Spectral estimation theory: beyond linear but before Bayesian

Jeffrey M. DiCarlo and Brian A. Wandell

Department of Electrical Engineering, Stanford University, Stanford, California 94305

Received September 30, 2002; revised manuscript received February 28, 2003; accepted March 5, 2003

Most color-acquisition devices capture spectral signals by acquiring only three samples, critically undersampling the spectral information. We analyze the problem of estimating high-dimensional spectral signals from low-dimensional device responses. We begin with the theory and geometry of linear estimation methods. These methods use linear models to characterize the likely input signals and reduce the number of estimation parameters. Next, we introduce two submanifold estimation methods. These methods are based on the observation that for many data sets the deviation between the signal and the linear estimate is systematic; the methods incorporate knowledge of these systematic deviations to improve upon linear estimation methods. We describe the geometric intuition of these methods and evaluate the submanifold method on hyperspectral image data.

OCIS codes: 330.169, 330.1710.

1. INTRODUCTION

Most color-acquisition architectures obtain a small number of spectral samples to estimate a high-dimensional spectral signal. This estimation is imprecise because the input signal is critically undersampled; there are no guarantees that the estimate will be correct. Yet as we all experience, both with our own eyes and by the success of the still and video camera industries, it is possible to capture enough information about the input signal to create a reasonable facsimile.

Much of the success can be explained by the fact that the final receiver of the signal, the human visual system, samples the input only coarsely by using the three types of cone photoreceptors. This is not a full explanation of why color cameras, whose sensors do not sample the spectral signals in the same way as the human eye, provide enough information to create reasonable reproductions. A second reason for the success is found in the input signals: The reflectance functions of surfaces and spectral power distributions of natural illuminants contain a great deal of structure. Daylights, for example, appear to be well characterized by no more than three independent components.[1] Many natural surfaces are described accurately by using linear parameterizations of fewer than eight terms.[2,3]

The significant structure in the input signals motivated the creation of several computational theories of color estimation based on linear methods. Investigators from engineering, neuroscience, and psychology developed linear computational methods that were designed to take advantage of the structure in natural scenes.[4–7] These were followed by increasingly complex algorithms designed to incorporate further knowledge about the signals, such as the nonnegativity of surface reflectance functions[8] or the likely distribution of reflectance values.[9] These theoretical developments have also been extended into practical procedures for illuminant and surface estimation in engineering applications.[10–12]

This paper introduces a method for estimating high-dimensional spectral signals from low-dimensional sensor responses. The spectral estimation method extends the current dominant paradigm, based on linear models. Specifically, we follow the general insight introduced by Brainard and Freeman, who argued that the best models will be derived from understanding the distribution of the input signals and then using a Bayesian approach for the estimation method.[9] It is commonly understood that a complete Bayesian approach can lead to unwieldy computations. Here we develop a compromise between the general Bayesian approach and the much simpler linear estimation methods that we think can be of practical use.

We begin in Section 2 by introducing the notation used throughout the remainder of the paper, and we develop the mathematical theory behind linear estimation. In Section 3 we develop a new estimation method, which we refer to as submanifold estimation. In Section 4 we evaluate the estimation method by using hyperspectral data. Finally, in Section 5 we discuss the limitations of the method, practical implementation issues, and related work.

2. BACKGROUND

A. Notation

There is a simple linear relationship between the input signal (light spectral power distribution) and the sensor response (sensor voltage) in most electronic image-acquisition technologies [charge-coupled devices, photomultipliers, or complementary metal-oxide semiconductor photodetectors]. This linear relation simplifies spectral estimation algorithms and justifies representing physical elements of the scene and imager as vectors or matrices.

In this paper we represent spectral signals as vectors, and we combine two or more spectral signals into the columns of a matrix. The dimension of these vectors is the number of wavelength samples, N_λ, and for any applica-

1084-7529/2003/071261-10$15.00

tion we assume that this value is chosen so that the spectral information is represented with essentially no loss of information. The choice of spacing between the wavelength samples, $\Delta\lambda$, depends on the usual considerations that apply when converting continuous signals to discrete signals.

