Chapter 4

Continuous Color Formation Model

4.1 Introduction

The main endeavour in this Chapter is the study from a theoretical point of view of the nature of the transformations which drive the change in the color of objects due to a variation of the lighting conditions that may exist at the moment two different images from roughly the same scene are taken. The precise knowledge of the kind of applications involved in such a process is an *a priori* stage in the treatment of any color constancy issue as will be more specifically tackled in the next Chapter.

To this purpose a framework is suggested undertaking both analytically and numerically that question as a natural generalization of the usual color formation equations. This formulation is able to describe the formation of multispectral color signals by means of a continuous expression that is discretized as needed in order to attain handier computational schemes. Such a reformulation of the color formation equations presents some advantages, e.g., sounder mathematical foundations, independence from the particular discretization employed, along with that of encompassing several other previous well–known approaches that have been suggested physically modeling the color phenomenon.

The continuous color formation equation turns out to be identified as a *Fredholm's Integral equation of the First Kind* (IFK), which is helpful to establish the least theoretical conditions for the solution to exist, be unique, and numerically well–behaved, as it is more deeply explained in Appendix A. We also propose an analytical solution to both the spectral recovery and the color mapping computation from where more useful numerical schemes are derived. The theoretical framework gives an inner insight on what is involved in the color change problem, besides being an interesting starting point to other matters such as numerical stability or the sort of constraints that should be fulfilled in order a proper solution exists.

As said before, a numerical version of the aforementioned IFK approach is given in order to make practical computations along with the type of data commonly available in this kind of research. The continuous equation has to be discretized and this can not be done in an unkindly manner, as sometimes it is the case. In Appendix A some helpful examples of discretization methods are described. This way, the IFK model can be related with several widely known physical models of color formation.

In the Section devoted to results two problems are numerically studied, namely, the spectral recovery of reflectance functions from sensor measures and the explicit computation of color mappings by applying the previous group of physical models. Our aim is that of comparing the results obtained using the IFK model to those obtained with other alternative models to evaluate their mutual performance in the two types of experiments. Independently, a least squares fitting of the maps accounting for the color change is also undergone. The goal of this approach is to discern whether or not actual data really fulfills the linear transformations that the physical models suggest and to what extent. Results show the accuracy for both the linear hypothesis and the physical models considered so far. Accordingly to these results, some simplifications to the general linear model can be established to ease the color constancy algorithms that will be proposed in the next Chapter.

4.2 Outline of the Chapter

In Section 4.3, we briefly describe the classical color formation model which will be used as the starting point in this Chapter. In Section 4.4, we rewrite this model as an integral equation exploited in Section 4.5 to set out the theoretical basis of the color change problem. In Section 4.6, the generalized color transformation is analytically proposed in terms of the solution of an integral equation. We also study its numerical properties and stability. In Section 4.7, a discretization of the previous color transformation is described. In Section 4.8, we relate the present approach to some of the most interesting former physical models which color constancy is based on to explicitly state how the latter ones can be derived from the former. Experiments and results are described in Section 4.9. Different subsections are concerned with sensor fitting, spectral recovery, and computation of color mappings. Finally, in Section 4.10, we briefly present our conclusions.

4.3 Color Formation Model

To begin with, we need a mathematical model embodying the predominant phenomena occurring in the formation of color images. Basically, there are two main processes involved, namely, light reflection on the object's surfaces and camera measurement of the light coming out from this reflection. For the first one, it is necessary to describe the mutual interaction between light and surfaces as it is seen from a point upon the image plane. Accounting for the second issue, the way a sensor integrates the light falling onto the image plane must be established.

These two issues are concisely dealt in this Section, where a complete mathematical model is put forward describing the physical process involved in the color formation, which are common to a number of other multispectral approaches. By a natural generalization of these equations, a continuous color formation function is suggested in the next Section. This model will not be used in posterior Sections to only study the spectral recovery, but also the color constancy problem.

4.3.1 Reflection Model

The alteration followed by a light beam from its birth to its fall into the camera sensor can be described as a set of successive reflections on the surfaces of the objects in the scene. This way, light changes its wavelength composition as it touches different surfaces. To explain this apparently chaotic process, it is usually defined a reflection model tying the light arriving onto and leaving from an infinitesimal surface element. This information is encompassed by the surface spectral reflectance function.

The diversity of reflection models found in literature is huge and their review is out of the scope of our work. However, let us cite those of Lambert (1760), Torrance and Sparrow [TS67], Wolff [Wol94], Phong [Pho75], the Bidirectional Reflectance Distribution Function (BRDF), and the dichromatic model [Tom91] just as the most widespread in both computer vision and computer graphics.

Basically, all of them model the surface spectral reflectance function as a linear combination of different approximating functions each one describing a specific physical hint of the reflection phenomenon. In order to avoid being computationally expensive or excessively complex, most of the above models only take into account two aspects of the reflection phenomenon, namely, the *diffuse* reflection and the *specular* reflection.

The first one appears when a surface reflects the same proportion of incident light in all directions. This reflection component is known as the *Lambertian* component and is the basic constant component for all the surface reflectance models. The specular reflection shows up when the incident light is mainly reflected in a particular direction. In this case, the proportion of reflected light depends upon both the incident and viewing directions and upon the physical structure of the surface (texture). This component causes highlights and the glossy aspect of objects.

If we assume that there is no specular component – because it was removed or did not exist – in the surface reflectance function, i.e., a Lambertian model for the surface reflectance, then the *image irradiance* I defined as the total light focused by the optical system on the camera onto the image plane in a surface element, forming an image pixel $\mathbf{x}$, will be the quantity

$$I^{\mathbf{x}}(\lambda) = R^{\mathbf{x}}(\lambda)\, E^{\mathbf{x}}(\lambda) \tag{4.1}$$

where $R^{\mathbf{x}}(\lambda)$ is the *spectral reflectance function* of the piece of surface that optically corresponds to the pixel $\mathbf{x}$ and $E^{\mathbf{x}}(\lambda)$ is the *spectral power distribution* of the light beam falling onto $\mathbf{x}$. Both I, E, and R are all *nonnegative* functions of both the wavelength λ and the pixel $\mathbf{x}$, but they depend neither on the incident nor on the viewing directions.

In this model, the geometry of the object surface is merely *a scale factor* in function E since the surface is Lambertian. Such a factor can be removed by normalization of this function. Despite its simplicity, this surface reflectance model is pretty correct unless sever interreflection and specular reflection occur. This is the reflection model we are using throughout this work.

At last, empirical studies showed that both real illumination and reflectance are relatively smooth functions of the wavelength of light in the visual spectrum ($400 \div 700$ nm). Hence, finite–dimensional linear models were introduced to describe spectral power distributions by Buchsbaum [Buc80], who extended previous works by Helson [Hel38] and Judd [Jud40, JMW64]. Similarly, reflectance functions of a great variety of materials have also been studied. Cohen [Coh64], Parkkinen et al. [PHJ86], and Maloney [Mal86] studied the reflectance properties of the Munsell chips as well as Vrhel et al. [VGI94], Krinov [Kri47], and Wyszecki and Stiles [WS82] considered the reflectance of naturally occurring surface materials.

4.3.2 Sensor Model

A general camera can be seen as an array of p sensors $S_h(\lambda)$, where $h = 1, \ldots, p$. Usually, $p = 3$ in color cameras, but this framework is likewise valid for multispectral imaging sets, where $p > 3$. Each of these sensors, $S_h(\lambda)$, measures the light $I^{\mathbf{x}}(\lambda)$ arriving onto the image plane and giving rise to the h^{th} channel value $y_h^{\mathbf{x}}$ for the pixel $\mathbf{x}$ in the image [VFTB97a, VFTB97b, VFTB97c, KAPH94, ST97]. This is generally modeled as

$$y_h^{\mathbf{x}} = \int_{\lambda_0}^{\lambda_1} S_h(\lambda)\, I^{\mathbf{x}}(\lambda)\, d\lambda\,, \;\; h = 1, \ldots, p \tag{4.2}$$

where $[\lambda_0, \lambda_1]$ is the interval where these sensors operates ($400 \div 700$ nm).

However, accordingly to Vora et al. [VFTB97a, VFTB97b, VFTB97c], in most of real CCD cameras some other processes may occasionally take place distorting the output signal of the sensor. For example,

- Sensor response is not always linear along with the amount of energy onto it and a nonlinear function $\mathcal{F}_h$ should be taken into account to model the sensor output.

- Since cameras were made for grabbing images to be displayed in a TV set, sensor outputs are softened by an exponential factor called *gamma correction*, γ_h.

- Also, there is an amplifying factor, α_h, that accounts for the *white correction* in color cameras and other electronic signal amplifications.

- The *dark current*, β_h, may also occur. This is caused by an offset noise of sensors that detect some light while in darkness.

For the h^{th} sensor in a camera and for every pixel $\mathbf{x}$ in an image, its correspondent output signal, $y_h^{\mathbf{x}}$, is

$$y_h^{\mathbf{x}} = \alpha_h \left[\mathcal{F}_h \left(\int_{\lambda_0}^{\lambda_1} S_h(\lambda)\, I^{\mathbf{x}}(\lambda)\, d\lambda \right) \right]^{\frac{1}{\gamma_h}} + \beta_h\,, \qquad h = 1, \ldots, p \tag{4.3}$$

To go further in our color research first we should remove all these sensor nonlinearities. Vora et al. [VFTB97a, VFTB97b, VFTB97c] and Vrhel and Trussell [VT93] showed some feasible methods to calibrate color devices. Barnard [BF99] also suggested calibration as a first step in any color research. Nevertheless, former calibrations are difficult to carry out because the needed set is expensive and pretty restrictive. Alternatively, there also exists a kind of calibration called *radiometric* [DM97, MN99]. In the underlying idea of this method, the sensor response is estimated varying the *exposure* under which the picture is taken. This observation permits estimating of a *linearized* response without any prior knowledge of the scene radiance and no special device. Without going into more details on this subject, from now on we suppose we are always using *linearized* sensors following Eq. (4.2).

4.4 Continuous Color Formation Model

Hence, it has been assumed that the response of any channel for any pixel in the image plane can be expressed with a set of equations as

$$y_h = \int_{\lambda_0}^{\lambda_1} S_h(\lambda)\, E(\lambda)\, R(\lambda)\, d\lambda \,, \;\; h = 1, \ldots, p \tag{4.4}$$

where y_h is the value in the h^{th} channel at a certain pixel[1]. Thus, we define the *color vector* as the set of values $\mathbf{y} = (y_1, \ldots, y_p)$, where p is the number of channels.

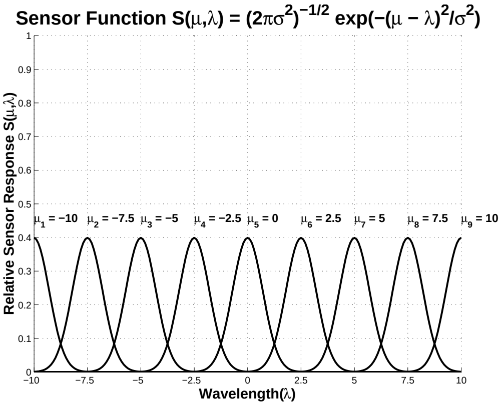


Figure 4.1: Example of an idealized sensor function.

From the sensor functions found in practice, we build up the new continuous function in a way that each of the former sensor functions can be derived as an instance centered at a particular wavelength. Let us define the *continuous sensor sensitivity function* S as a real–valued, smooth, nonnegative, bounded, and nonsingular function such that

- $S\colon [\mu_0, \mu_1] \times [\lambda_0, \lambda_1] \rightarrow [0, 1]$
- $0 \leq \int_{\mu_0}^{\mu_1}\int_{\lambda_0}^{\lambda_1} S(\mu, \lambda)\, d\mu\, d\lambda \leq 1$
- $S(\mu_h, \lambda) = S_h(\lambda)$, $\forall h = 1, \ldots, p$

[1] From now on we obviate the explicit reference to the pixel $\mathbf{x}$.

for a finite set of wavelengths μ_h. For example, an idealized sensor would be

$$S\left(\mu, \lambda\right) = k \cdot exp\left(-\frac{\left(\mu - \lambda\right)^2}{\sigma^2}\right) \tag{4.5}$$

as seen in Fig. 4.1, where some instances of that function are shown at a discrete set of wavelengths μ_h.

Then, as a consequence of the previous definition, we can also think of a continuous sensor output signal expressed as

$$\begin{array}{rcl} y\colon\ [\mu_0, \mu_1] & \longrightarrow & [0, 1] \\ \mu & \longmapsto & y(\mu) = \displaystyle\int_{\lambda_0}^{\lambda_1} S(\mu, \lambda)\, R(\lambda)\, E(\lambda)\, d\lambda \end{array} \tag{4.6}$$

The function $y(\mu)$ is the *continuous color function* whose discretization is given by a set of physical sensors. These sensor outputs have the following property

$$y(\mu_h) = y_h\,, \qquad h = 1, \ldots, p \tag{4.7}$$

so, any color output is a vector as $\mathbf{y} = (y(\mu_1), \ldots, y(\mu_p))$. This way, we have naturally generalized the equations for the formation of a color response into only one and continuous general expression, which therefore includes Eq. (4.4) as a particularization of it, despite the number and nature of the sensors involved.

It can be argued about the reasons of appending a new parameter to the original sensor model. Some reasons for such a generalization are

- It allows a more general and mathematically sounder theoretical treatment of the color formation issue.
- Other former well–known models of color formation can be derived from this continuous equation, as will be seen later.
- Any property developed from this equation is discretization–dependent, since it lives in the continuous domain, being valid for no matter the number of sensors, which should be considered prior to any particularization.
- This equation relates the color formation expressions to the set of *Fredholm's Integral equation of the First Kind* (IFK), whose properties and implications to color change will be extensively studied later on this Chapter and also in Appendix A.

