively under different constraints for Canon 40D camera

	Mean	Median	90%	95%	99%	Max
NAT	1.72	1.03	3.68	5.12	12.94	28.39
minimum transmittance of 20%						
DATA_luther	0.58	0.38	1.36	1.80	2.77	5.75
6 cosine basis with 20% minimum transmittance						
DATA_luther	0.94	0.54	2.03	2.84	7.00	21.14
DATA_sampling	0.59	0.35	1.30	1.83	3.77	14.19
8 cosine basis with 20% minimum transmittance						
DATA_luther	0.62	0.38	1.41	2.01	3.47	9.52
DATA_sampling	0.45	0.25	1.02	1.41	3.10	10.63

Here we populate \mathcal{F} with 20,000 uniformly and randomly generated filters where the angular threshold between any two filters in the set is 1 degree, $\theta = 1^\circ$, see Step 7 in Algorithm 6.1. Each filter in \mathcal{F} transmits at least 20% of the light (and \mathcal{F} is populated by filters described as linear combination of a 6- or 8-dimensional cosine basis).

Each filter in \mathcal{F} is used to initialise the Data-driven algorithm and we will find 20,000 optimised filters. The colour measurement performance of each filter in this solution set can be calculated. Then we simply choose the filter that delivers the best overall measurement performance. In Figures 6.5b and 6.5c, we show the best sample-optimised filters (red lines) which respectively lie in the span of a 6- and 8-dimensional cosine basis (and transmit at least 20% of the light). Here 'best' is defined to be the filter that results in the smallest mean ΔE_{ab}^* performance.

Table 6.3 reports the ΔE_{ab}^* colour error performance 1995 reflectances and 102 lights [6] for the Canon 40D sensitivities. The row **NAT** reports the baseline colour correction results when a per illuminant based linear correction matrix is applied while no filter is used.

In Table 6.3, we report the correction performances in 3 tranches. Row 2 corresponds to the filter without using cosine basis as shown in Figure 6.5a. Here we find the best filter with only the 20% minimum transmittance bound. Rows 3 and 4 report the performance when the 2 filters shown in Figure 6.5b are used, where the filter is additionally constrained to be in the span of the 6-dimensional cosine basis. Finally,

when the filter is constrained to belong to an 8-dimensional cosine basis, the 2 derived filters lead to the error statistics shown in rows 5 through 6.

Table 5.1 reported the colour measurement performance of the filters found using an unconstrained optimisation. Table 6.3 reports the colour measurement results that are found when filters are constrained to have a bounded transmittance (here at least 20% of the light) and be smooth. Let us consider the bounded transmittance first.

Comparing row 9 of Table 5.1 to row 2 of Table 6.3, we see that adding a lower transmittance bound returns a filter that delivers poorer measurement performance (but still much better compared with the native camera response). Additionally, requiring that our filters smooth on top of the minimum also yields relatively poorer performance compared to the unconstrained filter optimisation.

However, with either the 6- and 8-dimensional cosine basis constraint, we can find the best filter by seeding Algorithm 5.1 with many possible filter initialisations (and then choosing the best filter overall). Here, we find that comparable performance is possible. Compare rows 4 and 6 of Table 6.3 to row 9 of Table 5.1. It is remarkable how well a constrained filter can work: the performance is ever so slightly worse than the unconstrained optimisation. But, the filter is much smoother and more likely to be able to be manufactured.

6.4.4.1 High Transmissive Data-filters

The best filters with varying minimum transmittance of respectively 100% (i.e. no filter is applied), 80%, 60%, 40% and 20% (all linearly composed by 8-cosine basis using sampled optimisation) are also calculated. For each filter, we perform the colour correction experiment where we use the same data sets as in Table 6.3 (1995 reflectance spectra and 102 illuminant spectra). The colour correction results are given in Table 6.4. We also report the exposure adjustment required so that the filtered camera captures the same amount of light as an unfiltered camera (see the last column). The exposure numbers are calculated as the reciprocal of the averaged transmittance over the spectrum (when an equal-energy spectrum light is assumed).

A filter having a minimum transmittance value of 20% needs a doubling exposure to match the same light level as the unfiltered condition. The best filter with a minimum transmittance of 40% still makes the camera much more colorimetric but only a 60% increase in exposure is required. Exposure might be changed by opening the aperture a fraction, capturing over a slightly longer exposure time or by applying a scaling factor. When exposure is altered by applying a scaling factor (e.g. increasing the ISO number) then for low-light scenes there may be an increase in the conspicuity of noise.

TABLE 6.4: ΔE_{ab}^* statistics and relative exposure settings of the colour corrected camera with the optimised **Sampling** filter set under varying constraints of the minimum transmittance constructed by 8-cosine basis for Canon 40D camera

	Mean	Median	90%	95%	99%	Max	Exposure Setting
f_min = 100%	1.72	1.03	3.68	5.12	12.94	28.39	1.00
f_min = 80%	1.43	0.86	3.08	4.24	10.91	24.97	1.13
f_min = 60%	1.10	0.68	2.34	3.29	8.00	21.53	1.29
f_min = 40%	0.75	0.44	1.69	2.44	4.87	16.11	1.60
f_min = 20%	0.45	0.25	1.02	1.41	3.10	10.63	2.07

6.4.4.2 Sampled Initialisation

It is worth reflecting on our sample-based optimisation. Clearly, that sampling makes such a difference to the performance that our optimisation can deliver (for filtered colour measurement) teaches us that the minimisation at hand has many local minima. By sampling we are effectively allowing our minimiser (Algorithm 5.1) to find many solutions and then we have the latitude to choose the (closer to) global minimum. Given we seed our optimisation with 20,000 filters we might wonder whether we need to actually carry out the Data-driven optimisation.

In answering this question, first we remark that it is well known that as the dimension of a space increases, it is more *sparse*. On the Cartesian plane, if we have more than 360 vectors (anchored at the origin), then the closest angular distance to at least one vector's nearest neighbours must be less than 1 degree. In 3-dimensions, we can have thousands of vectors where every vector is more than 1 degree from its nearest neighbour.

So, let us investigate the number of sampled smooth filters we need for finding a good approximation to the optimal filter solution, e.g. within acceptable deviation (for the data at hand). We test on the 8-Cosine basis condition and evaluate the averaged mean colour errors by varying # filters in the initialisation set from 100, 200, 500, 1000, 2000, 5000, 10,000 to 20,000 (where these filters are selected using Algorithm 6.1). In Figure 6.6, the x-axis represents the number of sampled filters in the initialisation set and the y-axis represents the average mean colour error in terms of ΔE_{ab}^* (for the corresponding optimised filter which is refined for that set of initialisation). One standard deviation error bar is also shown. From the figure, we can see that little can be gained after using about 5000 sampled filters for initialisation. Therefore, by using 20000 sampled filters should be sufficient for our method.

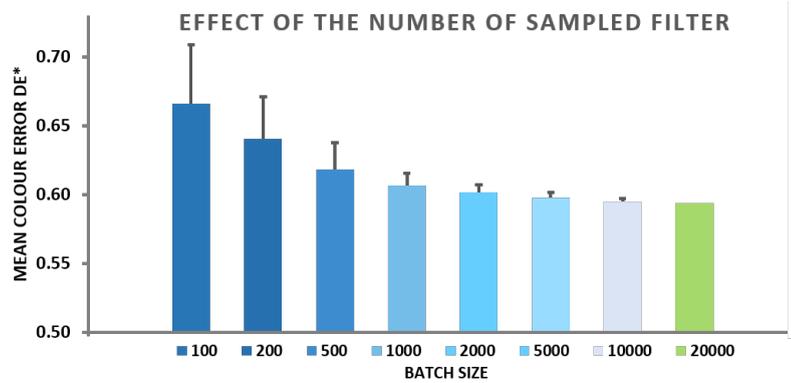


FIGURE 6.6: The effect of the number of filters in the initialisation set on the colour reproduction results in terms of mean colour error assuming filters fall within the 8-dimensional space. The magnitude of the bars represents the average colour error from groups of varying the number of filters. The upper one standard deviation error bar is also shown.

For our 20,000 member sample set \mathcal{F} , we calculated the average angular distance for each element to its nearest neighbour in the set. When \mathcal{F} is calculated subject to the 6-dimensional cosine basis constraint, the average nearest-neighbour distance was 2.6 degrees (with a maximum of 7) and for the 8-dimensional case it was 4.6 degrees (with a maximum of 10). Running the optimisation, Algorithm 5.1, with each element in \mathcal{F} , we effectively *refine* the initial guess. And, the refinement (difference between the starting and endpoint filter) is on the same order as the average nearest-neighbour distance.

Significantly, running the optimisation - carrying out the refinement - results in a significant performance increment compared to using the only sample filters. That is we cannot use the sampling strategy alone to find the best optimised filter. The importance of the refinement step will increase as a greater number of basis functions are used in the optimisation.