Using this matrix notation, the relationship between spectral input signals and sensor responses can be expressed compactly. Suppose that $\mathbf{s}$ represents a scene surface reflectance function; $\mathbf{e}$ represents the scene illuminant spectral power distribution; diag($\mathbf{x}$) represents a square matrix with $\mathbf{x}$ along the diagonal and all other elements are zero; a matrix $\mathbf{T}$ represents the imager sensor responsivities, where each column of the matrix is a different sensor responsivity function; and α is a scale factor. Then the sensor response, $\mathbf{r}$, is simply

$$\mathbf{r} = \alpha \mathbf{T}^t \,\mathrm{diag}(\mathbf{e})\mathbf{s}\Delta\lambda. \tag{1}$$

Equation (1) is a brief description that emphasizes the flow of spectral information but masks the role of several important factors that influence the relationship between the sensor response and the scene characteristics. For example, the orientation of the surface, the lens aperture and focal length, exposure duration, and other factors combine to influence the final sensor values. But for a fixed imaging condition, all these factors can be encapsulated into the scale factor, α. The separate factors must be made explicit and accounted for in a full simulator.[13,14]

Because Eq. (1) is bilinear with respect to the surface and illuminant, the role of the illuminant and surface can be switched. Throughout this paper we assume that the illumination is known; this greatly simplifies the language and is the most common application. Given this assumption, it is convenient to group the sensor responsivity matrix and the illuminant into one quantity, which we refer to as the system sensor matrix [$\mathbf{T_e} = \alpha\,\mathrm{diag}(\mathbf{e})\mathbf{T}\Delta\lambda$]. Hence Eq. (1) becomes

$$\mathbf{r} = \mathbf{T}_\mathbf{e}^t\mathbf{s}. \tag{2}$$

Because we review the basic structure of several common linear estimation problems and introduce submanifold estimation structure, geometry, and algorithm, we suppress consideration of the effects of noise and sensor properties to simplify the equations and clarify the principles. In subsequent empirical papers on estimation,[15] we will include these noise properties. The basic insights described here remain unchanged, although, when we include the effects of noise, the formulas become somewhat more complex.

For a fixed illuminant, we can treat the surface reflectance function as the input signal. The ability to estimate the signal from the responses is limited by the fact that there are a small number of sensors (typically, three), while the signal is a high-dimensional function of wavelength. Given this situation, it is always possible to find many signals that predict the measured response; hence it is reasonable to consider only methods in which the estimated signal predicts the measured response.[16] Consequently, the methods differ only in their estimate of the signal components that are invisible to the sensors (i.e., the null space of the sensor matrix). Estimation methods that use better information about the structural properties of the signal will produce better estimates.

Figure 1 illustrates the graphical relationship among the input signals (gray circles), the system sensor responsivity function (solid black vector), and the estimation rules. For simplicity, we assume that the signals contain energy at only two wavelengths so they can be represented as points on the plane. Furthermore, we assume that we have a monochrome sensor so that the number of system sensors is less than the dimension of the surface reflectance functions. The slope of the system sensor vector is the relative sensor responsivity to the two wavelengths.

For measurement, the sensor response to a surface reflectance, $\mathbf{s}$, is computed by dropping a perpendicular (dashed black line) from the reflectance function to the sensor arrow. The sensor response, $\mathbf{r}$, is proportional to the distance from the origin to the point on the sensor vector. From Fig. 1, we see that the sensor response could be produced by any signal along this perpendicular line. The requirement that an estimation method must explain the measured response implies that the reflectance estimate must fall on the perpendicular.

Estimation methods are influenced by two factors: (1) the error metric relating the estimated signal to the measured signal and (2) the knowledge about the likely values of the signal. We will fix the error metric and vary our assumptions about the signal in this paper. The error metric we will use is the sum of squared error or, equivalently, the L_2 error:

$$E_{\mathrm{sse}} = \sum_{i=1}^{N_s} \|\mathbf{s}_i - \hat{\mathbf{s}}_i\|^2. \tag{3}$$

Here $\mathbf{s}_i$ is a measured reflectance, $\hat{\mathbf{s}}_i$ is an estimated reflectance, and N_s is the number of estimated reflectance functions. We focus on this metric because it is used heavily throughout the estimation literature and because of its geometric properties: The error between estimated and measured signals is simply the squared Euclidean distance between two samples.