Let us define the following functions, called *kernels*, in order to transform Eq. (4.6) into a proper IFK-like formulation

$$\left.\begin{array}{rcl} K_E(\mu, \lambda) & = & S(\mu, \lambda)\, E(\lambda) \\[1ex] K_R(\mu, \lambda) & = & S(\mu, \lambda)\, R(\lambda) \end{array}\right\} \tag{4.8}$$

These functions represent, in a generalized way, a *measurement process.* Depending on the problem, we select one of the two functions to write the corresponding IFK. For example, in case we want to calculate the spectral power distribution $E(\lambda)$ from the spectral reflectance function $R(\lambda)$, we should use the kernel $K_R(\mu, \lambda)$ since function E is the unknown and R is the data.

In our case, we are interested in using the $K_E(\mu,\lambda)$ kernel because any change in the illumination can be thought as a change in the measurement of the color function, i.e., a change in kernel K_E. The kernel reflects how this measure is carried out by melting down sensors and illumination, which forms a new set of virtual sensors. If the illumination conditions change, the kernel will change accordingly. This is the reason why the illumination information must be embedded in the kernel function.

Finally, for any pixel, we get an IFK equation like those described in [Pip91, Kre89, Han00, Win91] and in Appendix A, i.e.,

$$y(\mu) = \int_{\lambda_0}^{\lambda_1} K_E(\mu,\lambda)\, R(\lambda)\, d\lambda, \qquad \mu \in [\mu_0,\mu_1] \tag{4.9}$$

4.5 Generalized Color Transformation

As seen, while it is the information about a surface that is carried by the surface reflectance function $R(\lambda)$, what color cameras provide is just a measure of the color function $y(\mu)$ at a discrete set of wavelengths $\{\mu_h\}$. Therefore, whereas the reflectance function is an obvious invariant descriptor for a surface, its color is largely dependent on two factors, namely, the sensor $S(\mu,\lambda)$ and the illumination $E(\lambda)$.

In order the data provided by a camera to be useful in machine vision, it is necessary to distill a better description from the surficial properties. This problem can be faced in two different ways. We could guess to directly recover the spectral surface reflectance function, $R(\lambda)$, from the raw data provided by the camera as has been suggested in [GJT88, MW86, TW89, TW90, Fin98]. An alternative approach opts for recovering only a description for the surfaces which is constant to changes of illumination in terms of color. We theoretically consider the two options hereafter.

Spectral recovery scheme can be viewed as solving an integral equation

$$\text{finding} \qquad R\colon [\lambda_0,\lambda_1] \to [0,1] \qquad \text{so that} \qquad y(\mu) = \int_{\lambda_0}^{\lambda_1} K_E(\mu,\lambda)\, R(\lambda)\, d\lambda \tag{4.10}$$

where $\mu \in [\mu_0,\mu_1]$ and the *kernel* function is $K_E(\mu,\lambda) = S(\mu,\lambda)\, E(\lambda)$. To explicitly solve this type of equations both functions $K_E(\mu,\lambda)$ and $y(\mu)$ should be known at least at some points, along with some other mathematical and/or physical properties that could be applied to help to search for a result.

Nevertheless, as it is known from Appendix A, this equation lacks of a unique and stable solution unless some constraints on the function spaces are established. Specifically, both the *solution space* $\mathcal{R} = \{R(\lambda)\}$ and the *data space* $\mathcal{Y} = \{y(\mu)\}$ must have a finite dimension. A usual problem with this approach has to do with the difficulty of finding a realistic solution space to model the reflectance functions for everyday objects. Besides, it is pretty normal to have only a reduced number of measures from existing sensors, which may restrict too much the solutions and hinder the attaining of good approximations for the reflectance functions $R(\lambda)$.

An alternative consists in avoiding the more involved process of completely recovering the surface reflectance functions and envisaging the *computation of a* *color transformation* from a set of color data to another set the way it was introduced by Forsyth in [For90]. If we maintain the camera features constant and only the scene illumination is changed, the function $E(\lambda)$ will change accordingly, and so will the kernel $K_E(\mu, \lambda)$. Then, it follows that for the same surface with reflectance $R(\lambda)$, we get two color measurements, each one corresponding to each illumination condition, E_1 and E_2, i.e.

$$\left.\begin{array}{rcl} y_1(\mu) & = & \int_{\lambda_0}^{\lambda_1} K_{E_1}(\mu, \lambda)\, R(\lambda)\, d\lambda \\ \\ y_2(\mu) & = & \int_{\lambda_0}^{\lambda_1} K_{E_2}(\mu, \lambda)\, R(\lambda)\, d\lambda \end{array}\right\} \qquad (4.11)$$

Hence, we need to relate two color representations, $y_1(\mu)$ and $y_2(\mu)$, coming from the same *unknown* surface $R(\lambda)$. Conceptually, it is the same case found when representing a point x in two given coordinate systems, i.e., $\psi_1(x) = \mathbf{x}_1$ and $\psi_2(x) = \mathbf{x}_2$. In our situation, the point is the function $R(\lambda)$ and its representation in two different *coordinate systems* are, respectively, $y_1(\mu)$ and $y_2(\mu)$. Thus, an approach based on the color representation of surfaces can be thought as that of finding the *change of coordinates* mapping $y_1(\mu)$ onto $y_2(\mu)$. In general, a change of coordinates of a point is the composition of two basic operations, namely, recovering the point from the old coordinates and projecting this point onto a new set of coordinates, i.e.,

$$\mathbf{x}_2 = (\psi_2 \circ \psi_1^{-1})(\mathbf{x}_1) \qquad (4.12)$$

In terms of integral equations, the recovery involves the solution of Eq. (4.9), whereas the projection stands for the use of the solution previously obtained to get a new color measure. *Id est*,

$$\begin{tikzcd} R(\lambda) \arrow[r, "K_{E_1}"] \arrow[d, "K_{E_2}"'] & y_1(\mu) \arrow[dl, "K_{E_2} \circ K_{E_1}^{-1}"] \\ y_2(\mu) & \end{tikzcd}$$

The study of integral equations in Appendix A provides a general expression for the *inverse* kernel $K_{E_1}^{-1}$ which will be used in the next Section to compute the analytical expression for the color map between two illuminant conditions in terms of the compound transformation $K_{E_2} \circ K_{E_1}^{-1}$.

4.6 Analytical Color Transformation

In this Section we infer the analytical expression for the transformation between two different data (color) functions, each one corresponding to different lighting conditions, as explained in Section 4.5, in terms of the *singular functions and values* of the kernel functions in the integral equations – see Appendix A. Notice this is a completely general expression for the color change which is independent of any particular discretization scheme. This way, the problem of color change can be studied in a wider framework and results will be valid for any posterior approach. In order this equation to be useful in practice it is necessary to further reduce and discretize it, which is carried out later in Section 4.7 and Section 4.8.

Let us think on a certain function f which is *measured* using two different operator, K_1 and K_2, giving rise to two, in general, different data functions $g_1 \in \mathcal{G}_1$ and $g_2 \in \mathcal{G}_2$. Our problem is that of

$$\text{finding} \qquad K_{21} \colon \mathcal{G}_1 \longrightarrow \mathcal{G}_2 \qquad \text{so that} \qquad g_2 = K_{21}\, g_1 \tag{4.13}$$

Formally, operator K_{21} is the composition of operators K_2 and inverse operator K_1^{-1}, as graphically expressed above. Theoretically, the operator K_1^{-1} brings function g_1 onto f, that is, the solution to the IFK equation. Afterwards, the operator K_2 projects the solution f just obtained onto its new representation as a function g_2.

So as to express those ideas in a more precise manner by means of two different IFK equations we obtain that

$$\left.\begin{array}{l} g_1 = K_1 f \\ \\ g_2 = K_2 f \end{array}\right\} \Longrightarrow g_2 = K_2 K_1^{-1} g_1 = K_{21} g_1 \tag{4.14}$$

If we express functions g_1, g_2, and f in terms of the singular values and singular functions of kernels K_1 and K_2, as it is obtained in Section A.4.3 in Appendix A, we get that

$$\begin{array}{rcl|rcl} g_1 & = & \sum_{j=1}^{\infty} \sigma_{1j}\,(f, v_{1j})\, u_{1j} & g_2 & = & \sum_{i=1}^{\infty} \sigma_{2i}\,(f, v_{2i})\, u_{2i} \\ f & = & \sum_{j=1}^{\infty} \frac{1}{\sigma_{1j}}\,(g_1, u_{1j})\, v_{1j} & f & = & \sum_{i=1}^{\infty} \frac{1}{\sigma_{2i}}\,(g_2, u_{2i})\, v_{2i} \\ K_1 & = & \sum_{j=1}^{\infty} \sigma_{1j}\, u_{1j}\, v_{1j} & K_2 & = & \sum_{i=1}^{\infty} \sigma_{2i}\, u_{2i}\, v_{2i} \end{array} \tag{4.15}$$

where $\{u_{1j}, v_{1j}\}$ and σ_{1j} are the singular functions and the singular values of K_1, respectively. The same holds for $\{u_{2i}, v_{2i}\}$ and σ_{2i} with regard to K_2.

If we now go on, step by step, translating every operational relation into a series, we first get that the inversion involves, in fact, solving the equation $g_1 = K_1 f$, i.e.,

$$\textit{Inversion}: \qquad f = K_1^{-1} g_1 \Longrightarrow f = \sum_{j=1}^{\infty} \frac{(g_1, u_{1j})}{\sigma_{1j}}\, v_{1j} \tag{4.16}$$

Then, by projecting the solution function f in accordance to K_2 we get that

$$\textit{Projection}: \qquad g_2 = K_2 f \Longrightarrow g_2 = \sum_{i=1}^{\infty} \sigma_{2i}\,(f, v_{2i})\, u_{2i} \tag{4.17}$$

Putting these two steps together, the composition is expressed as

$$g_2 = K_2 \left(K_1^{-1} g_1\right) \Longrightarrow g_2 = \sum_{i=1}^{\infty} \sigma_{2i}\, (\sum_{j=1}^{\infty} \frac{(g_1, u_{1j})}{\sigma_{1j}}\, v_{1j}, v_{2i})\, u_{2i} \tag{4.18}$$

Then, the data function g_2 can be written in terms of the data function g_1

$$g_2 = \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \left[\frac{\sigma_{2i}}{\sigma_{1j}}\,(v_{2i}, v_{1j}) \right]\, (g_1, u_{1j})\, u_{2i} \tag{4.19}$$

Bearing in mind that each data function can be expressed in terms of singular functions $\{u_{1j}\}$ and $\{u_{2i}\}$, respectively, we obtain that their coefficients are related as follows

$$\left.\begin{array}{rcl} g_{1j} & = & (g_1, u_{1j}) \\ \\ g_{2i} & = & (g_2, u_{2i}) \end{array}\right\} \Longrightarrow g_{2i} = \sum_{j=1}^{\infty} \left[\frac{\sigma_{2i}}{\sigma_{1j}} \, (v_{2i}, v_{1j}) \right] g_{1j} \qquad (4.20)$$

Hence,

$$g_{2i} = \sum_{j=1}^{\infty} b_{ij} \, g_{1j} \qquad (4.21)$$

where

$$b_{ij} = \frac{\sigma_{2i}}{\sigma_{1j}} \, (v_{2i}, v_{1j}) \,, \qquad \forall i, j = 1, \ldots, \infty$$

From this expression, two things can be said if the two parts corresponding to the singular values and to the singular functions are separately considered. First, the terms (v_{2i}, v_{1j}) tell us how different functions v_{2i} are from v_{1j}, and *vice versa*. These functions can not be anyone but depend on the kernels, and thus constrain the solution subspace, as it is appreciated in Eq. (4.15). Hence, there exists a constraint about the portion of the solution that can be recovered since it is only the part expressible as an expansion in both sets of functions, v_{2i} and v_{1j}. Also, the structure of the color map is limited since the dot–product (v_{2i}, v_{1j}) can be zero to a great extent.

In case we assume there exists an orthonormal basis $\{\omega_k\}$ encompassing both spaces $[v_{2i}]$ and $[v_{1j}]$, as suggested by Forsyth in [For90], then the values (v_{1j}, v_{2i}) can be changed into (ω_i, ω_j), which equal to the Kronecker's delta function δ_{ij}. This way,

$$b_{ij} = \frac{\sigma_{2i}}{\sigma_{1j}} \, \delta_{ij} \qquad (4.22)$$

which suggests that the important terms are the diagonal ones b_{ii}.

Secondly, the term $\frac{\sigma_{2i}}{\sigma_{1j}}$ can look like pretty dangerous at once because of the division between tiny values, which tend to zero as indexes grow. However, the series converges because Eq. (A.28) and Eq. (A.32) in Appendix A also converge. So, there is no problem with this expression, at least, theoretically. Difficulties may arise in real data because of errors, so that the series can be ill–conditioned. Nevertheless, in this case the ill–condition is slightly different from the case of the solution of an IFK and depends on the decreasing rate of both σ_{2i} and σ_{1j}.

Hansen in [Han00] points out that the singular values follows either a harmonic progression $\sigma_i \approx i^{-\alpha}$ or a geometric progression $\sigma_i \approx e^{-\alpha i}$, where α is a positive real constant. The worst case appears when values σ_{1j} are very small. In that case, from the previous approximation it can be stated that the quotient $\frac{\sigma_{2i}}{\sigma_{1j}}$ will not grow as fast as that of $\frac{1}{\sigma_{1j}}$, which is an improvement after all.