6.4.5 Colour Filtering vs Higher-order Correction Methods

We compared our colour filtering method (Data-driven filters smoothed by 8-cosine basis and transmittance constraints under sampled optimisation using Algorithm 6.1) to the hue-plane preserving colour correction (HPPCC) method presented in [55] (which is a data-driven modification of [3]). This method effectively decomposes 3-dimensional colour spaces into a number of cones: defined to be regions of colour spaces that are convex combination of 3 bounding vectors (all of which are anchored at the origin (0,0,0)). All cones share the achromatic axis (1,1,1) and neighbouring cones have two bounding vectors in common. Each cone has its own 3×3 colour correction transform. If an RGB lies within a given cone then this RGB is colour corrected by the correction transform for that cone. Significantly, because neighbouring cones share two bounding vectors the

TABLE 6.5: ΔE_{ab}^* statistics of the colour corrected camera with the **Data-driven** optimised filter by **Sampling** method compared to higher-order colour correction methods for Canon 40D camera

	Mean	Median	90%	95%	99%	Max
LCC	1.72	1.03	3.68	5.12	12.94	28.39
PCC,2	1.40	0.89	3.07	4.19	7.93	18.73
PCC,3	1.25	0.83	2.63	3.48	6.32	31.38
RPCC,2	1.30	0.80	2.73	3.81	7.45	22.94
RPCC,3	1.16	0.72	2.66	3.54	5.53	15.39
HPPCC	1.30	0.79	2.79	4.01	7.33	27.00
Filtering: f_min = 40%	0.75	0.44	1.69	2.44	4.87	16.11
Filtering: f_min = 20%	0.45	0.25	1.02	1.41	3.10	10.63

overall colour correction transform is *continuous* in the sense that the colours on the plane between neighbouring cones are mapped to the same colours using either of the colour correction matrices for each cone. See [55] for more details.

We chose the HPPCC method as it maintains the linear property invariant to exposure (the same HPPCC transform is applied when exposure changes), it is still relatively simple to implement, and it is one of the leading colour correction algorithms available. For our experiments colour space is split into 6 cones.

Finally, we also compared the performance to the linear, polynomial and root-polynomial colour correction regressions (up to degree of three). **LCC** denotes a linear colour correction method using a 3×3 mapping matrix. **PCC,2** denotes a regression of a 2nd order polynomial regression and **PCC,3** denotes the 3rd order polynomial expansion. Polynomial regression is not exposure invariant but a root-polynomial regression is invariant to exposure. The 2nd and 3rd order root-polynomial regressions are denoted **RPCC,2** and **RPCC,3**. colour correction results - using the same data sets as Table 6.4 - are reported in Table 6.5.

From the table, we can see that by adopting reasonably transmissive filter with minimum transmittances of 20% and 40% and using simple linear correction we outperform the higher order conventional colour correction methods by a large margin (these two results denoted **Filtering** are the same as the last two rows in Table 6.4). A filter with minimum transmittance of 60% (the fourth row in Table 6.4) has comparable performance to the best performed 3rd-order root-polynomial results (the fifth row in Table 6.5).

6.5 Discussion

The next challenge of our proposal of colour filters, after the concept supported by the promising results presented in this thesis, lies in the fabrication process. Though improvement have been made – the smoothness of the filter distribution – which will make our filters suitable for manufacturing, there are other technical issues on the fabrication. Besides, our filter design methods also suggest that the colour performance is sensitive at certain wavelength range so that manufacturing the filter with high precision will be necessary.

Of course, one convenient way (without the involvement of manufacture) of proving the effectiveness of the idea of a colour prefilter to a camera is to use commercially available filters that provide similar spectral transmittance property to the desired filter given a camera under investigation, such as the Wratten filter set by East Kodak [53]. The selected filters, however, are not optimal and the improvement, if there is any, will be limited by the similarity of the transmittance pattern between the chosen filter and the optimal filter (that minimise the errors in the colour measurement).

It is necessary to fabricate physical colour filters for determining the effectiveness in applications. Ultimately, measuring colour in the real-scene by the ‘filter+cameras’ setting is the only way to validate the idea of colour filters for colorimetric cameras. Of course, designing the filter that optimises for the fabrication process is beyond the scope of this thesis. Yet, it is of great interest for us to bring the idea forward into manufacture and make a real product.

6.6 Conclusion

In this chapter, we extend the filter optimisation by providing a method for incorporating smoothness and transmittance bounds into the filter optimisations given in Chapters 3, 4, and 5. Initially, when these constraints are adopted, the solved-for filters work less well. However, experiments demonstrated that our optimisations were highly dependent on the initialisation parameters, specifically the *seed* filter (initial guess) that drives the filter evolution. A simple sampling strategy - i.e. severally running the optimisation for a set of judiciously chosen seed filters - allows us to mitigate this problem. Significantly a smooth filter that transmits about 50% of the light across the visible spectrum delivers almost as good performance as a very non-transmissive and non-smooth filter found via the unconstrained optimisations.

We also show the benefit of introducing a colour filter for a camera that a simple linear colour correction is sufficient and can outperform the conventional colour correction methods (even using higher-order terms).

Chapter 7

Conclusion

7.1 Contributions

A novel way of making a digital camera more colorimetric has been introduced in this thesis. We set forth the method of designing a colour filter for a given off-the-shelf camera that when it is placed in front of the camera makes the new ‘filter+camera’ imaging system more colorimetric and thus can be used for accurate colour measurement. We develop both mathematical formulations of the filter design problem and algorithmic implementations. Source codes are available at <https://github.com/zhedazhu1012/Filterdesign>.

A colour imaging system measures colour like a human observer if it satisfies the Luther condition, i.e. the camera spectral sensitivities are a linear transform from the XYZ (or equivalently the cone) sensitivities. Unfortunately, for manufacturability reasons and signal processing practice, commercial RGB cameras are not linear combinations from the XYZ CMFs.

We propose that we can make a camera more colorimetric if we place a specially designed colour filter in front of the camera. We make six contributions. First, we design a colour filter for a given off-the-shelf camera such that the filtered camera better approximates the Luther condition. Our design problem is expressed in two ways. We either find a filter such that the linearly transformed filtered camera sensitivities best match the XYZ CMFs or, in our second contribution, we find the filter that optimises the Vora-Value. The Vora-Value quantifies the extent which the space spanned (or sampled) by the filtered camera sensitivities is similar to the space spanned by the XYZ CMFs. This second optimisation is, advantageously, not tied to any particular target sensitivities (e.g. the cone fundamentals or XYZ CMFs).

The Vora-Value optimisation - though arguably a conceptually superior method - is more complex to formulate and implement than the Luther-optimisation. In our third contribution, we show that if we use an orthonormal variant of the XYZ CMFs (i.e. a linear combination thereof that is orthonormal), then we can use the Luther-condition optimisation to solve for the best Vora-Value filter.

Our fourth contribution is that we extend our methods to find colour filters that take real world reflectances and lights into account. In our fifth contribution, we reformulate our method so that we can add linear inequality constraints into the optimisation formulation. Here we explore bounding the smoothness of the filter (hopefully making it easier to manufacture) and its transmissivity. The latter constraint is important as a correction filter will not be practically useful if it absorbs most of the incident light.

For making physically realisable filters, we restrict the filter solution set to solve for much smoother and more transmissive filters by adding constraints. We reform the optimisations as mentioned above by incorporating filter constraints where a group of smooth basis filters are linearly combined to generate a smooth filter that at the same time, is required to reach a minimum transmittance bound.

In our last contribution, we develop a simple sampling method to the filter optimisation where we systematically sample the set of possible initial filters. For each initialisation, we solve for the best refinement. This sampling method makes a substantial improvement in the performance that our optimisation algorithms can deliver.

For all cameras tested, experiment results demonstrate that by the insertion of the pre-filters, digital cameras perform significantly better in terms of the Luther condition and Vora-Value compared to the cameras without filtering. In the colour measurement simulations on real illuminants and reflectances spectra, we demonstrate that our optimised filters help digital cameras significantly improve colour accuracy compared to the prior arts of colour correction methods. We show that, in principle, by taking pictures through our optimised smooth and transmissive filters, we can make cameras significantly more colorimetric.

7.2 Future Work

There are still several interesting but unexplored issues:

- Currently, our design method is based on a noise-free model. The no-noise assumption simplifies our modelling; however, noise is inevitable for any practical colour imaging system. For one thing, when a colour filter is placed between the

scene and the camera, it will absorb part of the light and transmit the rest. The camera needs to adjust the exposure or ISO settings to match the light intensity as the condition without a filter, especially when the ambient light is dim. As such, of course, the insertion of a prefilter will affect the noise level in the camera responses. It is of great interest to explore how the filter transmittance and exposure will affect the noise level in the colour measurement experiment. Based on our proposal, Vrhel [92] recently remodelled the prefilter design in the presence of noise, which can be a good reference to start with when accounting for the role of noise.