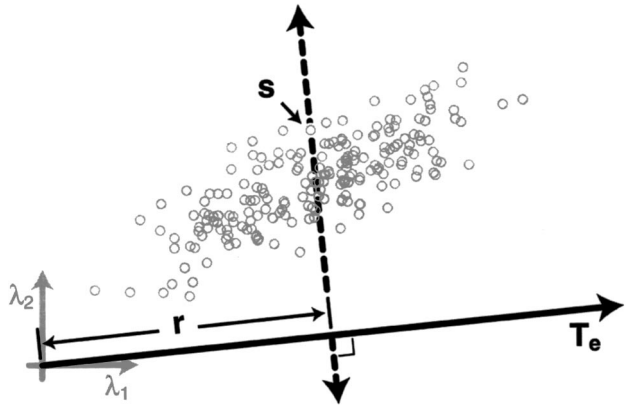

Fig. 1. Sample measurement and estimation. The gray circles represent sample reflectance functions. The black arrow represents a sensor responsivity function, $\mathbf{T_e}$. A sample, $\mathbf{s}$, when measured produces a sensor response, $\mathbf{r}$. An estimate of $\mathbf{s}$ from $\mathbf{r}$ may fall anywhere in the null space of the sensors, denoted by the dashed vector.

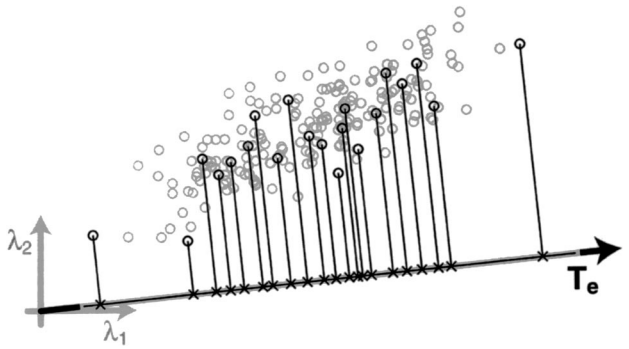

Fig. 2. Pseudoinverse estimation. The estimation function of the pseudoinverse method, the thin black line, is embedded in the sensor vector. The estimates for a subset of samples are denoted by the black ×'s. The estimation error for a sample is the length of the black line connecting the estimate with the sample. Other details as in Fig. 1.

B. Linear Estimation

Nearly all spectral estimation algorithms have been based on linear estimation methods.[3–7,17–22] Specifically, if $\mathbf{r}$ is a measured sensor response, linear methods seek a matrix $\mathbf{\Gamma}$ and a vector $\boldsymbol{\gamma}$ that estimates the reflectance function ($\hat{\mathbf{s}}$) by minimizing the error in Eq. (3):

$$\hat{\mathbf{s}} = \boldsymbol{\gamma} + \mathbf{\Gamma}\mathbf{r}. \tag{4}$$

Because we have fixed the error metric, the linear matrix, $\mathbf{\Gamma}$, and vector, $\boldsymbol{\gamma}$, are determined only by our assumptions about the measured reflectances. The more we know about the likely values of $\mathbf{s}$, the better we can construct $\mathbf{\Gamma}$ and $\boldsymbol{\gamma}$.

The first assumption is that nothing is known about the reflectance functions; every relative reflectance function is possible and equally likely. This implies that the covariance matrix of $\mathbf{s}$ is a scaled version of the identity matrix and the mean of $\mathbf{s}$ is zero.[23,24] On the basis of this assumption, the linear estimator matrix is the pseudoinverse of the sensor responses, and the fixed vector is derived from this matrix, the mean of the surface reflectance functions, $\bar{\mathbf{s}}$, and the system sensor responsivities, $\mathbf{T}_e$[16]:

$$\mathbf{\Gamma} = \mathbf{T}_e(\mathbf{T}_e^t\mathbf{T}_e)^{-1}, \tag{5a}$$

$$\boldsymbol{\gamma} = \bar{\mathbf{s}} - \mathbf{\Gamma}\mathbf{T}_e^t\bar{\mathbf{s}}. \tag{5b}$$

Figure 2 illustrates the geometric consequences of such an estimator. The figure is identical to Fig. 1 with the exception that the estimation function and estimation errors have been added. The gray circles represent the reflectance functions at two different wavelengths. The black line, which resides on the system sensor vector, denotes the estimation function, and the black ×'s represent a subset of reflectance estimates. Additional black lines connect reflectance functions to their corresponding estimates. Hence the squared length of each of these black lines represents the estimation error for a particular reflectance function.