In case there exists a basis whose coefficients are $b_{ij} = \frac{\sigma_{2i}}{\sigma_{1i}} \, \delta_{ij}$, the only interesting values to study are those in the diagonal, i.e., b_{ii}. Therefore,

$$b_{ii} = \frac{\sigma_{2i}}{\sigma_{1i}} \approx i^{(\alpha_1 - \alpha_2)} \qquad \text{or} \qquad b_{ii} = \frac{\sigma_{2i}}{\sigma_{1i}} \approx e^{i\,(\alpha_1 - \alpha_2)} \qquad (4.23)$$

Hence, in both cases, depending on whether $\alpha_1 > \alpha_2$, $\alpha_1 = \alpha_2$, or $\alpha_1 < \alpha_2$, b_{ii} will surge to ∞, stabilize to one, or decrease to zero, respectively. As it is quite likely than $\alpha_1 \approx \alpha_2$, the coefficients b_{ii} will tend to increase (or decrease) very slowly since $|\alpha_1 - \alpha_2| \approx 0$. Consequently, Eq. (4.21) behaves in general better than Eq. (A.32), which tells us that it is more advisable computing color transformations than to recover reflectance functions, in terms of stability.

In addition, since it is necessary that both the data (color) and the solution (reflectance) spaces be of finite dimension in order the IFK has a unique and well–conditioned solution, it is habitually useful to *truncate* the series to a finite number of elements in order to approximate them at the same time that errors are removed and really the important information is kept, as seen in Appendix A. Furthermore, finite dimensional expansions can be imposed to the equation. This way, Eq. (4.21) is easily expressed in a matrix framework as proposed in Section 4.7 hereafter.

4.7 Discrete Color Transformation

Despite the beauty of continuous functions, we need more practical expressions to handle IFK and to translate the previous framework into the language of matrices and vectors, where numerical schemes for the explicit computation of color mappings can be undergone. This task in relation to the IFK approach is carefully exposed in this Section.

As seen in Appendix A, any IFK is discretized as a linear system of equations, and solved in terms of its *Singular Value Decomposition* (SVD). Additionally, the SVD framework can be used to get a discretization of the general relation between color functions that is shown in Section 4.6.

Suppose we have a pair of matrices $\mathbf{K}_1$ and $\mathbf{K}_2$ (or their truncated versions) belonging to kernels K_1 and K_2, respectively. Using the SVD decomposition to compute inverse matrices, we get that the color mapping matrix $\mathbf{K}_{21}$ is

$$\left.\begin{array}{rcl} \mathbf{K}_1 & = & \mathbf{U}_1\,\mathbf{\Sigma}_1\,\mathbf{V}_1^t \\ \\ \mathbf{K}_2 & = & \mathbf{U}_2\,\mathbf{\Sigma}_2\,\mathbf{V}_2^t \end{array}\right\} \Longrightarrow \mathbf{K}_{21} = \mathbf{K}_2\,\mathbf{K}_1^{-1} = \mathbf{U}_2\,\left[\mathbf{\Sigma}_2\,\left(\mathbf{V}_2^t\,\mathbf{V}_1\right)\,\mathbf{\Sigma}_1^{-1}\right]\,\mathbf{U}_1^t \tag{4.24}$$

Knowing that matrices $\mathbf{U}_1$ and $\mathbf{U}_2$ are merely a pair of changes of coordinates, the important information of the color mapping $\mathbf{K}_{21}$ is embodied in matrix $\mathbf{B}$, defined as

$$\mathbf{B} = \mathbf{\Sigma}_2\,\left(\mathbf{V}_2^t\,\mathbf{V}_1\right)\,\mathbf{\Sigma}_1^{-1} \tag{4.25}$$

Then

$$\mathbf{K}_{21} = \mathbf{U}_2\,\mathbf{B}\,\mathbf{U}_1^t \tag{4.26}$$

For simplicity, let us write color vectors $\mathbf{g}_1$ and $\mathbf{g}_2$ in the vector spaces defined by matrices $\mathbf{U}_1$ and $\mathbf{U}_2$, respectively. That is,

$$\left.\begin{array}{rcl} \tilde{\mathbf{g}}_1 & = & \mathbf{g}_1\,\mathbf{U}_1 \\ \tilde{\mathbf{g}}_2 & = & \mathbf{g}_2\,\mathbf{U}_2 \end{array}\right\} \tag{4.27}$$

So, Eq. (4.21) turns into

$$\tilde{\mathbf{g}}_2^t = \mathbf{B}\,\tilde{\mathbf{g}}_1^t$$

where

$$\begin{array}{rcl} (\mathbf{B})_{ij} & = & \frac{\sigma_{2i}}{\sigma_{1j}}\left(\mathbf{v}_{2i}\cdot\mathbf{v}_{1j}^t\right) \\ \tilde{\mathbf{g}}_1 & = & (\tilde{g}_{11},\ldots,\tilde{g}_{1n_1}) \\ \tilde{\mathbf{g}}_2 & = & (\tilde{g}_{21},\ldots,\tilde{g}_{2n_2}) \end{array} \tag{4.28}$$

The dimension of vectors $\tilde{\mathbf{g}}_1$ and $\tilde{\mathbf{g}}_2$ are n_1 and n_2, respectively. $\mathbf{B}$ is a $n_1 \times n_2$ matrix. Since the sensor is usually the same, the number of data components does not vary, i.e., $n_1 = n_2$. Vectors $\mathbf{v}_{1j}^t$ and $\mathbf{v}_{2i}^t$ are the j^{th} and i^{th} columns of matrices $\mathbf{V}_1$ and $\mathbf{V}_2$, respectively. These vectors are fixed by the sensor and the illumination subspace. Therefore, it is clear that any color transformation can be written as a linear transformation $\mathbf{B}$ between color vectors.

The next step consists in computing any particular matrix $\mathbf{K}$ from real data. In Section A.4 of Appendix A we describe some ways of discretizing an IFK equation for the obtaining of the previous matrix. Due to the kind of data used in spectral recovery and color constancy, we think a good election is to consider the discretization attained using two different families of orthonormal basis functions to span the data (color) g and the solution (reflectance) f subspaces, respectively.

Precisely, if there are two sets of *orthonormal* basis functions $\{\psi_i\}_{i=1,\ldots,n}$ and $\{\theta_j\}_{j=1,\ldots,m}$ to expand functions g and f, respectively, we get that

$$g(\mu) = \sum_{i=1}^{n} g_i \, \psi_i(\mu) \qquad \text{and} \qquad f(\lambda) = \sum_{j=1}^{m} f_j \, \theta_j(\lambda) \tag{4.29}$$

Thus, the IFK equation is discretized in the following way

$$g(\mu) = \sum_{i=1}^{n} g_i \, \psi_i(\mu) = \sum_{j=1}^{m} f_j \int_a^b K_E(\mu, \lambda) \, \theta_j(\lambda) \, d\lambda \tag{4.30}$$

so that

$$g_i = (g, \psi_i) = \sum_{j=1}^{m} f_j \int_a^b \!\! \int_a^b K_E(\mu, \lambda) \, \psi_i(\mu) \, \theta_j(\lambda) \, d\mu d\lambda \tag{4.31}$$

In matrix formulation, the system is

$$\mathbf{g}^t = \mathbf{K} \, \mathbf{f}^t$$

where

$$\begin{array}{rcl} (\mathbf{K})_{ij} & = & \int_a^b \!\int_a^b K_E(\mu, \lambda) \, \psi_i(\mu) \, \theta_j(\lambda) \, d\mu d\lambda \\ \mathbf{g} & = & (g_1, \ldots, g_n) \\ \mathbf{f} & = & (f_1, \ldots, f_m) \end{array} \tag{4.32}$$

Given a matrix that encompasses a set of data vectors as columns, an orthogonal basis spanning these vectors is straightforwardly obtained from the SVD decomposition of the data matrix by taking from matrix $\mathbf{U}$ the set of vector columns corresponding to the greatest singular values [PFTV93]. This technique is widely used in Section 4.9, devoted to experiments and results.

However, as also will be seen later, the number of dimensions of the data vector $\tilde{\mathbf{g}}$, which corresponds to the continuous color functions, are not always the same as these of the RGB color vectors we are given in a real case. So, a way to relate the RGB coordinates and the continuous color components must be provided. A feasible method consists in computing, by least squares fitting, a matrix $\mathbf{A}$ relating the RGB data matrix $\mathbf{Y}$ and the continuous color data matrix $\mathbf{X}$, i.e., $\mathbf{Y} = \mathbf{A}\,\mathbf{X}$. Hence, the matrix $\mathbf{A}$ is obtained as

$$\mathbf{A} = \mathbf{Y}\,\mathbf{X}^t \left(\mathbf{X}\,\mathbf{X}^t\right)^{-1} \tag{4.33}$$

This way, if RGB values are the available data, the discretized IFK will be given by

$$\mathbf{c}^t = \mathbf{A}\,\mathbf{g}^t = (\mathbf{A}\,\mathbf{K})\;\mathbf{f}^t \tag{4.34}$$

where $\mathbf{c}$ is the RGB color vector. Therefore, the color mapping between two RGB colors $\mathbf{c}_1$ and $\mathbf{c}_2$ is as follows

$$\mathbf{c}_2^t = \left(\mathbf{A}_2\,\mathbf{K}_{21}\,\mathbf{A}_1^{-1}\right)\;\mathbf{c}_1^t = \tilde{\mathbf{K}}_{21}\,\mathbf{c}_1^t \tag{4.35}$$

where $\mathbf{K}_{21}$ is the color mapping computed in Eq. (4.24), and $\mathbf{A}_1$ and $\mathbf{A}_2$ are the matrices changing continuous components onto RGB under the two conditions of illumination. Thus, matrix $\tilde{\mathbf{K}}_{21}$ is the color mapping in RGB coordinates.

4.8 IFK Relation with Former Color Models

Now we want to manifest explicitly how the IFK generalization relates with some other well–known existing models on which color constancy has been based. It is our intention to show that these models can be derived from an IFK–based model taking into consideration some of the various mechanisms of discretization suggested in Appendix A.

There are an important lot of researchers who have studied this problem and so are there many different approaches, as was seen in Chapter 2. Nevertheless, three are the main directions into which these multiplicity of models are split hereafter. Basically, we are referring to as *Bilinear*, *Quadrature*, or *Diagonal* models, which are briefly taken into account next.

4.8.1 Bilinear Model

Finite–dimensional linear models for color constancy were introduced by Buchsbaum [Buc80] who extended previous works by Helson [Hel38] and Judd [Jud40]. Their use is based on empirical studies showing that both illumination and reflectance are relatively smooth functions of the wavelength of light in the visual spectrum.

Specifically, Judd et al. [JMW64] showed that the daylight could be accurately described by a linear mixture of three fixed basis functions. The reflectance functions of a great variety of materials have also been studied. Cohen [Coh64], Parkkinen et al. [PHJ86], Maloney [Mal86] studied the reflectance properties of the Munsell chips, which finally were described by anything from three to eight basis functions. The reflectance of naturally occurring surface materials can, with a few exceptions such as fluorescent materials and some metals, be approximated in a similar way (Vrhel et al. [VGI94], Krinov [Kri47], Wyszecki and Stiles [WS82]).

Formally, we have that functions $E(\lambda)$ and $R(\lambda)$ can be approximated as

$$E(\lambda) \approx \sum_{i=1}^{n} e_i\,E_i(\lambda) \qquad \text{and} \qquad R(\lambda) \approx \sum_{j=1}^{m} r_j\,R_j(\lambda) \tag{4.36}$$

where $E_i(\lambda)$ and $R_j(\lambda)$ are the *basis functions* for the illumination and the reflectance, respectively, and e_i and r_j are the corresponding coefficients. Numbers n and m are the dimension of the basis and are different in general. The main advantage of using finite–dimensional linear models is to provide a compact description of data, that is, very few basis functions can represent pretty general functions of both illumination spectrum and surface reflectance functions.

In general, these basis functions must fulfill some physical properties so as to correspond to feasible illumination and reflectance functions. The most important constraints are these of being *real–valued* and *nonnegative*. Other properties would be related with their well–behavior, such as being bounded, continuous and integrable functions – depending upon the context. SVD analysis gives a set of orthonormal basis functions which is optimal, although some function can happen to be negative for some λ. It might be therefore preferable to use a nonorthogonal, *positive* everywhere, basis even if it is not optimal in the sense of the least squares.

The relation between the discretization methods in Appendix A and the finite–dimensional linear model is pointed out in the following paragraphs. First, this approach is embedded into the *general series expansion* scheme where both data and solution functions are described in terms of their basis functions. As previously said, the set of basis functions representing the surface reflectance might be nonorthonormal, in which case we should use the *Galerkin method* to get a discretization of the IFK. Additionally, a real sensor having a finite set of responses is straightforward described using the *collocation method* over the solution function.

These two constraints over the data and solution functions supply us with the next expression

$$\left.\begin{array}{rcl} y(\mu) & = & \sum_{h=1}^{p} y_h\,\delta(\mu-\mu_h) \\ R(\lambda) & = & \sum_{j=1}^{m} r_j\,R_j(\lambda) \end{array}\right\} \Longrightarrow y_h = \sum_{j=1}^{m} r_j \int_{\lambda_0}^{\lambda_1} K_E(\mu_h,\lambda)\,R_j(\lambda)\,d\lambda \tag{4.37}$$

In order to approximate the kernel K_E, we use its definition and turn it into a separable kernel thanks to representing the illumination function as a (truncated) series

$$\left.\begin{array}{rcl} K_E(\mu_h,\lambda) & = & S(\mu_h,\lambda)\,E(\lambda) \\ S(\mu_h,\lambda) & = & S_h(\lambda) \\ E(\lambda) & = & \sum_{i=1}^{n} e_i\,E_i(\lambda) \end{array}\right\} \Longrightarrow y_h = \sum_{i=1}^{n}\sum_{j=1}^{m} e_i r_j \int_{\lambda_0}^{\lambda_1} S_h(\lambda)\,E_i(\lambda)\,R_j(\lambda)\,d\lambda \tag{4.38}$$

Thus, we obtain the following system of linear equations

$$y_h = \sum_{i=1}^{n}\sum_{j=1}^{m} e_i\,r_j\,\left(\mathbf{K}_i\right)_{hj} = \sum_{j=1}^{m}\left(\sum_{i=1}^{n} e_i\,\left(\mathbf{K}_i\right)_{hj}\right) r_j, \qquad h=1,\ldots,p \tag{4.39}$$

where

$$\left(\mathbf{K}_i\right)_{hj} = \int_{\lambda_0}^{\lambda_1} S_h(\lambda)\,E_i(\lambda)\,R_j(\lambda)\,d\lambda \tag{4.40}$$

In a matrix framework, it can also be written as

$$\mathbf{y}^t = \left(\sum_{i=1}^{n} e_i\,\mathbf{K_i}\right)\mathbf{r}^t = \mathbf{K}\,\mathbf{r}^t \tag{4.41}$$

where

$$\begin{array}{rcl} \left(\mathbf{K}\right)_{hj} & = & \sum_{i=1}^{n} e_i\,\left(\mathbf{K}_i\right)_{hj} \\ \mathbf{r} & = & (r_1,\ldots,r_m) \\ \mathbf{y} & = & (y_1,\ldots,y_p) \end{array} \tag{4.42}$$

This model is usually referred to as the *Bilinear model* and relates the measured color $\mathbf{y}$ to the coefficients that represent the surface reflectance $\mathbf{r}$ by means of the matrix $\mathbf{K}$, which encompasses the information about the basis functions for both light and reflectance, along with the sensor sensitivity. A given illumination induces its own matrix $\mathbf{K}$ as a linear combination of matrices $\mathbf{K}_i$, each one belonging to the i^{th} illumination basis function. Coefficients of this mixture generate a particular illumination by means of a linear combination of basis functions.