Significantly, any increase in noise due to the filter absorbing light will be magnified in the colour correction step [7, 86]. It is worth digging in to see how the linear correction in this study will propagate noise.

- Our filters are designed either to match the human visual sensitivities or to make the best estimate of the XYZ tristimuli. Yet, they are not quality metrics that relate to a perceptually uniform colour space, such as Figures of Merit [74]. Though a more complicated algorithm for solving the filter is expected, it is worth investigating what optimal filters will look like and how different they will be when modelled in a perceptually uniform colour space.
- In our work, as you may find, the colorimetric performance of the filters we show in the colour measurement test has the presumption that the colour of the light has been correctly discounted. The process of discounting the illuminant colour (which is called the white balance) is a step before the colour correction in the camera processing pipeline. Our eye has the ability (we say colour constancy) to adapt to the colour of the light. We see a white paper appear white both under a bluish fluorescent light or a yellowish incandescent light. However, a digital camera is incapable of discounting such light colour and the white balance process is necessarily employed in the processing pipeline for the purpose. Though, it is hard to balance out the light colour correctly for all lighting conditions. Attention should be paid when the colour filter is to be used in practical applications.
- A filter is designed for every individual camera in the data set. Since many cameras from the same manufacture or brand usually have very similar sensitivities, it is useful to produce a ‘general-purposed’ filter that works well for a group of cameras that have close sensitivity characteristics.
- In the Luther condition filter design, we formulate our optimisation with \mathbf{FQ} for which we interpret the diagonal matrix \mathbf{F} as a prefilter that alters the effective sensitivities of a camera \mathbf{Q} . While, in the data-driven filter design, assume we have a collection of reflectances \mathbf{S} measured under an illuminant \mathbf{E} , to form the

camera colour responses, we write as $\mathbf{S}^T \mathbf{E} \mathbf{F} \mathbf{Q}$. Now, instead of combining the filter and the camera as $\mathbf{E}(\mathbf{F} \mathbf{Q})$, we could combine the illuminant with the filter as $(\mathbf{E} \mathbf{F}) \mathbf{Q}$. Physically, we can regard the latter as the spectral power distribution of the illuminant is changed by the filter before coming into the camera. Bearing this in mind, if we have specified the optimal filter \mathbf{F} for a known camera \mathbf{Q} , the prediction of the visual responses of a scene under a given illuminant \mathbf{E} is equivalent to capture the scene under $\mathbf{E} \mathbf{F}$. Using a spectrally-tunable illumination system (e.g. a spectrally-tunable LED lighting system), we can perform camera colour measurement experiments under one specially-changed illuminant to match that of the desired illuminant condition.

- The ultimate way to examine the utility of our method is to turn our designed filters into manufacturing and fabricate physical filters. The practical effectiveness of the insertion of a prefilter can then be evaluated in measuring real object colours. Although smoothness constraint is imposed for the optimised filters, it is still far from that can be fabricated precisely as desired. It is worth refining our filters by taking more systematic errors in fabrication process into consideration [89]. In the future, we plan to make a filter and then test how well it works in practice.

7.3 Publications

Journal Publications

Finlayson, G. D., and Zhu, Y. Designing color filters that make cameras more colorimetric. *IEEE Transactions on Image Processing* 30 (2021), 853–867.

Zhu, Y., and Finlayson, G. D. A mathematical investigation into the design of prefilters that make cameras more colorimetric. *Sensors* 20, 23 (2020), 6882.

Conference Publications

Zhu, Y., and Finlayson, G. D. Designing a color filter via optimization of Vora-Value for making a camera more colorimetric. In *Color and Imaging Conference* (2020), Society for Imaging Science and Technology, pp. 181–186.

Finlayson, G. D., and Zhu, Y. Unifying optimization methods for color filter design. In *Colour and Visual Computing Symposium* (2020), vol. 2688, CEUR Workshop Proceedings.

Zhu, Y. Designing a physically-feasible colour filter to make a camera more colorimetric. In *London Imaging Meeting* (2020), Society for Imaging Science and Technology.

Zhu, Y., and Finlayson, G. D. An improved optimization method for finding a color filter to make a camera more colorimetric. In *Electronic Imaging* (2020), Society for Imaging Science and Technology.

Finlayson, G. D., and Zhu, Y. Finding a colour filter to make a camera colorimetric by optimisation. In *International Workshop on Computational Color Imaging* (2019), Springer, pp. 53–62.

Finlayson, G. D., Zhu, Y., and Gong, H. Using a simple colour prefilter to make cameras more colorimetric. In *Color and Imaging Conference* (2018), Society for Imaging Science and Technology, pp. 182–186.

Appendix A

Algorithm Implementation

For both alternating least-squares Algorithms 3.1 and 5.1, respectively for Luther-filters and Data-filters, the filter and the colour correction matrices can be found using simple least-squares regression. To remind the reader, given \mathbf{A} and \mathbf{B} - $m \times n$ matrices of rank n where $m \geq n$, then the least-squares regression \mathbf{M} - an $n \times n$ matrix, mapping \mathbf{A} to \mathbf{B} ($\mathbf{A}\mathbf{M} \approx \mathbf{B}$) can be found in closed-form using the Moore-Penrose inverse [31]:

$$\mathbf{M} = [\mathbf{A}^T \mathbf{A}]^{-1} \mathbf{A}^T \mathbf{B} = \mathbf{A}^+ \mathbf{B}$$

A.1 Algorithm for the Luther-condition Optimisation

In Step 5 of Algorithm 3.1, the optimal filter is found by finding scalars that maps each row of \mathbf{Q}^{i-1} to the corresponding row of \mathbf{X} . The best scalar α mapping the vector $\mathbf{v} = [\mathbf{Q}_j^{i-1}]^T$ to $\mathbf{w} = [\mathbf{X}_j]^T$ (for the j th row of the data matrices) can be written in closed form using the Moore-Penrose inverse: $\alpha = \frac{\mathbf{v}^T \mathbf{w}}{\mathbf{v}^T \mathbf{v}}$.

Similarly, in step 5, the Moore-Penrose inverse can be used for finding \mathbf{M} . Denoting $\mathbf{Q} = \mathbf{F}^i \mathbf{Q}^{i-1}$ then $\mathbf{M}^i = \mathbf{Q}^+ \mathbf{X} = [\mathbf{Q}^T \mathbf{Q}]^{-1} \mathbf{Q}^T \mathbf{X}$.

A.2 Algorithms for the Vora-Value Filter Optimisation

Notation

We will use the following notation for the gradient and Hessian matrix (given in the later section in Newton's method) with respect to the n -dimensional filter vector $\mathbf{f} =$

$[\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_n]^T$:

$$\nabla \nu(\mathbf{f}) = \begin{bmatrix} \frac{\partial \nu}{\partial \mathbf{f}_1} \\ \frac{\partial \nu}{\partial \mathbf{f}_2} \\ \vdots \\ \frac{\partial \nu}{\partial \mathbf{f}_n} \end{bmatrix} \quad \text{and} \quad \mathbf{H} = \nabla^2 \nu(\mathbf{f}) = \begin{bmatrix} \frac{\partial^2 \nu}{\partial \mathbf{f}_1^2} & \frac{\partial^2 \nu}{\partial \mathbf{f}_1 \partial \mathbf{f}_2} & \dots & \frac{\partial^2 \nu}{\partial \mathbf{f}_1 \partial \mathbf{f}_n} \\ \frac{\partial^2 \nu}{\partial \mathbf{f}_2 \partial \mathbf{f}_1} & \frac{\partial^2 \nu}{\partial \mathbf{f}_2^2} & \dots & \frac{\partial^2 \nu}{\partial \mathbf{f}_2 \partial \mathbf{f}_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \nu}{\partial \mathbf{f}_n \partial \mathbf{f}_1} & \frac{\partial^2 \nu}{\partial \mathbf{f}_n \partial \mathbf{f}_2} & \dots & \frac{\partial^2 \nu}{\partial \mathbf{f}_n^2} \end{bmatrix} \quad (\text{A.1})$$

or equivalently, using the indices, we can respectively express as $(\nabla \nu)_i = \frac{\partial \nu}{\partial \mathbf{f}_i}$ and $\mathbf{H}_{i,j} = \frac{\partial^2 \nu}{\partial \mathbf{f}_i \partial \mathbf{f}_j}$. From the definitions, we know that $\nabla \nu$ and $\nabla^2 \nu$ are respectively in the size of $n \times 1$ and $n \times n$.

A.2.1 Derivation of Gradient

We will show the derivation of the gradient function of the filter-modified Vora-Value optimisation.