The next important assumption is that the reflectance functions reside in a lower-dimensional subspace.[2,3,25,26] Principal components analysis can be used to confirm this assumption. When the reflectance data are limited to a subspace, linear estimation methods constrain the estimates to fall within that subspace. The signal covariance matrix, which is used to derive the lower-dimensional subspace, is used to derive $\mathbf{\Gamma}$ and $\boldsymbol{\gamma}$. If we suppose that $\mathbf{\Sigma}_s$ is the covariance matrix of reflectance functions obtained from a training data set, then the best linear estimator matrix is[16]

$$\mathbf{\Gamma} = \mathbf{\Sigma}_s\mathbf{T}_e(\mathbf{T}_e^t\mathbf{\Sigma}_s\mathbf{T}_e)^{-1}. \tag{6}$$

The fixed vector, $\boldsymbol{\gamma}$, is derived from the linear estimator matrix in the same way ($\boldsymbol{\gamma} = \bar{\mathbf{s}} - \mathbf{\Gamma}\mathbf{T}_e\bar{\mathbf{s}}$).

Figure 3 shows the estimation function based on knowing the mean and covariance matrix of the reflectance signals used in the training data. It is identical to Fig. 2 except that the estimation function is shifted to the space where the reflectance data reside instead of where the system sensors reside. This shift reduces the estimation error, which can be seen by comparing the estimation errors (lengths of the black lines connecting the gray circles with the black x's) with those in Fig. 2.

Some estimation methods restrict the estimated values to a subspace, but they do not compute the mean and covariance matrix of the signals directly. Instead, they indirectly compute these statistics by assuming a signal distribution. For example, the method of maximum ignorance with positivity assumes the surface reflectance data is uniformly distributed between zero and one.[27] The smoothest constraint method extends the positivity method by assuming near wavelengths are linearly correlated with each other.[28] The more closely these assumptions approximate the signal distribution, the better the assumed mean and covariance matrix and hence the better the resulting estimation matrix. Assuming a signal distribution is a good idea when representative data are not available; however, if data are available, most likely it is better to compute the statistics from the measured data than to assume a distribution.

Linear estimation based on the L_2 metric is optimal when the signal data set conforms to a normal distribution.[9,24,29] However, when the data deviate from normal, we can improve on the linear estimation methods.[9,22,30,31] Figure 4 shows reflectance data that do not follow a normal distribution. Each gray circle denotes a reflectance function, and the black line shows the best linear estimate surface. Additional black lines connect estimates of some reflectance functions with their

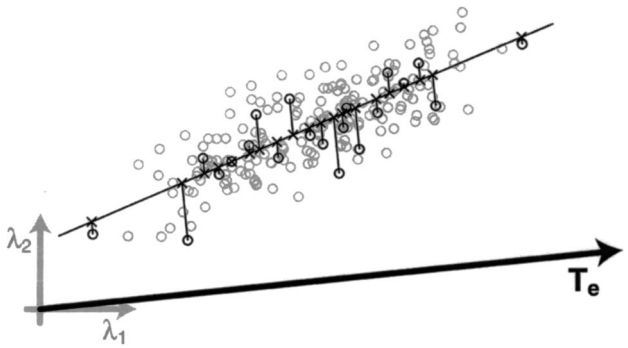

Fig. 3. Linear model estimation. The estimation function of the linear model method, the thin black line, is embedded in the data. Other details as in Fig. 1.