Finally, it is interesting to state that the finite–dimensional model has been extensively used in some color constancy algorithms where the main aim was that of recovering either the illumination or the reflectance functions, or both of them. Main works in such topics are those of Maloney and Wandell [MW86], Marimont and Wandell [MW92], and its extension by D'Zmura and Inverson [DI93], which uses various surfaces viewed under several different illuminants.

An interesting work by Ho et al. [HFD90] wants to recover both illumination and reflectance employing an iterative algorithm. This scheme was later extended into a nonlinear iterative scheme by Chang et al. [CH95], but the improvement was just marginally better. Other methods not trying to separate both types of signals but still based on finite–dimensional models are those of Tominaga and Okajima [TO00] and Cheng et al. [FC84]. Recently, Sapiro [Sap98, Sap99] proposed a procedure for color constancy whose first step was the recovery of the illumination by voting among some of the possibilities[2].

4.8.2 Quadrature Model

Some algorithms, such as those of Wandell [Wan87] and Sharma and Trussell [ST93, ST96], use a discretization of both illumination and reflectance functions instead of that of their basis functions. This procedure is related to the quadrature rule described in Section A.5 of Appendix A. If we apply a *numerical integration* and the *collocation method* we get the following discretization

$$y(\mu_h) = \sum_{i=1}^{n} \omega_i \, K_E(\mu_h, \lambda_i) \, R(\lambda_i), \qquad h = 1, \ldots, p \tag{4.43}$$

Thus, the kernel has been approximated by its values at points (μ_h, λ_i)

$$\left.\begin{array}{rcl} K_E(\mu_h, \lambda_i) & = & S(\mu_h, \lambda_i) \, E(\lambda_i) \\ S_{hi} & = & S(\mu_h, \lambda_i) \end{array}\right\} \Longrightarrow y_h = \sum_{i=1}^{n} \omega_i \, S_{hi} \, E(\lambda_i) \, R(\lambda_i) \tag{4.44}$$

Defining the set of constants e_i and r_i, we can rewrite the equation as

$$\left.\begin{array}{rcl} e_i & = & E(\lambda_i) \\ r_i & = & R(\lambda_i) \end{array}\right\} \Longrightarrow y_h = \sum_{i=1}^{n} \omega_i \, S_{hi} \, e_i \, r_i, \quad h = 1, \ldots, p \tag{4.45}$$

In order to obtain a linear system of equations in terms of reflectances, it is useful to define two matrices representing the sensor and the illumination separately. Matrix $\mathbf{S}$ embodies the sensor sensitivity functions, while matrix $\mathbf{E}$ is a diagonal matrix with the discrete illumination spectrum

$$\left.\begin{array}{rcl} (\mathbf{S})_{hi} & = & \omega_i \, S_{hi} \\ \mathbf{E} & = & diag(e_i) \end{array}\right\} \tag{4.46}$$

[2] Equations can be reversed to compute the illumination E if K_R is used instead of K_E.

In matrix formalism

$$\mathbf{y}^t = (\mathbf{S}\,\mathbf{E})\ \mathbf{r}^t = \mathbf{K}\, r^t \tag{4.47}$$

where

$$\begin{array}{rcl} (\mathbf{K})_{hi} & = & \omega_i\, S_{hi}\, e_i \\ \mathbf{r} & = & (r_1, \ldots, r_n) \\ \mathbf{y} & = & (y_1, \ldots, y_p) \end{array} \tag{4.48}$$

Therefore, the color transformation between two illumination conditions is as follows

$$\mathbf{K}_{21} = (\mathbf{S}_2\, \mathbf{E}_2)\ (\mathbf{S}_1\, \mathbf{E}_1)^{-1} = \mathbf{S}_2\ \left(\mathbf{E}_2\, \mathbf{E}_1^{-1}\right)\ \mathbf{S}_1^{-1} \tag{4.49}$$

Since the sensor has not changed, $\mathbf{S}_1 = \mathbf{S}_2 = \mathbf{S}$, the final color map is

$$\mathbf{K}_{21} = \mathbf{S}\ \left[diag\left(\frac{e_{2i}}{e_{1i}}\right)\right]\ \mathbf{S}^{-1} \tag{4.50}$$

This approach is pretty similar to the Bilinear model previously exposed, but rather here the transference matrix has been split into two independent matrices with information about the illumination and the sensor each, whereas in the previous approach the transference matrix was a linear combination of them. As mentioned in Section A.5 of Appendix A, the quadrature approach is not the best way to discretize an IFK, being numerically unstable and little accurate, as reported in [ST96]. Furthermore, the splitting of matrix $\mathbf{K}_{21}$ into illumination and sensor matrices is kind of simplistic and artificial, while parallel objectives and better precision are attained by the SVD decomposition.

4.8.3 Diagonal Model

A diagonal model for the color change was first introduced by von Kries [WS82] and is based on the hypothesis by which the set of sensors are totally independents one another in a way that any variation in one of these sensors does not affect the measures carried out by the rest of them.

This supposition can be mathematically written in several ways, such as by means of a set of sensor functions which do not overlap at all one another. Nevertheless, the usual expression is the one called *sharp sensor functions*, which can be simply written down in terms of Dirac's δ-functions [FDF93b, FDF94b, BF98, DF00], despite it may not be the most accurate approach.

If we express the fact of having a finite number of sensors as taking a finite set of measures by a *collocation scheme*, we will obtain the following expression

$$y(\mu) = \sum_h^p y_h\, \delta(\mu - \mu_h) \Longrightarrow y_h = \int_{\lambda_0}^{\lambda_1} K_E(\mu_h, \lambda)\, R(\lambda)\, d\lambda\,,\ \ h = 1, \ldots, p \tag{4.51}$$

Now, every sensor is considered as a sharp sensor with its δ–function center at a precise wavelength

$$\left.\begin{array}{rcl} K_E(\mu_h, \lambda) & = & S_h(\lambda)\, E(\lambda) \\ S_h(\lambda) & = & S_h\, \delta(\lambda - \lambda_h) \end{array}\right\} \Longrightarrow y_h = \int_{\lambda_0}^{\lambda_1} S_h\, \delta(\lambda - \lambda_h)\, E(\lambda)\, R(\lambda)\, d\lambda \tag{4.52}$$

As a consequence,

$$y_h = S_h\, E(\lambda_h)\, R(\lambda_h)\,,\ \ h = 1, \ldots, p \tag{4.53}$$

With the help of the set of constants e_h and r_h as values of the illumination and reflectance functions, respectively, evaluated in a series of points λ_h, we get that

$$\left.\begin{array}{rcl} e_h &=& E(\lambda_h) \\ r_h &=& R(\lambda_h) \end{array}\right\} \Longrightarrow y_h = S_h\, E(\lambda_h)\, R(\lambda_h) = S_h\, e_h\, r_h \tag{4.54}$$

In matrix terms

$$\mathbf{y}^t = \mathbf{K}\,\mathbf{r}^t$$

where

$$\begin{array}{rcl} \mathbf{K} &=& diag(S_h\, e_h) \\ \mathbf{r} &=& (r_1, \ldots, r_p) \\ \mathbf{y} &=& (y_1, \ldots, y_p) \end{array} \tag{4.55}$$

Here, the transference matrix $\mathbf{K}_{21}$ is the simplest possible and is composed of only a diagonal matrix. Coefficients at the diagonal embrace both sensor sensibility and illumination. This structure is similar to the models considered before, but in the present case the relation is even far simpler. Therefore, the color transformation between two illumination conditions is

$$\mathbf{K}_{21} = \mathbf{K}_2\,\mathbf{K}_1^{-1} = diag\left(\frac{S_{2h}\, e_{2h}}{S_{1h}\, e_{1h}}\right) \tag{4.56}$$

Since the sensor has not changed, $S_{1h} = S_{2h}$, the final color map is

$$\mathbf{K}_{21} = diag\left(\frac{e_{2h}}{e_{1h}}\right) \tag{4.57}$$

The diagonal model has been extensively used in several color constancy algorithms because of its simplicity. Forsyth [For90] applied this model as a simplification of a linear transformation in his method based on the gamut–mapping. Afterwards, Finlayson et al. in a very extensive number of works [FDF93a, FDF94a, Fin95a, Fin95b, FF96, Fin96, FH97, FH98b, FH99, FH00, Fin00, FHH01] have been using as well the diagonal model along with the habitual RGB coordinates and a modified version of them named *perspective color*, which consists in factoring out the intensity from the original responses dividing them by the response of one of the color channels, usually the blue (B) one.

4.9 Experiments and Results

This Section encompasses a set of experiments whose aim is to evaluate the performance of the former physical models in two kind of tasks, namely, the recovery of spectral functions and the explicit computation of color mappings. Nonetheless, after describing the set of data employed in the subsequent computations, the first problem to considered is that of fitting the sensor sensitivity in order to obtain a continuous function.

Afterwards, our continuous color formation approach is compared with the Bilinear model in the spectral recovery task in order to study their similarities and differences as well as their respective performances. Later, a similar analysis is carried out comparing the IFK approach with all other physical models considered in the previous Section in computing color mappings between different lighting conditions.

Since linear applications are suggested as the general model for the color mappings taken into account here, a regression using a linear least squares fitting is accomplished as an additional way to evaluate the feasibility of such linear assumption. These results are also a reference for the performance of previous physical models because they represent the best possible result as long as a linear transformation is taken into consideration.

4.9.1 Data Description

First of all, let us briefly talk about the data employed throughout this Chapter to carry out the set of experiments mentioned before. We are using illumination and reflectance information that has been previously recorded by a number of authors, and put together and made completely available to research at `http://www.cs.arizona.edu/~kobus/` by Barnard. This data is divided into spectral power distributions describing usual illuminants and spectral reflectance functions for the surfaces. A complete description of the whole set of data can be found in [BFC00, BCF02, BMFC02], though we succinctly take it into account hereafter.

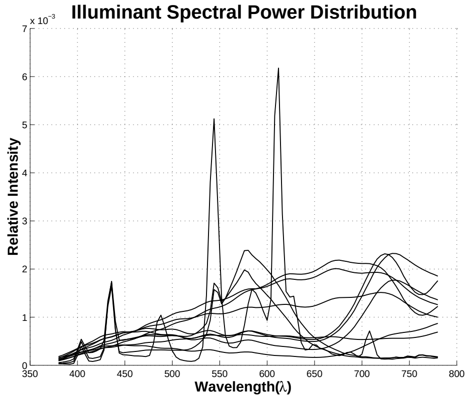


Figure 4.2: Examples of spectral power distributions of some usual illuminants.

The set of illuminant data consists of four different groups of illuminants corresponding to both real and simulated lights. Leading the collection, there is a set of 11 illuminants commonly encountered in human–made indoor environments (fluorescent lamps and bulbs). Since these seemed to be too few illuminants, there are two other sets of 102 and 11 measured lights, respectively. In addition to them, there still exist two other sets created as linear combinations of randomly selected existing illuminants in order to generate more samples to enlarge the total number of available lights.

Barnard generated 287 and 87 illuminants in two new sets. The total number of feasible illuminations is then 498. Nevertheless, we only took into account 124 of them – those corresponding to real illuminants – since it is a sufficient number of samples for our purposes. A special consideration is given to the first 11 illuminants, which have been used as canonic lights along this work. These functions are exhibited as an example of spectral power distributions in Fig. 4.2.

From this graph, it is at once evident that we are managing at least two kinds of illuminant functions. Basically, some illuminants have smooth and slow–varying functions, while others have pretty steep peaks at a quite restricted group of wavelengths. First kind of illuminations corresponds to incandescent sources, such as bulbs or the sun. The second family is the one generated by fluorescent lamps.

In respect to the reflectances, there are many available sets, such as the measures by Cohen [Coh64], Parkkinen et al. [PHJ86], Maloney [Mal86], Vrhel et al. [VGI94], Krinov [Kri47], and Wyszecki and Stiles [WS82], estimating both Munsell chips and surface reflectances of natural objects. The importance of the job done by Barnard dwells in the fact of collecting all those disperse sets into a huge compilation of 3989 different reflectance functions. Some of these functions, corresponding to natural objects, can be seen in Fig. 4.3. In general, reflectances are smooth functions and do not present defects such as discontinuities, highly varying episodes, peaks, notches, gaps, or ripples.