Theorem A.1. $\frac{\partial \nu(\mathbf{FQ}, \mathbf{X})}{\partial \mathbf{F}} = \frac{2}{3} \left((\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\}) \mathbf{P}\{\mathbf{X}\} \mathbf{P}\{\mathbf{FQ}\} \mathbf{F}^{-1} \right) \circ \mathbf{I}$

Proof. The following rules of matrix calculus are used to obtain the required differentials:

$$\begin{aligned} d \operatorname{trace}(\mathbf{U}) &= \operatorname{trace}(d\mathbf{U}) \\ d(\mathbf{UV}) &= \mathbf{U} d\mathbf{V} + d\mathbf{U} \mathbf{V} \\ d(\mathbf{AU}) &= \mathbf{A} d\mathbf{U} \\ d\mathbf{U}^{-1} &= -\mathbf{U}^{-1}(d\mathbf{U})\mathbf{U}^{-1} \end{aligned} \quad (\text{A.2})$$

Using the above rules, we have

$$\begin{aligned} d\nu(\mathbf{FQ}, \mathbf{X}) &= \frac{1}{3} \operatorname{trace} \left(d\mathbf{F} \mathbf{Q} (\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} \mathbf{V} \mathbf{V}^T - \right. \\ &\quad \left. 2\mathbf{F} \mathbf{Q} (\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} d\mathbf{F} \mathbf{Q} (\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} \mathbf{V} \mathbf{V}^T + \right. \\ &\quad \left. \mathbf{F} \mathbf{Q} (\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T d\mathbf{F} \mathbf{V} \mathbf{V}^T \right). \end{aligned} \quad (\text{A.3})$$

Using the acyclic property of trace that $\operatorname{trace}(\mathbf{ABC}) = \operatorname{trace}(\mathbf{BCA}) = \operatorname{trace}(\mathbf{CAB})$, we can move the $d\mathbf{F}$ in each of the term to the end of the formulation. We also use the projector representation of $\mathbf{P}\{\mathbf{X}\}$ to make it more compact as

$$\begin{aligned} d\nu(\mathbf{FQ}, \mathbf{X}) &= \frac{1}{3} \operatorname{trace} \left(\mathbf{Q} (\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} \mathbf{P}\{\mathbf{X}\} d\mathbf{F} - \right. \\ &\quad \left. 2\mathbf{Q} (\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} \mathbf{P}\{\mathbf{X}\} \mathbf{F} \mathbf{Q} (\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} d\mathbf{F} + \right. \\ &\quad \left. \mathbf{P}\{\mathbf{X}\} \mathbf{F} \mathbf{Q} (\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T d\mathbf{F} \right). \end{aligned} \quad (\text{A.4})$$

Now we can make the derivative $\frac{d\nu(\mathbf{FQ}, \mathbf{X})}{d\mathbf{F}}$ to the right side of the equation in (A.4) according to the rule of $\frac{trace(\mathbf{AdF})}{d\mathbf{F}} = \mathbf{A}$. Also remind that the filter solution we are looking for lies in the diagonal of matrix \mathbf{F} , we have

$$\begin{aligned} \frac{\partial \nu(\mathbf{FQ}, \mathbf{X})}{\partial \mathbf{F}_{ii}} = & \frac{1}{3} \left[\mathbf{Q}(\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} \mathbf{P}\{\mathbf{X}\} - \right. \\ & 2\mathbf{Q}(\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} \mathbf{P}\{\mathbf{X}\} \mathbf{FQ}(\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F} + \\ & \left. \mathbf{P}\{\mathbf{X}\} \mathbf{FQ}(\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \right]_{ii} \end{aligned} \quad (\text{A.5})$$

We use $\mathbf{F}^{-1} \mathbf{P}\{\mathbf{FQ}\} = \mathbf{Q}(\mathbf{Q}^T \mathbf{F}^2 \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{F}$ to ease the notation (the diagonality of \mathbf{F} guarantees it to be invertible). Our gradient function can be expressed as

$$\frac{\partial \nu(\mathbf{FQ}, \mathbf{X})}{\partial \mathbf{F}_{ii}} = \frac{1}{3} \left[\mathbf{F}^{-1} \mathbf{P}\{\mathbf{FQ}\} \mathbf{P}\{\mathbf{X}\} - 2\mathbf{F}^{-1} \mathbf{P}\{\mathbf{FQ}\} \mathbf{P}\{\mathbf{X}\} \mathbf{P}\{\mathbf{FQ}\} + \mathbf{P}\{\mathbf{X}\} \mathbf{P}\{\mathbf{FQ}\} \mathbf{F}^{-1} \right]_{ii} \quad (\text{A.6})$$

Given $\mathbf{F}^{-1} \mathbf{P}\{\mathbf{FQ}\} \mathbf{P}\{\mathbf{X}\}$ is symmetric, we have

$$\frac{\partial \nu(\mathbf{FQ}, \mathbf{X})}{\partial \mathbf{F}_{ii}} = \frac{2}{3} \left[\mathbf{F}^{-1} \mathbf{P}\{\mathbf{FQ}\} \mathbf{P}\{\mathbf{X}\} - \mathbf{F}^{-1} \mathbf{P}\{\mathbf{FQ}\} \mathbf{P}\{\mathbf{X}\} \mathbf{P}\{\mathbf{FQ}\} \right]_{ii} \quad (\text{A.7})$$

This equation can be further merged into

$$\frac{\partial \nu(\mathbf{FQ}, \mathbf{X})}{\partial \mathbf{F}_{ii}} = \frac{2}{3} \left[\mathbf{F}^{-1} \mathbf{P}\{\mathbf{FQ}\} \mathbf{P}\{\mathbf{X}\} (\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\}) \right]_{ii} \quad (\text{A.8})$$

where \mathbf{I} is the identity matrix.

Now, let us rewrite the derivative in terms of the underlying filter vector \mathbf{f} . First, remember that $\mathbf{F} = \text{diag}(\mathbf{f})$. Let us denote the inverse operator - the one that extracts the diagonal from a square matrix and places the result in a vector - as *ediag* ('e' signifies to 'extract' the diagonal elements). Clearly, $\text{ediag}(\text{diag}(\mathbf{f})) = \mathbf{f}$. Here, $\text{diag}()$ is a forward operation turning a vector into a diagonal matrix and *ediag* is the companion reverse operator extracting the diagonal.

Now we can derive the gradient, in terms of the underlying filter vector, as

$$\nabla \nu(\mathbf{f}) = \frac{\partial \nu(\mathbf{FQ}, \mathbf{X})}{\partial \mathbf{f}} = \frac{2}{3} \text{ediag} \left(\mathbf{F}^{-1} \mathbf{P}\{\mathbf{FQ}\} \mathbf{P}\{\mathbf{X}\} (\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\}) \right) \quad (\text{A.9})$$

where the gradient with respect to the filter vector, $\nabla \nu(\mathbf{f})$, is a $n \times 1$ vector (in our case, $n = 31$). \square

A.2.2 Relation of ALS and Gradient Algorithms

In the gradient algorithm, the filter is updated in Step 5 of Algorithm 4.1 at each iteration as

$$\mathbf{f}^{k+1} := \mathbf{f}^k + t^k \nabla \nu(\mathbf{f}^k) \quad (\text{A.10})$$

where t^k is the step size at k th iteration. By substituting the gradient function given in (A.9), we have

$$\mathbf{f}^{k+1} := \mathbf{f}^k - t^k \text{diag} \left((\mathbf{F}^k)^{-1} \mathbf{P}\{\mathbf{F}^k \mathbf{Q}\} \mathbf{P}\{\mathbf{X}\} (\mathbf{I} - \mathbf{P}\{\mathbf{F}^k \mathbf{Q}\}) \right) \quad (\text{A.11})$$

Remember that $\mathbf{F}^k = \text{diag}(\mathbf{f}^k)$.

Now let's look at how the filter is updated at each iteration in the Alternating Least-Squares (ALS) algorithm. As shown in Algorithm 3.1, we first calculate the correction matrix \mathbf{M} given an initial filter guess and in turn solves for the filter using this newly solved \mathbf{M} .

Explicitly, in the $k + 1$ th iteration, the correction matrix \mathbf{M} is calculated using the Moore-Penrose inverse solution to the least-squares regression

$$\mathbf{M}^{k+1} = \left((\mathbf{F}^k \mathbf{Q})^T \mathbf{F}^k \mathbf{Q} \right)^{-1} (\mathbf{F}^k \mathbf{Q})^T X \quad (\text{A.12})$$

By holding \mathbf{M}^k , we can solve for a new filter solution as

$$\begin{aligned} \mathbf{f}^{k+1} &:= \frac{\mathbf{P}\{\mathbf{F}^k \mathbf{Q}\} \mathbf{P}\{\mathbf{X}\}}{\mathbf{P}\{\mathbf{F}^k \mathbf{Q}\} \mathbf{P}\{\mathbf{X}\} \mathbf{P}\{\mathbf{F}^k \mathbf{Q}\}} \mathbf{f}^k \\ &:= \mathbf{f}^k - \frac{(\mathbf{F}^k)^2}{\mathbf{P}\{\mathbf{F}^k \mathbf{Q}\} \mathbf{P}\{\mathbf{X}\} \mathbf{P}\{\mathbf{F}^k \mathbf{Q}\}} \text{diag} \left((\mathbf{F}^k)^{-1} \mathbf{P}\{\mathbf{F}^k \mathbf{Q}\} \mathbf{P}\{\mathbf{X}\} (\mathbf{I} - \mathbf{P}\{\mathbf{F}^k \mathbf{Q}\}) \right) \end{aligned} \quad (\text{A.13})$$

Comparing the calculation formula in (A.11) and (A.13), we can see that these two algorithms update the filter matrix/vector in a similar way. The difference is the step size for updating the filter: it is a constant stepsize t^k in the Gradient algorithm; however, in the ALS algorithm, the step size is not a constant but changes as a function of the previous filter refinement.