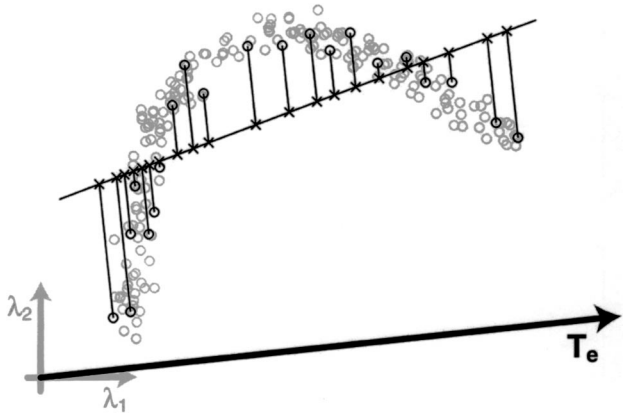

Fig. 4. Linear estimation limitation. The reflectance samples have a nonlinear relationship. The estimation function of the best linear estimator, the thin black line, cannot follow the data. Other details as in Fig. 1.

corresponding measured reflectance functions, and the squared lengths of each line equal the estimation error for the particular sample.

Even though the estimation function is contained within the data subspace, large estimation errors occur because of the nonlinear structure in the data. This is a fundamental limitation of linear estimation. Submanifold estimation is designed to clarify and describe how we can overcome this limitation.

3. SUBMANIFOLD ESTIMATION

In this section we introduce a method to estimate nonlinear structure in spectral data. The method overcomes a major limitation of linear estimation by extending the estimation function from a hyperplane to a more general surface. Specifically, the method finds certain nonlinear structure in the data and uses this structure to improve the spectral estimation.

As an intuition for the method, consider a physical phenomenon that depends on a small number of parameters that are nonlinearly related to the measured values. For example, the reflectance function of human skin depends on a series of nonlinear interactions between three pigments.[32] The spectral properties of the combination of the three pigments require a (global) linear model description of skin surface reflectances with more than three dimensions. Yet locally, the surface reflectance functions can be described by using only a three-dimensional linear model.

The procedure of using a small number of local measurements to predict the remaining measurements is very similar to Poincaré's definition of a submanifold. In his paper "Analysis situs," Poincaré[33] defines a submanifold of $\mathbf{R}^N$ when the values of a point in $\mathbf{R}^k$ are sufficient to predict the values in $\mathbf{R}^{N-k}$.[34] In addition, the mapping itself should vary smoothly. Hence we refer to the method as submanifold estimation. The number of submanifold dimensions, k, is referred to as the intrinsic, or cover, dimensionality. Estimation methods benefit from the knowledge that the intrinsic dimensionality of the spectral data is small (i.e., $k \ll N$) because then fewer parameters need to be estimated. When fewer estimation parameters are needed, the signal can usually be estimated by using fewer sensors.[30]

Submanifold methods can be applied to imaging conditions when we have *a priori* knowledge of the likely properties of the image contents. This knowledge could be represented in the form of either a database of surfaces or a mathematical formula. For example, the Macbeth Color Checker is a chart that comprises 24 different patches, including important and challenging surfaces for imaging applications.[35] A second example is the American National Standards Institute IT 8.7/2 surface reflectance chart. These charts are manufactured to capture the reflectance functions intrinsic to specific print-process outputs. This information contains the *a priori* information that is necessary for submanifold methods.

We describe two versions of the submanifold estimation method. The first method applies to the situation when the absolute spectral power distribution of the illuminant is known. We refer to this method as absolute-scale estimation because scale information can be used in the estimation process. Second, we describe a method that applies when only the *relative* spectral power distribution of the illuminant is known; the absolute level is unknown. We refer to this method as relative-scale estimation. Under these conditions, a solution, $\hat{\mathbf{s}}$, represents the entire family of solutions, $\alpha\hat{\mathbf{s}}$, where α is an unknown scale factor.

A. Absolute-Scale Estimation

In certain imaging applications the lighting is controlled and the intensity level is known, e.g., imaging with an optical scanner or imaging artwork with a camera. Absolute-scale estimation is designed for these imaging conditions. The intuition of absolute-scale estimation flows from examining the limitation of the linear estimation method (Fig. 4). The best linear estimates fall on a line; however, the true data systemically deviate from the line. If we use *a priori* information to compute the likely deviation from the line, say, by tabulating the expected deviations, then we can correct the linear estimate by adding in these deviations.