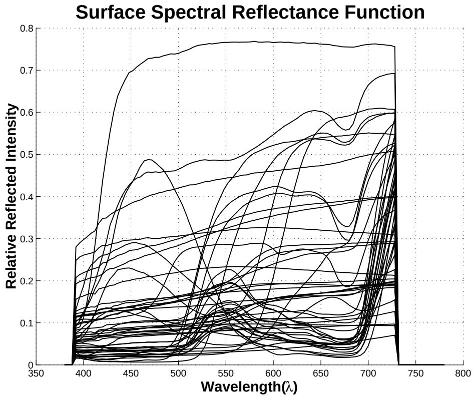


Figure 4.3: Examples of spectral reflectance functions of some natural surfaces.

In some of the following applications described by and by it is necessary to extract orthonormal basis spanning the above sets of data. The right way to construct an orthonormal basis for a subspace is by means of the SVD decomposition of a data matrix $\mathbf{A}$. Hence, if $\mathbf{A} = \mathbf{U}\,\mathbf{\Sigma}\,\mathbf{V}^t$, then the columns contained in matrix $\mathbf{U}$ encompass the vectors of the desired orthonormal basis. In addition, it might be advisable checking the resulting singular values in $\mathbf{\Sigma}$ for zero values. If any occurs, then the spanned subspace has not, in fact, the same dimension as the number of rows of $\mathbf{A}$. Hence, the columns of $\mathbf{U}$ corresponding to zero singular values should be discarded from the orthonormal basis. As will be seen later, sometimes it is better to reduce even further the number of vectors in order to span the subspace with a higher precision, taking into account only the vectors corresponding to a few of the greatest singular values.

4.9.2 Fitting of Continuous Sensors

In order to apply any computational scheme derived earlier from the IFK model we need to rebuild a continuous *virtual* sensor function $S(\mu, \lambda)$ out of the set of *real sensor sensitivities* $S_k(\lambda)$, such that $S_k(\lambda) = S(\mu_k, \lambda)$, for $k = 1, \ldots, p$ and p being the number of camera sensors. Note that in the RGB case, $B = S(\mu_1, \lambda)$, $G = S(\mu_2, \lambda)$, and $R = S(\mu_3, \lambda)$.

The main problem we face at this stage is the scarce number of real measures $S_k(\lambda)$ we commonly possess. As known, a usual color camera has only three responses RGB, so $p = 3$. In multispectral cameras, $p > 3$. For the purpose of increasing the available number of such responses, the first thing to account for is the interpolation of the existing measures so as to be able to extract new data in a posterior extrapolation stage. Since extrapolated points have no measurable counterpart, our only constraint on the feasible sensor functions is that of being as *smooth* as possible. The method applied to build such a continuous sensor function is fully described hereafter.

Unfortunately, it was not feasible to interpolate any result smooth and precise enough for our requirements taking as the source of information the one provided by the sensor sensitivities in Fig. 4.4. To solve this lack of information, we suggest employing the idea of *continuously deforming* a function until it reaches any other function we want. This way, once a morphing scheme is proposed, it will be possible to get any intermediate value between two real sensor sensitivity functions satisfying the smoothness constraint. In other words, we get all the in–between functions $S(\mu, \lambda)$, $\forall \mu \in [\mu_k, \mu_{k+1}]$, where $k = 1, \ldots, p-1$.

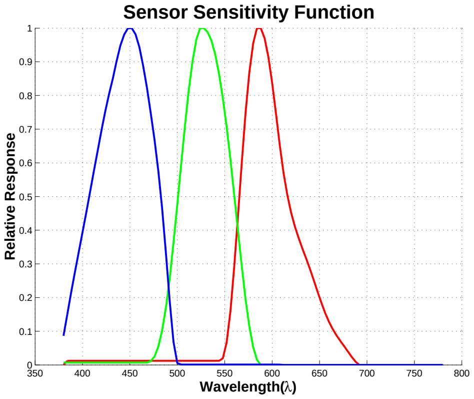
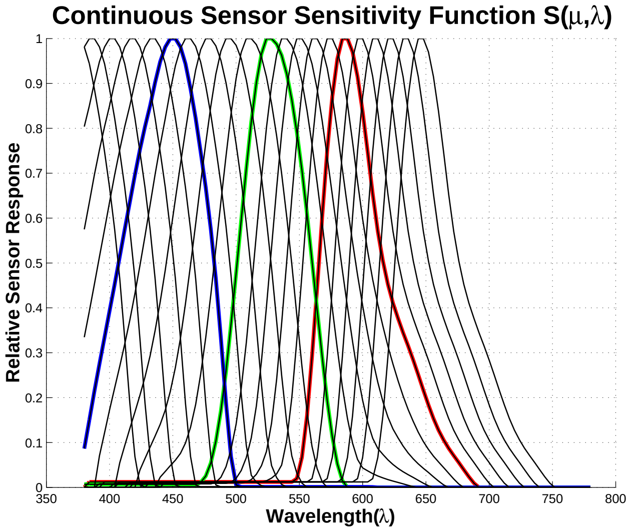


Figure 4.4: Sensor sensitivity functions of the camera Sony XDC–930.

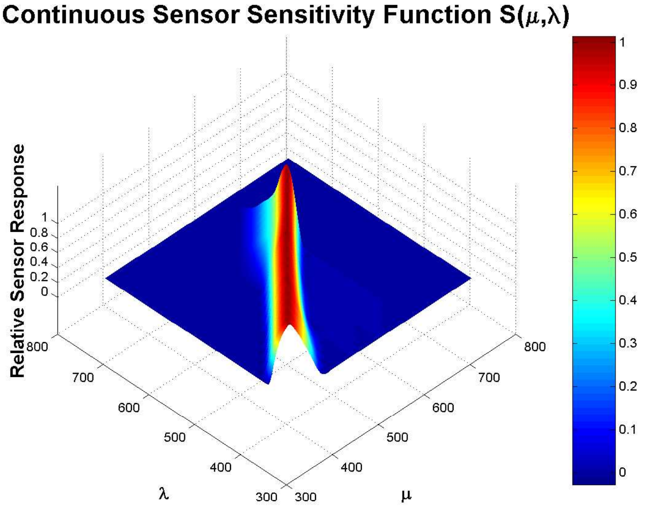
The idea to get a function $S(\mu, \lambda)$ is pretty simple and consists in computing for each point $\mu \in [\mu_k, \mu_{k+1}]$ a linear blending of the two functions $S(\mu_k, \lambda)$ and $S(\mu_{k+1}, \lambda)$. Nevertheless, this will not be directly carried out on the original functions, but rather on a shifted version of them. More precisely, each of the two functions, $S(\mu_k, \lambda)$ and $S(\mu_{k+1}, \lambda)$, will differently shift depending on the side of the interval it is placed. The right–hand function $S(\mu_{k+1}, \lambda)$ shifts to the left, while the left–hand function $S(\mu_k, \lambda)$, to the right. Note that functions $S(\mu_k, \lambda)$, for any $k = 1, \ldots, p$, correspond to actual sensor sensitivities $S_k(\lambda)$.

Mathematically, we define an intermediate sensor function $S(t, \lambda)$ parametrized by $t \in [0, 1]$ spanning between sensors $S(\mu_k, \lambda)$ and $S(\mu_{k+1}, \lambda)$ as

$$S(t, \lambda) = (1 - t) \cdot \tilde{S}_k(t, \lambda) + t \cdot \tilde{S}_{k+1}(t, \lambda)\,, \ t \in [0, 1] \qquad (4.58)$$



(a)



(b)

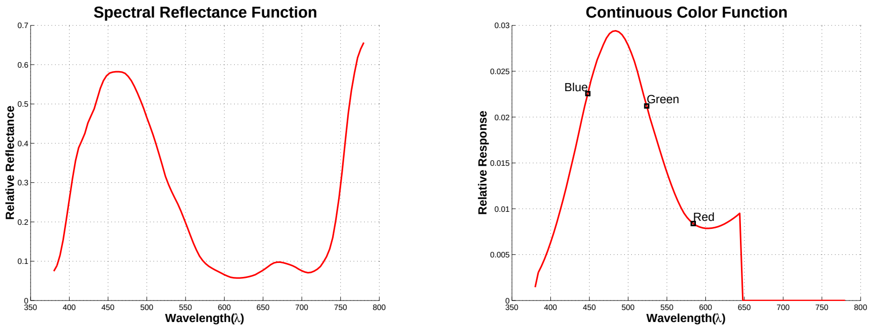
Figure 4.5: Continuous sensor sensitivity function $S(\mu, \lambda)$: (a) Slice view, $y = S(\mu_i, \lambda)$, where each μ_i remains constant. (b) 3D view, $z = S(\mu, \lambda)$.

where functions $\tilde{S}_k(t,\lambda)$ and $\tilde{S}_{k+1}(t,\lambda)$ are the shifted versions of functions $S(\mu_k,\lambda)$ and $S(\mu_{k+1},\lambda)$, respectively. Precisely, if $\Delta = \mu_{k+1} - \mu_k$, we get that

$$\left.\begin{array}{lrcl} \text{Rightmost:} & \tilde{S}_{k+1}(t,\lambda) & = & S_{k+1}(\lambda + \Delta\cdot(1-t)) \\[2ex] \text{Leftmost:} & \tilde{S}_k(t,\lambda) & = & S_k(\lambda - \Delta\cdot t) \end{array}\right\} \tag{4.59}$$

Consequently, it is true that $S(0,\lambda) = \tilde{S}_k(0,\lambda) = S_k(\lambda)$ and $S(1,\lambda) = \tilde{S}_{k+1}(1,\lambda) = S_{k+1}(\lambda)$. Moreover, $\tilde{S}_k(1,\mu_{k+1}) = S_k(\mu_k)$ and $\tilde{S}_{k+1}(0,\mu_k) = S_{k+1}(\mu_{k+1})$. For any $t \in (0,1)$, the function $S(t,\lambda)$ is, as shown in Fig. 4.5(a), a continuous function spanning somewhere in between $S_k(\lambda)$ and $S_{k+1}(\lambda)$.

The variable $\mu \in [\mu_k,\mu_{k+1}]$ is straightforwardly transformed into the parameter driving the above equations $t \in [0,1]$ using the expression $t = (\mu - \mu_k)/(\mu_{k+1} - \mu_k)$, for any $k = 1,\ldots,p-1$. Therefore, we are able to get a piecewise–defined function $S(\mu,\lambda)$ for any $\mu \in [\mu_1,\mu_p]$. The interval domain can be further stretched if functions at the edges, S_1 and S_p, are indefinitely repeated for any wavelength smaller than μ_1 and greater than μ_p, respectively.



(a) (b)

Figure 4.6: Continuous sensor response: (a) Spectral Reflectance. (b) Continuous Color Function. Black squares represent RGB response.

The result of the above set of equations is exhibited in Fig. 4.5(a), where the sensor sensitivity functions of the camera Sony XDC–930 CCD[3] depicted in Fig. 4.4 were extended through the parameter μ as an example of the sensor fitting method exposed here. These functions were normalized so that their maximum value was the unity. As a way to make our point clearer, in Fig. 4.5(a) real sensor functions are painted in their correspondent color. This way, it is easy to notice how the shape smoothly changes from one sensor to another.

Due to the discretization and the different distances between S_1 and S_2, and S_2 and S_3, middle functions thus built are not evenly spaced. Moreover, it is necessary to get some more intermediate values in order to properly integrate the equations related with the set of aforementioned physical color models. Hence, we make use of these functions as a way to interpolate the surface $z = S(\mu,\lambda)$,

[3] This is a typical example of CCD color camera and was employed to take the images that will be posteriorly used in the Chapter devoted to color constancy. Further information about it can be found at `http://www.cs.arizona.edu/~kobus/`

which encompasses all of those sensor functions, employing the *bicubic* interpolation scheme. The resulting surface that represents the continuous sensor sensitivity function $S(\mu, \lambda)$ is displayed in Fig. 4.5(b). We painted its height – z value – using the color scale on the right side, where blue stands for zero and red stands for one. It is important to notice that the original sensor functions represent only three slices of the new function, so there has been an increase in the available information related to the sensors.

As an example of the IFK–based color formation for the generation of continuous responses, we show in Fig. 4.6(b) the continuous color function obtained from the reflectance depicted in Fig. 4.6(a). We marked the true RGB responses as black squares in Fig. 4.6(b). Smoothness of the color function is clearly appreciable as well as the coincidence with the true responses provided by the camera. Both of these features are fully desirable in the definition of a continuous sensor function $S(\mu, \lambda)$.

4.9.3 Spectral Recovery of Reflectance Functions

This Section is versed on the study of the spectral recovery of reflectance functions from sensor responses. To that purpose, we employ both the responses generated by the continuous sensor as well as those which were obtained from the more common RGB measures for the color of surfaces. In addition to our IFK description of that issue, the Bilinear model described in Section 4.8.1 is also applied in order to compare both approaches.

Besides the interest that spectral recovery might arise by itself, our final concern in this Chapter is the explicitly computation of color mappings. As explained in Section 4.5, any color map may be see as the composition of two applications, an *inversion* by solving an IFK, and the *projection* of the recovered function onto a new color subspace. The first step, the inversion, also implies a spectral recovery and follows the basic principles stated in Appendix A, where some general numeric schemes to invert IFKs are suggested. The method which best suits our data is the one described in Section A.5, where two families of orthonormal functions are used to describe the data (reflectance) and the solution (color) functions.

Because of the ill–conditioning of the inversion problem, the number of basis vectors selected to build a particular solution *drastically* affects the quality of it. This is not a problem of simply taking more dimensions, but one of choosing the correct set of vectors. This issue is investigated hereafter, namely, the selection of the correct basis which best recovers reflectance functions in order to later compute a complete color mapping. As explained in Section 4.9.1, those basis are fruit of the SVD decomposition of the reflectance and the continuous color functions, and will be different each other in general.