A.2.3 Derivation of Hessian Matrix

Here we present how we derive the second derivative, i.e. the Hessian matrix, of our objective function: the filter-modified Vora-Value optimisation.

Theorem A.2. *The Hessian matrix of the filter-modified Vora-Value optimisation:*

$$\begin{aligned} \nabla^2 \nu(\mathbf{f}) = & -2 \left(\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) \circ \left((\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\})\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) + \\ & \left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) \circ \left((\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\})\mathbf{P}\{\mathbf{V}\} \right) + \\ & \left((\mathbf{I} - 2\mathbf{P}\{\mathbf{FQ}\})\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) \circ \left(\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) + \\ & \left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \right) \circ \mathbf{I} \end{aligned} \quad (\text{A.14})$$

Proof. Starting from (A.9), the Hessian matrix makes a further derivative of its first derivative (the second derivative). As the matrix variable \mathbf{F} appears multiple places in the equation, to ease the notation, we use \mathbf{A} , \mathbf{B} and \mathbf{C} to respectively denote $\mathbf{P}\{\mathbf{X}\}$, $\mathbf{P}\{\mathbf{FQ}\}$, \mathbf{F}^{-1} hereafter in this subsection.

Now we differentiate for the second derivative by applying the rules of matrix calculus as given in (A.2)

$$\begin{aligned} d^2 \nu = & \frac{2}{3} \text{trace}(-2\mathbf{CB} d\mathbf{F} \mathbf{CBA} d\mathbf{F} + 2\mathbf{CB} d\mathbf{F} \mathbf{CBAB} d\mathbf{F} + \\ & \mathbf{CBC} d\mathbf{F} \mathbf{A} d\mathbf{F} - \mathbf{CBC} d\mathbf{F} \mathbf{AB} d\mathbf{F} - \\ & \mathbf{CBA} d\mathbf{F} \mathbf{CB} d\mathbf{F} + 2\mathbf{CBAB} d\mathbf{F} \mathbf{CB} d\mathbf{F} - \\ & \mathbf{CBABC} d\mathbf{F} d\mathbf{F}) \end{aligned} \quad (\text{A.15})$$

After further merging between two terms in each line (where *trace* is a linear mapping and the distributive property holds), we obtain

$$\begin{aligned} d^2 \nu = & \frac{2}{3} \text{trace} \left(-2\mathbf{CB} d\mathbf{F} \mathbf{CBA}(\mathbf{I} - \mathbf{B}) d\mathbf{F} + \mathbf{CBC} d\mathbf{F} \mathbf{A}(\mathbf{I} - \mathbf{B}) d\mathbf{F} - \right. \\ & \left. \mathbf{CBA}(\mathbf{I} - 2\mathbf{B}) d\mathbf{F} \mathbf{CB} d\mathbf{F} - \mathbf{CBABC} d\mathbf{F} d\mathbf{F} \right) \end{aligned} \quad (\text{A.16})$$

where \mathbf{I} denotes the 31×31 identity matrix.

Given $\mathbf{f} = \text{diag}(\mathbf{F})$ and any two square matrices \mathbf{M} and \mathbf{N} , we have $\text{trace}(\mathbf{M}^T d\mathbf{F} \mathbf{N} d\mathbf{F}) = \sum_i \sum_j \mathbf{M}_{i,j} \mathbf{N}_{j,i} d\mathbf{f}_i d\mathbf{f}_j$ where elements having the same indices in two matrices are multiplied. Using this property into (A.16) and the symmetric property of matrices \mathbf{A} , \mathbf{B} , \mathbf{C} (as projector matrices and the diagonal filter matrix \mathbf{F} are symmetric), we can derive the Hessian matrix as

$$\begin{aligned} d^2 \nu(\mathbf{f}) = & -2 \left(\mathbf{BC} \right) \circ \left((\mathbf{I} - \mathbf{B})\mathbf{ABC} \right) + \left(\mathbf{CBC} \right) \circ \left((\mathbf{I} - \mathbf{B})\mathbf{A} \right) + \\ & \left((\mathbf{I} - 2\mathbf{B})\mathbf{ABC} \right) \circ \left(\mathbf{BC} \right) - \left(\mathbf{CBABC} \right) \circ \mathbf{I} \end{aligned} \quad (\text{A.17})$$

where \circ denotes the Hadamard product (or elementwise product) of two matrices, i.e. $(\mathbf{M} \circ \mathbf{N})_{i,j} = \mathbf{M}_{i,j} \mathbf{N}_{i,j}$. Here you may notice that we did not add the $\frac{2}{3}$ so as to ease the notation as a scalar does not affect the optimisation.

The explicit expansion of the equation over projector matrices are written as

$$\begin{aligned} d^2\nu(\mathbf{f}) = & -2\left(\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) \circ \left(\left(\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\}\right)\mathbf{P}\{\mathbf{X}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) + \\ & \left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) \circ \left(\left(\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\}\right)\mathbf{P}\{\mathbf{X}\}\right) + \\ & \left(\mathbf{I} - 2\mathbf{P}\{\mathbf{FQ}\}\right)\mathbf{P}\{\mathbf{X}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1} \circ \left(\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) + \\ & \left(\mathbf{F}^{-1}\mathbf{P}\{\mathbf{FQ}\}\mathbf{P}\{\mathbf{X}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}\right) \circ \mathbf{I} \end{aligned} \quad (\text{A.18})$$

□

A.2.4 Positive Definite of the Hessian Matrix

As the inverse of Hessian is essential in driving the Newton's method, it is necessary that the Hessian matrix has full rank. Here we prove that the Hessian matrix of the modified Luther optimisation with a positive regulariser is to be positive definite.

We start from the gradient function of the Vora-Value based objective function (when we discount the regulariser): the gradient is the product of three projector matrices and the inverse of the filter matrix. The special property can be revealed if we multiply \mathbf{f}^T to it and we find

$$\mathbf{f}^T \nabla\nu(\mathbf{f}) = \mathbf{f}^T \text{diag}((\mathbf{I} - \mathbf{P}\{\mathbf{FQ}\})\mathbf{P}\{\mathbf{V}\}\mathbf{P}\{\mathbf{FQ}\}\mathbf{F}^{-1}) = 0 \quad (\text{A.19})$$

That is, the inner product of the filter vector, \mathbf{f} , and the gradient, $\nabla\nu(\mathbf{f})$, are perpendicular.

By using this property for the regularised formula that $\mathbf{f}^T \nabla\nu(\mathbf{f}) = 0$, we have

$$\begin{aligned} \mathbf{f}^T \nabla\mu(\mathbf{f}) &= \mathbf{f}^T \left(-3\nabla\nu(\mathbf{f}) + 2\alpha \mathbf{f} \right) \\ &= 2\alpha \mathbf{f}^T \mathbf{f} \end{aligned} \quad (\text{A.20})$$

Further, if we make the derivative to the both sides of (A.20) with respect to \mathbf{f} , we obtain

$$\nabla\mu + (\nabla^2\mu) \mathbf{f} = 4\alpha \mathbf{f}. \quad (\text{A.21})$$

If we multiply \mathbf{f}^T to this equation, we have $\mathbf{f}^T \nabla \mu + \mathbf{f}^T (\nabla^2 \mu) \mathbf{f} = 2\alpha \mathbf{f}^T \mathbf{f}$. From (A.20), we know $\mathbf{f}^T \nabla \mu(\mathbf{f}) = \alpha \mathbf{f}^T \mathbf{f}$. Hence, we get

$$\mathbf{f}^T (\nabla^2 \mu) \mathbf{f} = \alpha \mathbf{f}^T \mathbf{f} > 0, \quad \text{if } \alpha > 0 \quad (\text{A.22})$$

which guarantees the Hessian to be positive-definite under a positive α (and of course, a physically plausible filter is a non-zero vector, $\mathbf{f} > 0$).

A.3 Algorithm for the Data-driven Optimisation

In Step 4 of the ALS Algorithm 5.1 for the Data-driven optimisation, each \mathbf{M}_j^i can be solved directly using the Moore-Penrose inverse. Denoting $\mathcal{Q} = \mathbf{C}_j^T \mathbf{Q}_j^{i-1}$ then $\mathbf{M}_j^i = \mathcal{Q}^+ \mathbf{C}_k^T \mathbf{X} = [\mathcal{Q}^T \mathcal{Q}]^{-1} \mathcal{Q}^T \mathbf{C}_k^T \mathbf{X}$.