We can express the absolute-scale estimation algorithm by an equation that includes both the linear estimation and a nonlinear function that records the expected deviation. Suppose $\mathbf{\Gamma}_A$ is the linear estimator matrix, $\boldsymbol{\gamma}_A$ is a fixed vector, and $\boldsymbol{\delta}_A(\mathbf{r})$ is a nonlinear vector-valued function of $\mathbf{r}$ that stores the expected deviation. The subscript A is used to denote the absolute-scale algorithm. Then we can extend Eq. (4) to write a new estimation equation:

$$\hat{\mathbf{s}} = \boldsymbol{\gamma}_A + \mathbf{\Gamma}_A\mathbf{r} + \boldsymbol{\delta}_A(\mathbf{r}). \tag{7}$$

Figure 5 shows an example of absolute-scale submanifold estimation. The gray circles represent the reflectance data as in Fig. 4. The curve represents the submanifold estimation function, where all the estimates will fall. Combining the linear estimate [first two terms in Eq. (7)] with the expected deviations [summarized by the function $\boldsymbol{\delta}_A(\mathbf{r})$ in Eq. (7)] creates the curve. When the data form a compact set that cluster near the typical deviation from the linear estimate, the submanifold procedure improves the estimation accuracy substantially. The short black lines represent the measurement error in the submani-

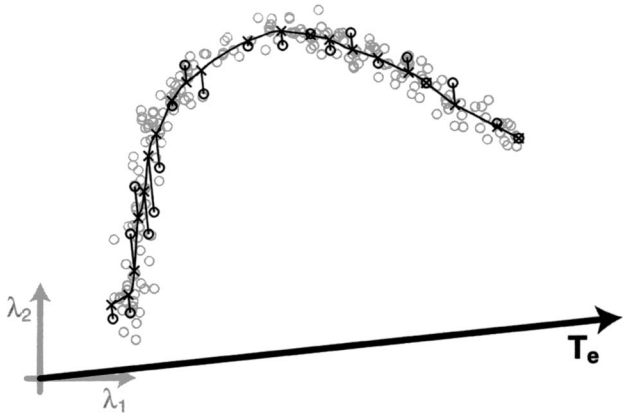

Fig. 5. Submanifold absolute-scale estimation example. The estimation function of the submanifold method, the thin black curve, is embedded in the data even with the nonlinear relationship among the samples. Other details as in Fig. 1.

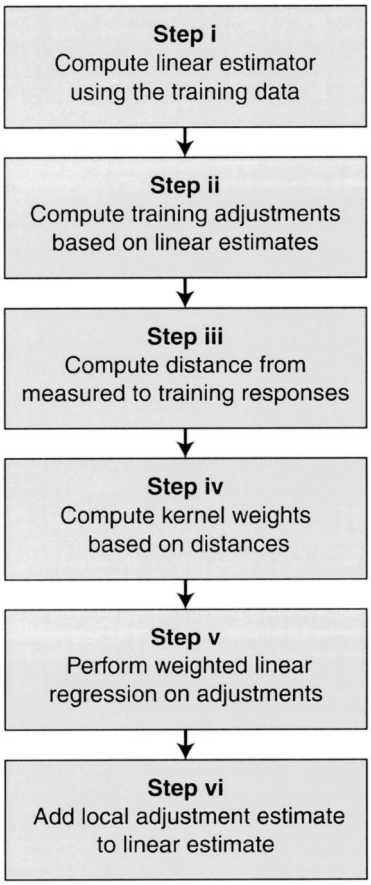

Fig. 6. Submanifold estimation flow-chart overview. See text for details.

fold procedure; the line lengths are substantially shorter than the corresponding lines in Fig. 4.