Under a certain illuminant, the IFK color formation model generates a continuous color function for each reflectance. Since 124 illuminants were utilized, there will be 124 sets containing 3498 color functions each. An orthonormal basis will be extracted for each of these sets, valid only for one single illumination conditions each. On the other hand, similarity between recovered and actual reflectance functions are also dependent on the number of basis vectors used. Hence, in order to obtain the best recovery results when using the Bilinear model the optimal number of basis vectors for the reflectance subspace is also investigated.

The main difference between the two approaches at these stage lies on how *continuity* is translated into their finite–dimensional counterparts. While the Bilinear model uses orthonormal expansions for both the illuminant and the reflectance functions, our model does likewise for the continuous color functions and the reflectance functions, leaving both the sensor and the illumination functions alone. The chief advantage of the IFK technique dwells, besides its mathematical soundness, not only in not being restricted to the usual three color coordinates, but also in coping with the intrinsic multispectral nature of the recovery problem.

The next experiments compute an error measure between every recovered function and its corresponding actual one. We employ the *2–norm* of the difference between these two functions, which are, in fact, expressed as vectors in a certain orthonormal basis. Thus, for the i^{th} illuminant the total amount of error is

$$E_i = \sum_{k=1}^{n} \|\mathbf{r}_k - \tilde{\mathbf{r}}_k\| = \sum_{k=1}^{n} \|\mathbf{r}_k^t - \mathbf{K}_i^+ \, \mathbf{y}_k^t\| \tag{4.60}$$

where $\mathbf{y}_k$ and $\mathbf{r}_k$ are, respectively, the color and the reflectance functions expressed as vectors. The number of functions is always the same – $n = 3498$ –, so it is not necessary to compute the mean error.

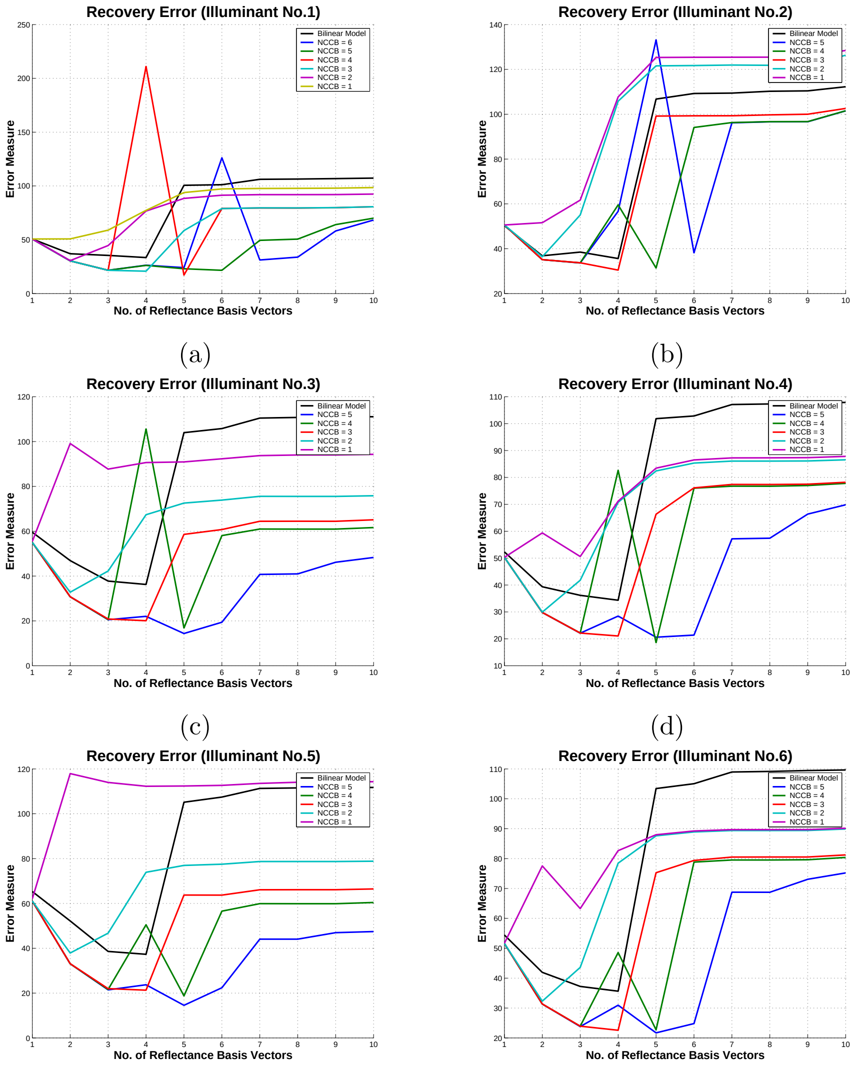
Matrix $\mathbf{K}_i$ corresponds to the particular method of discretization applied in the spectral recovery. On the one hand, if we follow the IFK approach, matrix $\mathbf{K}_i$ is computed using Eq. (4.32) in Section A.5 of Appendix A. On the other hand, if the Bilinear method is applied, $\mathbf{K}_i$ is computed employing Eq. (4.42) in Section 4.8.1. Furthermore, the inverse matrix $\mathbf{K}_i^+$ is computed using the *pseudoinverse* procedure[4], along with the removal of null singular values.

Results corresponding to the spectral recovery experiments are depicted in a series of graphics from Fig. 4.7 to Fig. 4.10. Each of these graphics exhibits the evolution of the error E_i in Eq. (4.60) as a function of the number of basis vectors employed for spanning both the reflectances and the continuous color functions. The latter only affects to the IFK approach, while the former, to both of them.

Precisely, the number of vectors of the reflectance basis are marked in the abscissas, whereas there exists a curve for each group of color basis vectors. NCCB in the legends stands for the *Number of Continuous Color Basis vectors* spanning the continuous color subspaces (IFK model). Both kinds of basis vectors are sorted accordingly to the decreasing magnitude of the corresponding singular values. A maximum of 10 reflectance and 6 color basis vectors are considered, since no more precision is gained by a higher number of components. More specifically, in all of those graphs the results from the Bilinear model are painted in black.

The above set of Figures can be grouped into two pairs, Fig. 4.7 and Fig. 4.8, and Fig. 4.9 and Fig. 4.10. The first pair shows the results using the continu-

[4] If $\mathbf{K} = \mathbf{U}\,\boldsymbol{\Sigma}\,\mathbf{V}^t$, then $\mathbf{K}^+ = \mathbf{V}\,\boldsymbol{\Sigma}^+\,\mathbf{U}^t$, where $\boldsymbol{\Sigma}^+ = diag(\frac{1}{\sigma_1}, \ldots, \frac{1}{\sigma_n})$ so that $\sigma_i \neq 0\,,\ \forall i = 1, \ldots, n$.

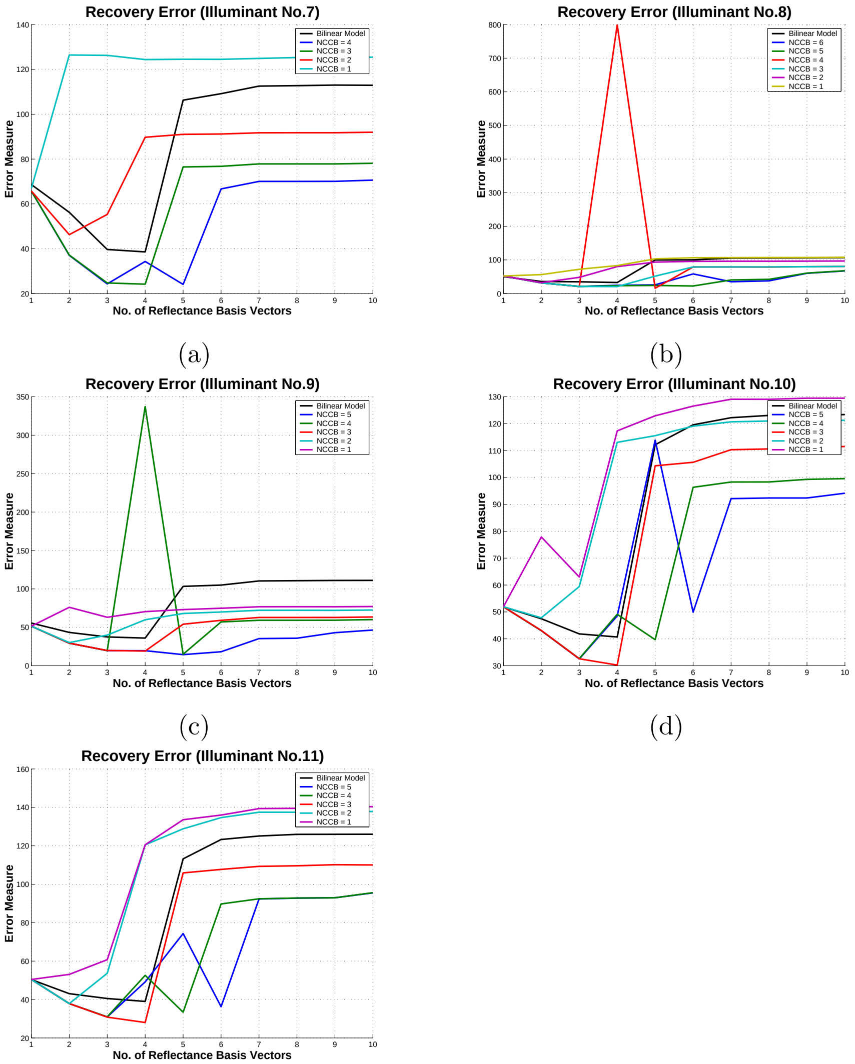


(a) (b)

(c) (d)

(e) (f)

Figure 4.7: Error of the recovery of the reflectance functions employing continuous colors (I). From Illuminant no. 1 – (a) – to Illuminant no. 6 – (f).



(a)

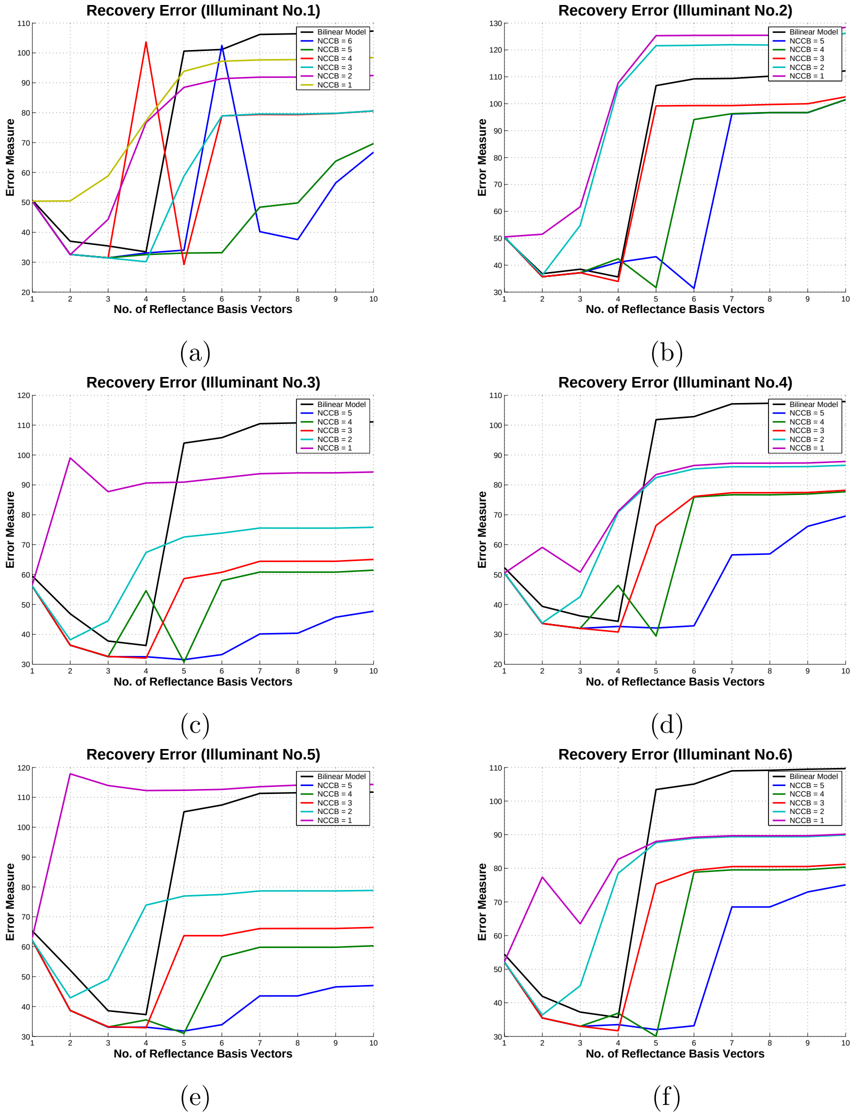
(b)

(c)

(d)

(e)

Figure 4.8: Error of the recovery of the reflectance functions employing continuous colors (II). From Illuminant no. 7 – (a) – to Illuminant no. 11 – (e).