In Step 5 of Algorithm 5.1, the filter \mathbf{f} is embedded in the equation and so we cannot solve for it directly as we could for the Luther-condition case. To solve for the filter, it is useful to vectorise the minimisation. We recapitulate the minimisation statement of Step 5:

$$\min_{\mathbf{F}} \sum_{j=1}^{cnt} \| (\mathbf{C}_j^T \mathbf{F}) \mathbf{Q}_j \mathbf{M}_j - \mathbf{C}_k^T \mathbf{X} \|_F^2 \quad (\text{A.23})$$

This Frobenius norm $\| \cdot \|_F^2$ is generally applied to matrices as here but can equally be applied to vectors. The operator $vec(\cdot)$ stacks the columns of a matrix on top of each other. We rewrite (A.23) by using the vectorisation as:

$$\min_{\mathbf{F}} \sum_{j=1}^{cnt} \| vec(\mathbf{C}_j^T \mathbf{F} \mathbf{Q}_j \mathbf{M}_j) - vec(\mathbf{C}_k^T \mathbf{X}) \|_F^2 \quad (\text{A.24})$$

Now let us rewrite the diagonal filter matrix \mathbf{F} in the following way

$$\begin{aligned} \mathbf{F} &= f_1 \begin{pmatrix} 1 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 0 \end{pmatrix} + f_2 \begin{pmatrix} 0 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 0 \end{pmatrix} + \dots + f_{31} \begin{pmatrix} 0 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix} \\ &= f_1 \mathbf{D}_1 + f_2 \mathbf{D}_2 + \dots + f_3 \mathbf{D}_{31} \end{aligned} \quad (\text{A.25})$$

where the filter matrix is represented as a summation of each value in the diagonal value, f_i , multiplied with a single entry matrix \mathbf{D}_i as $\mathbf{F} = \sum_{i=1}^{31} f_i \mathbf{D}_i$. Here, \mathbf{D}_i is a 31×31 matrix with a single non-zero entry at $\mathbf{D}(i, i) = 1$.

By substituting this new filter representation into the first term of the minimisation in (A.24), we obtain

$$\text{vec}(\mathbf{C}_j^T \mathbf{F} \mathbf{Q}_j \mathbf{M}_j) = \sum_{i=1}^{31} f_i \text{vec}(\mathbf{C}_j^T \mathbf{D}_i \mathbf{Q}_j \mathbf{M}_j) \quad (\text{A.26})$$

Now let us denote a matrix $\mathbf{V}_j = [\mathbf{v}_{1,j} \mathbf{v}_{2,j} \cdots \mathbf{v}_{31,j}]$ where its column represents $\mathbf{v}_{i,j} = \text{vec}(\mathbf{C}_j^T \mathbf{D}_i \mathbf{Q}_j \mathbf{M}_j)$, Equation (A.26) can be expressed more compactly as

$$\text{vec}(\mathbf{C}_j^T \mathbf{F} \mathbf{Q}_j \mathbf{M}_j) = \mathbf{V}_j \mathbf{f} \quad (\text{A.27})$$

where $\mathbf{f} = [f_1 \ f_2 \ \dots \ f_{31}]^T$. Note that if \mathbf{C}_j is a $31 \times n$ matrix, then $\mathbf{C}_j^T \mathbf{D}_i \mathbf{Q}_j \mathbf{M}_j$ is an $n \times 3$ matrix (where 3 denotes the number of colour channels) and thus $\mathbf{v}_{i,j}$ is $3n \times 1$ which makes matrix \mathbf{V}_j have size of $3n \times 31$.

Now we stack all \mathbf{V}_j , $j = 1, 2, \dots, cnt$ matrices (under cnt different lighting conditions) on top of each other making an $(3n * cnt) \times 31$ matrix, \mathbf{V} . Similarly, we stack all cnt targeting XYZs on top of each other denoted as $\mathbf{w} = \text{vec}(\mathbf{C}_k^T \mathbf{X})$ which has the size of $(3n * cnt) \times 1$. We remind the reader that k might equal j . Or, k might denote a single privileged illuminant such as CIE D65.

Now the minimisation in (A.24) can be equivalently rewritten as:

$$\min_{\mathbf{f}} \|\mathbf{V}\mathbf{f} - \mathbf{w}\|_F^2 \quad (\text{A.28})$$

The best \mathbf{f} can be found in closed form using the Moore-Penrose inverse:

$$\mathbf{f} = [\mathbf{V}^T \mathbf{V}]^{-1} \mathbf{V}^T \mathbf{w}. \quad (\text{A.29})$$

A.4 Algorithm for the Filter Constraints

Equation (A.29) solves for the 31-component \mathbf{f} in one step. Suppose we write $\mathbf{f} = \mathbf{B}\mathbf{c}$. We constrain the filter to be describable by a linear basis (\mathbf{B} is $31 \times m$ where $1 \leq m \leq 31$). Additionally, the filter is restrained by a minimum f_{min} and maximum f_{max} bounds on the transmittance. Then to solve for the filter we find the coefficient vector \mathbf{c} that minimises:

$$\min_{\mathbf{c}} \|\mathbf{V}\mathbf{B}\mathbf{c} - \mathbf{w}\|_F^2 \quad \text{s.t.} \quad f_{min} \preceq \mathbf{B}\mathbf{c} \preceq f_{max} \quad (\text{A.30})$$

By expanding the Frobenius norm in (A.30), we obtain

$$\min_{\mathbf{c}} \mathbf{c}^T \mathbf{B}^T \mathbf{V}^T \mathbf{V} \mathbf{B} \mathbf{c} - 2\mathbf{w}^T \mathbf{V} \mathbf{B} \mathbf{c} \quad \text{s.t. } f_{min} \preceq \mathbf{B} \mathbf{c} \preceq f_{max} \quad (\text{A.31})$$

In (A.31), we actually have two inequalities, i.e. $\mathbf{B} \mathbf{c} \succeq f_{min}$ and $\mathbf{B} \mathbf{c} \preceq f_{max}$. In order to convert (A.31) into the standard quadratic problem formulation [49] (only one inequality equation is allowed), we make a slight modification of the inequality constraints:

$$\begin{bmatrix} \mathbf{B} \\ -\mathbf{B} \end{bmatrix} \mathbf{c} \preceq \begin{bmatrix} \mathbf{f}_{max} \\ -\mathbf{f}_{min} \end{bmatrix} \quad (\text{A.32})$$

Now we can use the Quadratic Programming function to solve for the the filter coefficient vector \mathbf{c} . Finally, the filter solution can be calculated as $\mathbf{f} = \mathbf{B} \mathbf{c}$.

Bibliography

- [1] Synthetic data for computational colour constancy experiments. https://www2.cs.sfu.ca/~colour/data/colour_constancy_synthetic_test_data/ [Accessed: 01 Feb 2018].
- [2] ANDERSEN, C. F., AND CONNAH, D. Weighted constrained hue-plane preserving camera characterization. *IEEE Transactions on Image Processing* 25, 9 (2016), 4329–4339.
- [3] ANDERSON, C. F., AND HARDEBERG, J. Y. Colorimetric characterization of digital cameras preserving hue planes. In *Color and Imaging Conference* (2005), Society for Imaging Science and Technology, pp. 141–146.
- [4] ARAD, B., AND BEN-SHAHAR, O. Filter selection for hyperspectral estimation. In *International Conference on Computer Vision* (2017), IEEE, pp. 3172–3180.
- [5] ASANO, Y., FAIRCHILD, M. D., BLONDÉ, L., AND MORVAN, P. Color matching experiment for highlighting interobserver variability. *Color Research & Application* 41, 5 (2016), 530–539.
- [6] BARNARD, K., MARTIN, L., FUNT, B., AND COATH, A. A data set for color research. *Color Research & Application* 27, 3 (2002), 147–151.
- [7] BARNHÖFER, U., DICARLO, J. M., OLDING, B. P., AND WANDELL, B. A. Color estimation error trade-offs. In *Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications IV* (2003), vol. 5017, SPIE, pp. 263–273.
- [8] BERNS, R. S. *Billmeyer and Saltzman’s Principles of Color Technology*. John Wiley & Sons, 2019.
- [9] BERTSEKAS, D. P. Projected Newton methods for optimization problems with simple constraints. *SIAM Journal on Control and Optimization* 20, 2 (1982), 221–246.
- [10] BOYCE, P. R. *Human Factors in Lighting*, 2 ed. Talor & Francis, 2003.