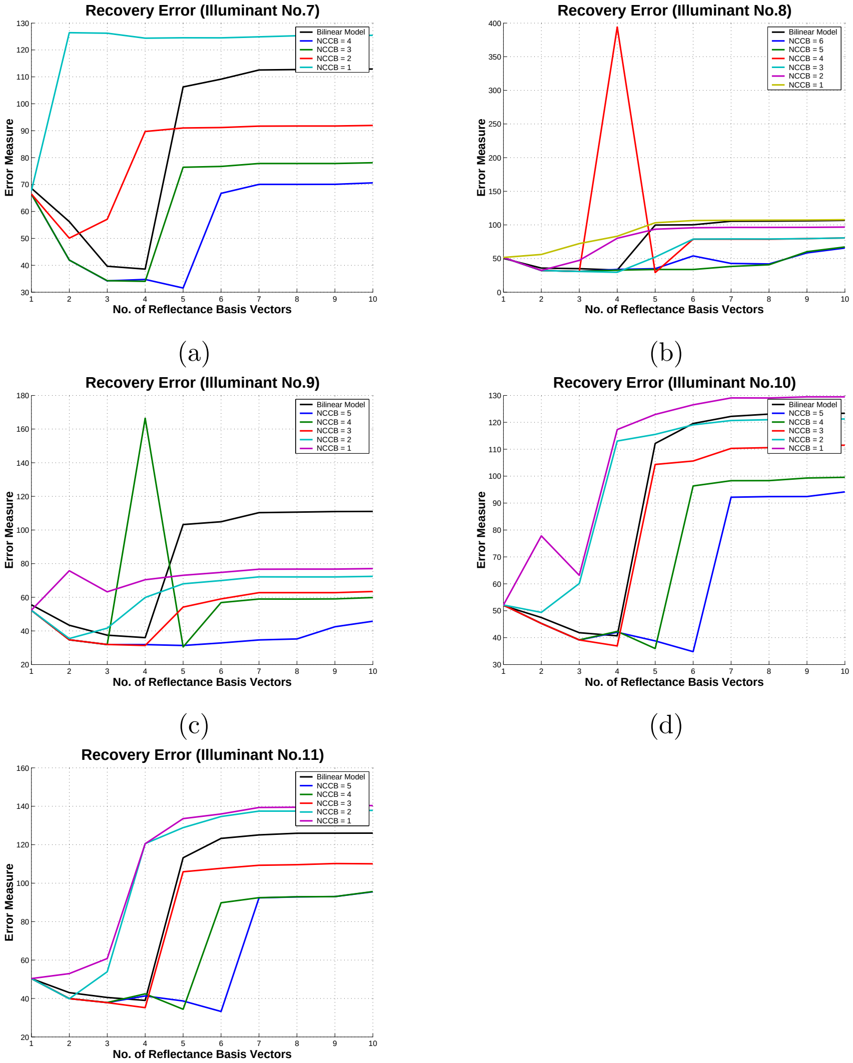


(a) (b)

(c) (d)

(e) (f)

Figure 4.9: Error of the recovery of the reflectance functions employing RGB values (I). From Illuminant no. 1 – (a) – to Illuminant no. 6 – (f).



(a)

(b)

(c)

(d)

(f)

Figure 4.10: Error of the recovery of the reflectance functions employing RGB values (II). From Illuminant no. 7 – (a) – to Illuminant no. 11 – (e).

ous color functions to recover reflectances, whereas the second pair uses RGB coordinates. This only affects to the IFK approach, since the Bilinear model is always expressed in RGB coordinates and merely the number of reflectance basis vectors varies ($1 \div 10$).

As explained in Section 4.7, it is possible to compute a change of basis to transform RGB components onto those continuous color basis. Since the number of components of the continuous color may be higher ($3 \div 6$), eventually there can be a loss of information when using just RGB values. Nevertheless, since RGBs are the color data immediately available from cameras, any method should be able to use this kind of coordinates.

To better understand the former results, in Table 4.1 we put into figures the best errors in regard to the recovery method and the canonic illuminant. The column IFK(C) means that the IFK model was used in conjunction to RGB coordinates, whereas the use of continuous color components is noted as IFK(CC). We also provide the percent error reduction of the IFK approaches in respect to the Bilinear model, as well as the mean through all the illuminants.

As it can be appraised from Fig. 4.7 to Fig. 4.10, results in Table 4.1 were obtained using 5 and 4 basis vectors for the reflectance and continuous color subspaces, respectively, for IFK(C) and most illuminants, except for illuminants n° 2, n° 10, and n° 11, which used 6 and 5 basis vectors each. For IFK(CC), the number of basis vectors were also 5 and 4 for illuminants n° 1, n° 4, n° 7, and n° 8; 5 and 5 for illuminants n° 3, n° 5, n° 6, and n° 9; and 4 and 3 for illuminants n° 2, n° 10, and n° 11. Hence, it seems a good election to use 5 basis vectors for the reflectances and 4 for the continuous color for any further application of the IFK approach in this work. In respect to the Bilinear model, only 4 reflectance basis functions were employed by all the illuminants.

Illum.	**Bilinear**	**IFK (C)**	**% Red.**	**IFK (CC)**	**% Red.**
n° 1	33.45	29.20	12.72	17.17	48.66
n° 2	35.62	31.56	11.40	30.46	14.47
n° 3	36.27	30.72	15.31	14.38	60.34
n° 4	34.36	29.45	14.29	18.59	45.89
n° 5	37.32	31.02	16.88	14.55	61.03
n° 6	35.63	30.06	15.65	21.70	39.10
n° 7	38.57	31.54	18.24	24.08	37.57
n° 8	32.88	29.15	11.35	15.80	51.94
n° 9	36.01	30.45	15.44	14.43	59.92
n° 10	40.69	34.78	14.51	30.25	25.66
n° 11	38.98	33.21	14.79	28.04	28.07
Mean	**36.34**	**31.01**	**14.60**	**20.86**	**42.97**

Table 4.1: Error of the spectral reflectance recovery. *Bilinear*: Bilinear model, *IFK(C)*: IFK using RGB, *IFK(CC)*: IFK using continuous color, *% Red.*: percent error reduction.

Two principal conclusions are drawn from these values. First, the IFK method beats in terms of precision of the recovered reflectances compared with the Bilinear model, for both continuous colors and RGB values. Secondly, precision increases whenever the continuous color components is used instead of the RGB coordinates. This directly follows from the fact that, in general, continuous color handles some more dimensions than RGB values.

Nevertheless, an interesting point derived from Table 4.1 is that, in the continuous case – IFK(CC) –, by using only one (maybe two) more dimensions than in RGB, there is a *mean* error reduction of 42.97%. Besides, even in case the same number of coordinates is used – IFK(C) –, there still exists an appreciable error reduction of 14.60%. That is, the continuous description of color does not mean as a consequence an extremely high number of dimensions, rather a lot of information is gained with the addition of just one or two more dimensions.

Partial loss of information could be overcome by a more suitable technique to identify continuous colors from RGB responses than the least squares fitting of the basis change matrix in Section 4.7. A likely solution would be the addition of one or two more actual sensors. For example, in [Gui04] an *intelligent* system adds *yellow* and *cyan* coordinates to the standard RGB from a classic video signal to impressively increase the number of colors pictured in a TV set. Nevertheless, we are not going into further details on this subject by now.

4.9.4 Color Mapping Computations

As a conclusion of the previous Section, this one is devoted to the explicit computation of color mappings between different lighting conditions in order to evaluate the performance of each one of the physical models previously mentioned in terms of the error of prediction for the color change. To that purpose, we exploit both the IFK and the Bilinear models, as before, along with the two other models that were described in Section 4.8.2 and Section 4.8.3, namely, the Quadrature and Diagonal models.

Additionally, since the correspondence between colors under different illuminations is known *a priori* in our experiments, we can take advantage of it and use a regression approach to compute color maps too. In that case, three kinds of linear applications are checked as likely to explain the color change, i.e., *affine*, *homogeneous*, and *diagonal* applications. Each of them is computed using two different algorithms, a standard linear least squares fitting method and the constrained version of this algorithm by Lawson and Hanson [LH74], which generates nonnegative solutions.

Next, a brief description of the regression schemes employed in the experiments is provided within the context of the linear least squares fitting. Let us suppose that two color sets, $\{\mathbf{x}_i\}_{i=1,\ldots,n}$ and $\{\mathbf{y}_i\}_{i=1,\ldots,n}$, are furnished, each generated under some illumination condition. There is a correspondence between vectors $\mathbf{x}_i$ and $\mathbf{y}_i$

$$\mathbf{x}_i \sim \mathbf{y}_i \Longleftrightarrow \exists\ A \in \mathcal{L}(\mathbb{R}^p) \mid A(\mathbf{x}_i^t) = \mathbf{y}_i^t \tag{4.61}$$

where $\mathcal{L}(\mathbb{R}^p)$ is the set of all linear mappings defined in $\mathbb{R}^p$. These vectors can be put together forming two matrices $\mathbf{X}^t = [\mathbf{x}_1^t, \ldots, \mathbf{x}_n^t]$ and $\mathbf{Y}^t = [\mathbf{y}_1^t, \ldots, \mathbf{y}_n^t]$. Then, our aim is to find a linear transformation mapping set $\mathbf{X}$ onto $\mathbf{Y}$, that is, $A(\mathbf{X}^t) = \mathbf{Y}^t$. What comes next is a description of the three different approaches followed to compute the application A by means of the least squares fitting.

Affine Applications: For any color $\mathbf{x}_i$, a general affine transformation will project it onto another color $\mathbf{y}_i$ in the following way

$$\mathbf{y}_i^t = \mathbf{A}\,\mathbf{x}_i^t + \mathbf{b}^t \tag{4.62}$$

This expression can be more compactly rewritten as

$$\begin{pmatrix} \mathbf{y}_i^t \\ 1 \end{pmatrix} = \left(\begin{array}{c|c} \mathbf{A} & \mathbf{b}^t \\ \hline \mathbf{0} & 1 \end{array}\right) \begin{pmatrix} \mathbf{x}_i^t \\ 1 \end{pmatrix} \tag{4.63}$$

where $\mathbf{0} = [0, \overset{n}{\ldots}, 0]$. If all the colors in $\mathbf{X}$ and $\mathbf{Y}$ are considered at the same time, the previous expression turns into the next one

$$\left(\begin{array}{c} \mathbf{Y}^t \\ \hline \mathbf{1} \end{array}\right) = \left(\begin{array}{c|c} \mathbf{A} & \mathbf{b}^t \\ \hline \mathbf{0} & 1 \end{array}\right) \left(\begin{array}{c} \mathbf{X}^t \\ \hline \mathbf{1} \end{array}\right) \tag{4.64}$$

where $\mathbf{1} = [1, \overset{n}{\ldots}, 1]$. This linear system can be solved in terms of its *normal equations*

$$\left(\begin{array}{c|c} \mathbf{A} & \mathbf{b}^t \\ \hline \mathbf{0} & 1 \end{array}\right) = \left(\begin{array}{c|c} \mathbf{Y}^t \cdot \mathbf{X} & \mathbf{Y}^t \cdot \mathbf{1}^t \\ \hline \mathbf{1} \cdot \mathbf{Y} & \mathbf{1} \cdot \mathbf{1}^t \end{array}\right) \left(\begin{array}{c|c} \mathbf{X}^t \cdot \mathbf{X} & \mathbf{X}^t \cdot \mathbf{1}^t \\ \hline \mathbf{1} \cdot \mathbf{X} & \mathbf{1} \cdot \mathbf{1}^t \end{array}\right)^{-1} \tag{4.65}$$

Nevertheless, since normal equations are generally ill–conditioned, it would be better to employ the SVD decomposition in the computation of the pseudoinverse of the system matrix.

Homogeneous Applications: The previous general linear application can be simplified to a homogeneous one, that is, $\mathbf{Y}^t = \mathbf{A}\,\mathbf{X}^t$. Therefore, the resultant color mapping is computed as

$$\mathbf{A} = \left(\mathbf{Y}^t\,\mathbf{X}\right)\,\left(\mathbf{X}^t\,\mathbf{X}\right)^{-1} \tag{4.66}$$

Diagonal Applications: If our intuition or the facts drive us to think that the map is likely to have tiny or null values off the diagonal components, it might be advisable even reducing the corresponding structure of the linear application into a diagonal one. Hence, the diagonal elements a_k are calculated as

$$a_k = \frac{\sum_{i=1}^n y_{ik} x_{ik}}{\sum_{i=1}^n x_{ik}^2}\,,\ k = 1, \ldots, p \tag{4.67}$$

where p is the number of elements in the diagonal, i.e., the number of color components. Therefore, $\mathbf{A} = diag(a_1, \ldots, a_p)$.

The set of experiments considered hereafter tries to establish the performance of all of the above mentioned computational schemes. Performance is measured as the total error in predicting the set of colors under a canonic light and is computed, for the i^{th} canonic illuminant, as the total amount of distances between actual colors and transformed ones

$$E_i = \sum_{j=1}^{m} \sum_{k=1}^{n} \|\mathbf{y}_k - \tilde{\mathbf{y}}_k\| = \sum_{j=1}^{m} \sum_{k=1}^{n} \|\mathbf{y}_k^t - \mathbf{K}_{ij}\,\mathbf{x}_k^t\|\,,\ i = 1, \ldots, l \tag{4.68}$$

where $l = 11$ is the number of canonic lights, $m = 124$ is the total number of illuminants, and there are $n = 3989$ different surfaces (colors). Matrix $\mathbf{K}_{ij}$ is the color mapping transforming colors under the j^{th} illuminant into those under the i^{th} canonic light.

The resulting errors for each of the above schemes are depicted in Fig. 4.11 and Fig. 4.12. The first group of graphics in Fig. 4.11 shows the results belonging to the linear regression, whereas the second one exhibits those belonging to the physical models. Each Figure consists of a set of vertical lines in red, spanning from the *minimum* (▼) to the *maximum* (▲), with additional marks for the *mean* (●) and the *median* ($*$) values of the distribution of errors computed for each canonic illuminant (numbers $1 \div 11$ in the abscissas). There is also a black line showing the performance attained for the whole set of data by each scheme. These graphics succinctly illustrate the way error values distribute. Notice that ordinate scales are the same in all plots but those in Fig. 4.12(c) and Fig. 4.12(d), which are significantly greater, specially the latter.

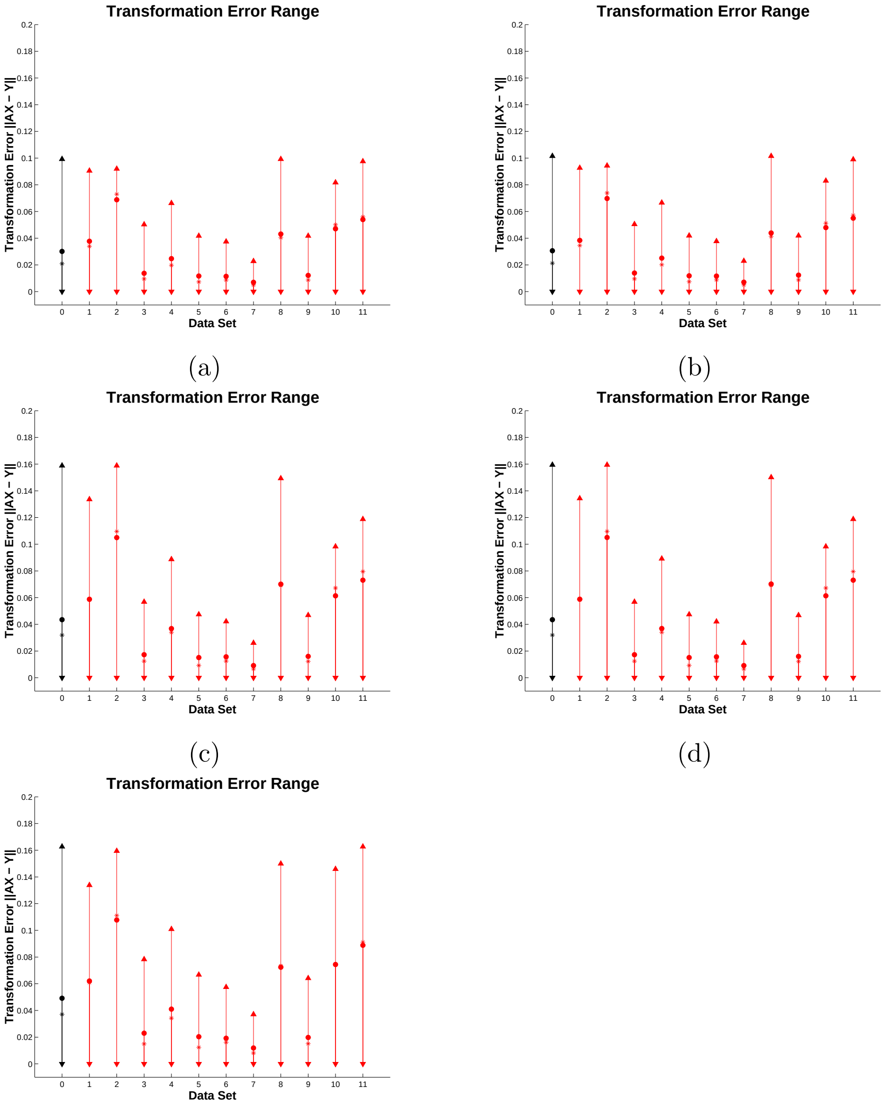
In order to clarify the results obtained, in Table 4.2 and Table 4.3 we also collected the mean values for each scheme and canonic illumination. As before, Table 4.2 encompasses the results belonging to the least squares fitting and Table 4.3, those attained using the physical models. The bottom row of these tables displays the mean values computed through the 11 canonic illuminants.

Let us now consider the results obtained specifically from the linear regression. The idea in carrying out this experiment was to guess which is the best linear model to be used as a general color map, and if there exists any further advantage in constraining the applications to only have nonnegative coefficients[5]. The conclusion that can be drawn from the results in Fig. 4.11 and Table 4.2 is that the difference between using an affine transformation in front of a simple homogeneous model is very small.

Moreover, when the nonnegative constraint is put, the difference between affine and homogeneous transformations is almost null. In that case, from simple eye inspection on the resulting maps, we checked that mappings were almost completely diagonal. That is why the diagonal scheme generates nearly the same error than the nonnegative ones, either affine or homogeneous. Thus, despite other linear models better predict the color change, it is a rather good approximation to apply just a simple diagonal mapping when sensors are of a similar class to those employed here. Secondly, physical models are not all of them equally helpful in regard to their error results. Both the IFK and the Bilinear models behave far better than the Quadrature and the Diagonal ones. This is a fair aftermath of the particular discretizing rules applied to derive these models, specially the latter.

The reader should be aware that in Table 4.2 and Table 4.3 two approximations were used that are equally named as *Diagonal*. They correspond to the physical model proposed in Section 4.8.3 and to the regression scheme introduced earlier in this Section. The former performs pretty badly, whereas the latter comparatively does better. There is no contradiction here because, despite the two approaches share a common diagonal structure, the hypotheses applied to compute the coefficients are essentially different.

[5] The nonnegativity constraint comes from physical considerations about the nature of all the functions involved in color in order to be physically feasible.



(a)

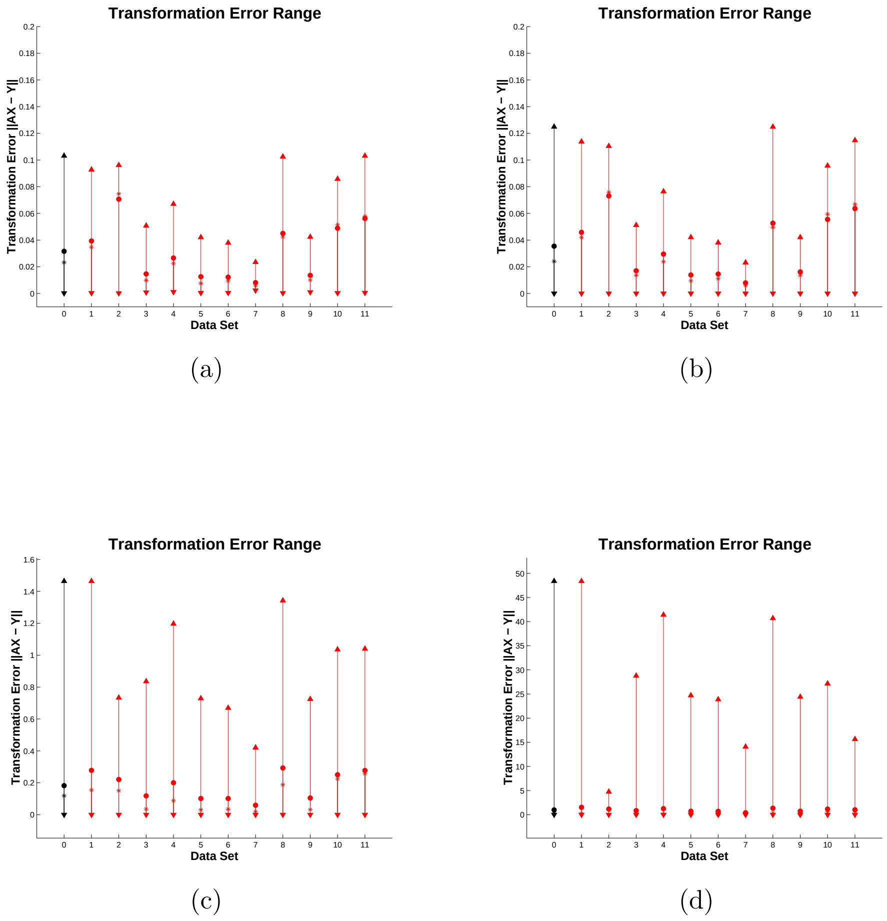
(b)

(c)

(d)

(e)

Figure 4.11: Error of the color transformations using least squares fitting: (a) Affine model, (b) Homogeneous model, (c) Nonnegative affine model, (d) Nonnegative homogeneous model, and (e) Diagonal model.



(a)

(b)

(c)

(d)

Figure 4.12: Error of the color transformations using explicit physical models: (a) IFK model, (b) Bilinear model, (c) Quadrature model, and (d) Diagonal model.

	Pseudoinverse			Nonnegative	
Illum.	**Affine**	**Homogen.**	**Diagonal**	**Affine**	**Homogen.**
no. 1	0.0377	0.0384	0.0620	0.0588	0.0588
no. 2	0.0688	0.0698	0.1077	0.1050	0.1050
no. 3	0.0137	0.0139	0.0230	0.0172	0.0172
no. 4	0.0247	0.0251	0.0410	0.0368	0.0368
no. 5	0.0117	0.0118	0.0203	0.0151	0.0151
no. 6	0.0114	0.0116	0.0192	0.0156	0.0156
no. 7	0.0697	0.0070	0.0119	0.0091	0.0091
no. 8	0.0431	0.0439	0.0724	0.0701	0.0701
no. 9	0.0121	0.0123	0.0198	0.0160	0.0160
no. 10	0.0470	0.0480	0.0744	0.0614	0.0614
no. 11	0.0540	0.0550	0.0888	0.0731	0.0731
Mean	**0.0301**	**0.0306**	**0.0491**	**0.0435**	**0.0435**

Table 4.2: Error of the color transformations using least squares fitting.

Illum.	**IFK**	**Bilinear**	**Quadrat.**	**Diagonal**
no. 1	0.0393	0.0458	0.2782	1.5273
no. 2	0.0707	0.0731	0.2206	1.1692
no. 3	0.0147	0.0170	0.1179	0.8467
no. 4	0.0266	0.0295	0.2004	1.2561
no. 5	0.0126	0.0138	0.1014	0.7116
no. 6	0.0122	0.0146	0.1015	0.7092
no. 7	0.0081	0.0080	0.0590	0.3992
no. 8	0.0450	0.0527	0.2929	1.3514
no. 9	0.0135	0.0162	0.1047	0.7271
no. 10	0.0489	0.0555	0.2506	1.1683
no. 11	0.0562	0.0636	0.2774	1.0198
Mean	**0.0316**	**0.0354**	**0.1822**	**0.9896**

Table 4.3: Error of the color transformations using physical models.

Changing out interest to another question, we realize that a seemingly surprising result is obtained if the errors of the IFK and Bilinear models are compared, i.e., both results are nearly the same. However, in the previous Section we showed not only that the IFK approach outperformed the Bilinear one, especially when the continuous color was used, but also that this was true in RGB coordinates. In our opinion, there is one main reason for obtaining such similar results now.

As we showed in Section 4.8.1, the Bilinear model was derived as a discretization of the continuous IFK equation, and appeared to be very similar to the one obtained along with the IFK approach once the color coordinates were fixed to be RGB in both sides of the color mapping, though no finite–dimensional approximation was taken for the illumination.

Therefore, despite the IFK approach attained better results than the Bilinear model in spectral recovery, as a consequence of projecting the recovered function again onto their RGB values to get a color mapping, any additional information that might be brought by the IFK model and its continuous color is lost in the way, giving rise both approaches to very close color mappings consequently. Thus, it seems as if no room is left to obtain more exact color mappings unless more coordinates are used.

Nonetheless, the precision of IFK model (0.0316) is almost as good as the best one attained by the least squares fitting (0.0301), which is the best result possible in terms of error of prediction, outperforming both the nonnegative and the diagonal schemes, as can be appreciated in Table 4.2 and Table 4.3.

4.10 Conclusions

In this Chapter a continuous framework is suggested analytically tackling the color change issue as a natural continuous generalization of the usual color formation equations. This model is able to describe the formation of multispectral color signals by means of a unique expression which is particularized as needed in order to attain handier discrete formulations. As a result, the continuous color formation equation is identified as a Fredholm's integral equation of the first kind (IFK). Based on the description of the IFK equations in Appendix A, several numerical schemes are provided so as to undergo practical computations related with the recovery of spectral functions and the calculation of color maps.

Some conclusions can be derived from the work accomplished hitherto. First, the IFK model is a general model which contributes to the analytical study of the solutions to the color change problem, its existence, uniqueness, and behavior. Second, constraints to be fulfilled by the solution and the data subspaces in order a solution to the problem exists are established. This is new since that issue has always been approached from an *ad hoc* point of view, materialized into a discrete model, as one of those described above. Moreover, we explicitly show the structure of the solution function, as well as translate the continuous general expression into a numerical approach. Third, we relate several former well–established color constancy models to the one suggested by means of a set of discretization tools extensively detailed in Appendix A. In addition, it is clearly shown how those models are derived from an IFK and which of them are allegedly likelier to furnish better results in describing the color change problem.

After the theoretical study, some results are provided to support the statements done before. First, the problem of generalizing a discrete sensor is solved. Without this step, the IFK model would be a pure discrete model which could not provide any extra information based on the continuity of the sensor function. Once such a sensor is obtained, the spectral recovery is posteriorly investigated, where it is shown that the IFK model in general provides better results than the Bilinear model, both in the continuous color subspace and within the more restricted case of RGB coordinates, despite the fact that in the latter the improvement is not as great as in the former. This makes the IFK model an optimum candidate to be used in deeper recovery schemes, which are not our main goal in this work. Furthermore, by computing the combination of basis that provides the best results, we state that, by the addition of only one more sensor, it could be possible to recover pretty good approximations of the reflectance functions and the amount of such improvement is display.

Thereafter, the explicit computation of color mappings using physical models is as well attempted. We apply all the models described so far, as well as a linear fitting scheme to determine two different things. First, which linear model is the best at explaining the color change and how much the error of each of them is. Secondly, we compare the physical models with the results obtained by regression in order to determine which models do better. Results confirm that almost there is no difference between a general linear (affine) transformation and a homogeneous one, principally when the coefficients of the applications are constrained to be positive. Additionally, the error made by a diagonal application is taken into account, which turns out to be of the same order as the error made by the rest of models. These facts confirm that a linear transformation suffices to model color changes and that a diagonal map can be taken as a simplification of the general linear model with only a marginal loss in precision. Despite those are not new results, they are proven here for a greater number of surfaces and illuminations.

Finally, it is shown that the IFK and the Bilinear models almost provide as good results as regression, whereas the Quadrature and the Diagonal models perform poorly. Hence, it seems clear that both the IFK and the Bilinear models closely predict the variation of colors due to light changes, while Quadrature and Diagonal discretizations are too weak for the same purpose, especially the latter. In particular, both the IFK and the Bilinear models present very similar error values, which suggests that in case a transformation in RGB coordinates is computed, both schemes basically converge. This is because both sides of those mappings are equally discretized to be RGB coordinates and no further information can be comparatively drawn from the continuous color description of the IFK model.