-
- [11] BOYD, S., AND VANDENBERGHE, L. *Convex Optimization*. Cambridge University Press, 2004.
- [12] CALAMAI, P. H., AND MORÉ, J. J. Projected gradient methods for linearly constrained problems. *Mathematical programming* 39, 1 (1987), 93–116.
- [13] CHEUNG, V., WESTLAND, S., CONNAH, D., AND RIPAMONTI, C. A comparative study of the characterisation of colour cameras by means of neural networks and polynomial transforms. *Coloration Technology* 120, 1 (2004), 19–25.
- [14] CHIAO, C.-C., CRONIN, T. W., AND OSORIO, D. Color signals in natural scenes: characteristics of reflectance spectra and effects of natural illuminants. *Journal of the Optical Society of America A* 17, 2 (2000), 218–224.
- [15] COHEN, J. Dependency of the spectral reflectance curves of the Munsell color chips. *Psychonomic Science* 1, 12 (1964), 369–370.
- [16] COHEN, J. B., AND KAPPAUF, W. E. Metameric color stimuli, fundamental metamers, and Wyszecki’s metameric blacks. *The American Journal of Psychology* (1982), 537–564.
- [17] COHEN, J. B., AND KAPPAUF, W. E. Color mixture and fundamental metamers: Theory, algebra, geometry, application. *The American Journal of Psychology* (1985), 171–259.
- [18] CONNAH, D., WESTLAND, S., AND THOMSON, M. Recovering spectral information using digital camera systems. *Coloration Technology* 117, 6 (2001), 309–312.
- [19] DAVIES, W., AND WYSZECKI, G. Physical approximation of color-mixture functions. *Journal of the Optical Society of America* 52, 6 (1962), 679–685.
- [20] DEMARCO, P., POKORNY, J., AND SMITH, V. C. Full-spectrum cone sensitivity functions for X-chromosome-linked anomalous trichromats. *Journal of the Optical Society of America A* 9, 9 (1992), 1465–1476.
- [21] DREW, M. S., AND FUNT, B. V. Natural metamers. *CVGIP: Image Understanding* 56, 2 (1992), 139–151.
- [22] ENGELHARDT, K., AND SEITZ, P. Optimum color filters for CCD digital cameras. *Applied Optics* 32, 16 (1993), 3015–3023.
- [23] FAIRMAN, H. S., AND BRILL, M. H. The principal components of reflectances. *Color Research & Application* 29, 2 (2004), 104–110.

-
- [24] FAIRMAN, H. S., BRILL, M. H., AND HEMMENDINGER, H. How the CIE 1931 color-matching functions were derived from Wright-Guild data. *Color Research & Application* 22, 1 (1997), 11–23.
- [25] FARRELL, J. E., AND WANDELL, B. A. Method and apparatus for identifying the color of an image, December 26, 1995. U.S. Patent 5479524.
- [26] FINLAYSON, G. D., DARRODI, M. M., AND MACKIEWICZ, M. The alternating least squares technique for nonuniform intensity color correction. *Color Research & Application* 40, 3 (2015), 232–242.
- [27] FINLAYSON, G. D., AND DREW, M. S. The maximum ignorance assumption with positivity. In *Color and Imaging Conference* (1996), Society for Imaging Science and Technology, pp. 202–205.
- [28] FINLAYSON, G. D., AND DREW, M. S. Constrained least-squares regression in color spaces. *Journal of Electronic Imaging* 6, 4 (1997), 484–494.
- [29] FINLAYSON, G. D., MACKIEWICZ, M., AND HURLBERT, A. Color correction using root-polynomial regression. *IEEE Transactions on Image Processing* 24, 5 (2015), 1460–1470.
- [30] FOSSUM, E. R. CMOS image sensors: Electronic camera-on-a-chip. *IEEE Transactions on Electron Devices* 44, 10 (1997), 1689–1698.
- [31] GOLUB, G. H., AND VAN LOAN, C. F. *Matrix Computations*, 3 ed. Johns Hopkins University Press, 1996.
- [32] GRASSMANN, H. Zur theorie der farbenmischung. *Annalen der Physik und Chemie* 165, 5 (1853), 69–84.
- [33] GUILD, J. The colorimetric properties of the spectrum. *Philosophical Transactions of the Royal Society of London* 230, 681-693 (1931), 149–187.
- [34] HARDEBERG, J. Y. *Acquisition and reproduction of color images: colorimetric and multispectral approaches*. Universal-Publishers, 2001.
- [35] HONG, G., LUO, M. R., AND RHODES, P. A. A study of digital camera colorimetric characterization based on polynomial modeling. *Color Research & Application* 26, 1 (2001), 76–84.
- [36] HORN, B. K. P. Exact reproduction of colored images. *Computer Vision, Graphics, and Image Processing* 26, 2 (1984), 135 – 167.

- [37] HUNG, P.-C. Colorimetric calibration in electronic imaging devices using a look-up-table model and interpolations. *Journal of Electronic Imaging* 2, 1 (1993), 53–62.
- [38] HUNT, R. W. G., AND POINTER, M. R. *Measuring Colour*, 4 ed. John Wiley & Sons, 2011.
- [39] IMAI, F. H., QUAN, S., ROSEN, M. R., AND BERNS, R. S. Digital camera filter design for colorimetric and spectral accuracy. In *International Conference on Multispectral Color Science* (2001), pp. 13–16.
- [40] IVES, H. E. The transformation of color-mixture equations from one system to another. *Journal of the Franklin Institute* 180, 6 (1915), 673–701.
- [41] JIANG, J., LIU, D., GU, J., AND SÜSTRUNK, S. What is the space of spectral sensitivity functions for digital color cameras? In *Workshop on Applications of Computer Vision* (2013), IEEE, pp. 168–179.
- [42] JUDD, D. B., MACADAM, D. L., WYSZECKI, G., BUDDE, H., CONDIT, H., HENDERSON, S., AND SIMONDS, J. Spectral distribution of typical daylight as a function of correlated color temperature. *Journal of the Optical Society of America* 54, 8 (1964), 1031–1040.
- [43] KLINKER, G. J., SHAFER, S. A., AND KANADE, T. A physical approach to color image understanding. *International Journal of Computer Vision* 4, 1 (1990), 7–38.
- [44] KOLLARITS, R. V., AND GIBBON, D. C. Improving the color fidelity of cameras for advanced television systems. In *High-Resolution Sensors and Hybrid Systems* (1992), vol. 1656, SPIE, pp. 19–29.
- [45] KRANTZ, D. H. Color measurement and color theory: I. representation theorem for Grassmann structures. *Journal of Mathematical Psychology* 12, 3 (1975), 283–303.
- [46] KRINOV, E. L. Spectral reflectance properties of natural formations. Tech. rep., National Research Council of Canada, 1947.
- [47] LI, J., SKINNER, K. A., EUSTICE, R. M., AND JOHNSON-ROBERSON, M. Waternan: Unsupervised generative network to enable real-time color correction of monocular underwater images. *IEEE Robotics and Automation Letters* 3, 1 (2017), 387–394.
- [48] LIM, S., AND SILVERSTEIN, A. Spatially varying colour correction matrices for reduced noise. In *Color and Imaging Conference* (2004), Society for Imaging Science and Technology, pp. 105–111.

- [49] LUENBERGER, D. G., AND YE, Y. *Linear and Nonlinear Programming*, 4 ed. Springer, 2015.
- [50] LUO, M. R., CUI, G., AND RIGG, B. The development of the CIE 2000 colour-difference formula: CIEDE2000. *Color Research & Application* 26, 5 (2001), 340–350.
- [51] LUO, M. R., AND RIGG, B. BFD (l: c) colour-difference formula part I - development of the formula. *Journal of the Society of Dyers and Colourists* 103, 2 (1987), 86–94.
- [52] LUTHER, R. Aus dem gebiet der farbreizmetrik. *Zeitschrift Technische Physik* 8 (1927), 540–558.
- [53] MACADAM, D. L. Colorimetric specifications of Wratten light filters. *Journal of the Optical Society of America* 35, 10 (1945), 670–675.
- [54] MACADAM, D. L. Orthogonal color-mixture functions. *Journal of the Optical Society of America A* 44, 9 (1954), 713–724.
- [55] MACKIEWICZ, M., ANDERSEN, C. F., AND FINLAYSON, G. D. Method for hue plane preserving color correction. *Journal of the Optical Society of America A* 33, 11 (2016), 2166–2177.
- [56] MALONEY, L. T. Evaluation of linear models of surface spectral reflectance with small numbers of parameters. *Journal of the Optical Society of America A* 3, 10 (1986), 1673–1683.
- [57] MARIMONT, D. H., AND WANDELL, B. A. Linear models of surface and illuminant spectra. *Journal of the Optical Society of America A* 9, 11 (1992), 1905–1913.
- [58] MARTÍNEZ-DOMINGO, M. Á., MELGOSA, M., OKAJIMA, K., MEDINA, V. J., AND COLLADO-MONTERO, F. J. Spectral image processing for museum lighting using CIE LED illuminants. *Sensors* 19, 24 (2019), 5400.
- [59] MCCAMY, C. S., MARCUS, H., AND DAVIDSON, J. G. A color-rendition chart. *Journal of Applied Photographic Engineering* 2, 3 (1976), 95–99.
- [60] MONNO, Y., KIKUCHI, S., TANAKA, M., AND OKUTOMI, M. A practical one-shot multispectral imaging system using a single image sensor. *IEEE Transactions on Image Processing* 24, 10 (2015), 3048–3059.
- [61] NAKAMURA, J. *Image Sensors and Signal Processing for Digital Still Cameras*. CRC Press, 2016.

- [62] NEUGEBAUER, H. Quality factor for filters whose spectral transmittances are different from color mixture curves, and its application to color photography. *Journal of the Optical Society of America* 46, 10 (1956), 821–824.
- [63] NIEVES, J. L., VALERO, E. M., NASCIMENTO, S. M., HERNÁNDEZ-ANDRÉS, J., AND ROMERO, J. Multispectral synthesis of daylight using a commercial digital CCD camera. *Applied Optics* 44, 27 (2005), 5696–5703.
- [64] NOCEDAL, J., AND WRIGHT, S. *Numerical Optimization*, 2 ed. Springer, 2006.
- [65] OHTA, N., AND ROBERTSON, A. *Colorimetry: Fundamentals and Applications*. John Wiley & Sons, 2006.
- [66] OREN, M., AND NAYAR, S. K. Generalization of Lambert’s reflectance model. In *Annual Conference on Computer Graphics and Interactive Techniques* (1994), pp. 239–246.
- [67] PARKKINEN, J. P. S., HALLIKAINEN, J., AND JAASKELAINEN, T. Characteristic spectra of munsell colors. *Journal of the Optical Society of America A* 6, 2 (1989), 318–322.
- [68] PENROSE, R. A generalized inverse for matrices. In *Mathematical Proceedings of the Cambridge Philosophical Society* (1955), vol. 51, Cambridge University Press, pp. 406–413.
- [69] PIERI, E., AND PYTLARZ, J. Hitting the mark—a new color difference metric for HDR and WCG imagery. *SMPTE Motion Imaging Journal* 127, 3 (2018), 18–25.
- [70] QUAN, S. *Evaluation and optimal design of spectral sensitivities for digital color imaging*. PhD thesis, Rochester Institute of Technology, 2002.
- [71] QUAN, S., OHTA, N., BERNS, R. S., JIANG, X., AND KATOH, N. Unified measure of goodness and optimal design of spectral. *Journal of Imaging Science and Technology* 46, 6 (2002), 485–497.
- [72] SHARMA, G., AND BALA, R. *Digital Color Imaging Handbook*. CRC Press, 2002.
- [73] SHARMA, G., AND TRUSSELL, H. J. Digital color imaging. *IEEE Transactions on Image Processing* 6, 7 (1997), 901–932.
- [74] SHARMA, G., AND TRUSSELL, H. J. Figures of merit for color scanners. *IEEE Transactions on Image Processing* 6, 7 (1997), 990–1001.
- [75] SHARMA, O., TRUSSELL, H. J., AND VRHEL, M. J. Optimal nonnegative color scanning filters. *IEEE Transactions on Image Processing* 7, 1 (1998), 129–133.

- [76] SMITH, B., SPIEKERMANN, C., AND SEMBER, R. Numerical methods for colorimetric calculations: sampling density requirements. *Color Research & Application* 17, 6 (1992), 394–401.
- [77] SPERANSKAYA, N. I. Determination of spectrum color coordinates for twenty-seven normal observers. *Optics and Spectroscopy* 7 (1959), 424–428.
- [78] STILES, W. S., AND BURCH, J. M. NPL colour-matching investigation: final report. *Optica Acta: International Journal of Optics* 6, 1 (1959), 1–26.
- [79] STOCKMAN, A., MACLEOD, D. I., AND JOHNSON, N. E. Spectral sensitivities of the human cones. *Journal of the Optical Society of America A* 10, 12 (1993), 2491–2521.
- [80] STOCKMAN, A., AND SHARPE, L. T. The spectral sensitivities of the middle- and long-wavelength-sensitive cones derived from measurements in observers of known genotype. *Vision Research* 40, 13 (2000), 1711–1737.
- [81] STRANG, G. The discrete cosine transform. *SIAM Review* 41, 1 (1999), 135–147.
- [82] TRUSSELL, H. J., ERCAN, A. O., AND KINGSBURY, N. G. Color filters: When “optimal” is not optimal. In *International Conference on Image Processing* (2016), IEEE, pp. 3987–3991.
- [83] TYNDALL, E. P. T., AND GIBSON, K. S. Visibility of radiant energy. *Journal of the Optical Society of America* 9, 4 (1924), 403–403.
- [84] VALERO, E. M., NIEVES, J. L., NASCIMENTO, S. M., AMANO, K., AND FOSTER, D. H. Recovering spectral data from natural scenes with an RGB digital camera and colored filters. *Color Research & Application* 32, 5 (2007), 352–360.
- [85] VAZQUEZ-CORRAL, J., CONNAH, D., AND BERTALMÍO, M. Perceptual color characterization of cameras. *Sensors* 14, 12 (2014), 23205–23229.
- [86] VORA, P., AND HERLEY, C. Trade-offs between color saturation and noise sensitivity in image sensors. In *International Conference on Image Processing* (1998), IEEE, pp. 196–200.
- [87] VORA, P. L. Inner products and orthogonality in color recording filter design. *IEEE Transactions on Image Processing* 10, 4 (2001), 632–642.
- [88] VORA, P. L., AND TRUSSELL, H. J. Measure of goodness of a set of color-scanning filters. *Journal of the Optical Society of America A* 10, 7 (1993), 1499–1508.

- [89] VORA, P. L., AND TRUSSELL, H. J. Mathematical methods for the analysis of color scanning filters. *IEEE Transactions on Image Processing* 6, 2 (1997), 321–327.
- [90] VORA, P. L., AND TRUSSELL, H. J. Mathematical methods for the design of color scanning filters. *IEEE Transactions on Image Processing* 6, 2 (1997), 312–320.
- [91] VORA, P. L., TRUSSELL, H. J., AND IWAN, L. S. Mathematical method for designing a set of color scanning filters. In *Color Hard Copy and Graphic Arts II* (1993), vol. 1912, SPIE, pp. 322–333.
- [92] VRHEL, M. J. Improved camera color accuracy in the presence of noise with a color prefilter. In *Color and Imaging Conference* (2020), Society for Imaging Science and Technology, pp. 187–192.
- [93] VRHEL, M. J., GERSHON, R., AND IWAN, L. S. Measurement and analysis of object reflectance spectra. *Color Research & Application* 19, 1 (1994), 4–9.
- [94] VRHEL, M. J., AND TRUSSELL, H. J. Color correction using principal components. *Color Research & Application* 17, 5 (1992), 328–338.
- [95] VRHEL, M. J., AND TRUSSELL, H. J. Filter considerations in color correction. *IEEE Transactions on Image Processing* 3, 2 (1994), 147–161.
- [96] VRHEL, M. J., AND TRUSSELL, H. J. Optimal color filters in the presence of noise. *IEEE Transactions on Image Processing* 4, 6 (1995), 814–823.
- [97] WANDELL, B. A. *Foundations of Vision*. Sinauer Associates, 1995.
- [98] WANDELL, B. A., AND FARRELL, J. E. Water into wine: Converting scanner RGB to tristimulus XYZ. In *Device-Independent Color Imaging and Imaging Systems Integration* (1993), vol. 1909, SPIE, pp. 92–101.
- [99] WESTLAND, S., SHAW, J., AND OWENS, H. Colour statistics of natural and man-made surfaces. *Sensor Review* 20, 1 (2000), 50–55.
- [100] WOLSKI, M., BOUMAN, C., ALLEBACH, J. P., AND WALOWIT, E. Optimization of sensor response functions for colorimetry of reflective and emissive objects. *IEEE Transactions on Image Processing* 5, 3 (1996), 507–517.
- [101] WORTHEY, J. A., AND BRILL, M. H. Camera design using locus of unit monochromats. In *Color and Imaging Conference* (2006), Society for Imaging Science and Technology, pp. 185–190.
- [102] WRIGHT, W. D. A re-determination of the trichromatic coefficients of the spectral colours. *Transactions of the Optical Society* 30 (1929), 141–164.

-
- [103] WYSZECKI, G., AND STILES, W. S. *Color Science: Concepts and Methods, Quantitative Data and Formulae*, 2 ed. Wiley, 1982.
- [104] XINWU, L. A new color correction model for based on BP neural network. *Advances in Information Sciences and Service Sciences* 3, 5 (2011).
- [105] XU, P., AND XU, H. Filter selection based on light source for multispectral imaging. *Optical Engineering* 55, 7 (2016), 074102.
- [106] ZHANG, T., AND GOLUB, G. H. Rank-one approximation to high order tensors. *Journal on Matrix Analysis and Applications* 23, 2 (2001), 534–550.
- [107] ZHU, Y., AND FINLAYSON, G. D. A mathematical investigation into the design of prefilters that make cameras more colorimetric. *Sensors* 20, 23 (2020), 6882.