The flow diagram for absolute-scale submanifold estimation is summarized in Fig. 6. First, the linear estimator matrix, $\mathbf{\Gamma}_A$, and the fixed vector, $\boldsymbol{\gamma}_A$, are derived from the training data by using the methods described in Section 2, $\mathbf{\Gamma}_A = \mathbf{\Sigma}_\mathbf{s}\mathbf{T}_\mathbf{e}(\mathbf{T}_\mathbf{e}^t\mathbf{\Sigma}_\mathbf{s}\mathbf{T}_\mathbf{e})^{-1}$ and $\boldsymbol{\gamma}_A = \bar{\mathbf{s}} - \mathbf{\Gamma}_A\mathbf{T}_\mathbf{e}^t\bar{\mathbf{s}}$ (step i).

The expected deviation function, $\boldsymbol{\delta}_A(\mathbf{r})$, is derived in several steps. For each training data sample, we compute the residual errors from the linear estimator (step ii). These residual errors represent adjustments that improve the linear estimate. The adjustment, $\mathbf{a}_j$, depends on the difference between the jth training data reflectance function, $\mathbf{s}_j$, and the linear estimate,

$$\mathbf{a}_j = \mathbf{s}_j - \boldsymbol{\gamma}_A - \mathbf{\Gamma}_A\mathbf{T}_\mathbf{e}^t\mathbf{s}_j. \tag{8}$$

For any measured response, $\mathbf{r}$, we compute the distance to each of the training responses, $d_j(\mathbf{r}) = \|\mathbf{r} - \mathbf{T}_\mathbf{e}^t\mathbf{s}_j\|$ (step iii). These distances govern how much each adjustment value, $\mathbf{a}_j$, contributes to the submanifold estimate. Training responses near (small d_j) the measured response, $\mathbf{r}$, are given more weight than training responses far (large d_j) from the response. We implement this principle by using a kernel function, $g(d, d_L)$, which specifies a weight that depends inversely on the distance for responses less than d_L from the measured response (step iv). The kernel function used in the calculations shown in this paper, the tricube function,[29] is given in Eq. (9):

$$g(d, d_L) = \begin{cases} [1 - (d/d_L)^3]^3 & \text{if } d/d_L \leqslant 1 \\ 0 & \text{otherwise} \end{cases}. \tag{9}$$

The weights assigned to each training data response, for a measured response, are $w_j(\mathbf{r}) = g(d_j(\mathbf{r}), d_L)$.

The only remaining decision is to specify the distance, d_L. This distance should be chosen to capture the structure of the data submanifold as represented by the training data set. Choosing small values of d_L in regions where the local structure changes rapidly and large values in regions where the local structure remains constant optimizes performance of the submanifold algorithm. This constraint on d_L must be balanced by the need to include an adequate number of training data points in each local region. When only a small number of training points is contained in the local region, the estimate in the region may have a large variance and may overfit in the region. We recommend defining a minimum d_L value based on the data submanifold structure and expanding this value if fewer than, say, 16 training samples fall within this distance.

Next, we compute the expected deviation function (step v). We use a piecewise linear function to map a measured response into adjustment values. The formula is shown in Eq. (10):

$$\boldsymbol{\delta}_A(\mathbf{r}) = \boldsymbol{\phi}_A(\mathbf{r}) + \mathbf{\Phi}_A(\mathbf{r})\mathbf{r}. \tag{10}$$

Here $\mathbf{\Phi}_A(\mathbf{r})$ and $\boldsymbol{\phi}_A(\mathbf{r})$ are the linear adjustment matrix and offset vector, respectively. Both quantities depend on $\mathbf{r}$. They are derived by using weighted linear regression:

$$\begin{aligned}\mathbf{\Phi}_A(\mathbf{r}) = {} & [\mathbf{A} - \tilde{\mathbf{a}}(\mathbf{r})\mathbf{1}_{N_t}^t]\mathbf{W}(\mathbf{r})^2[\mathbf{T}_\mathbf{e}^t\mathbf{S} - \mathbf{T}_\mathbf{e}^t\tilde{\mathbf{s}}(\mathbf{r})\mathbf{1}_{N_t}^t]^t \\ & \times \{[\mathbf{T}_\mathbf{e}^t\mathbf{S} - \mathbf{T}_\mathbf{e}^t\tilde{\mathbf{s}}(\mathbf{r})\mathbf{1}_{N_t}^t]\mathbf{W}(\mathbf{r})^2 \\ & \times [\mathbf{T}_\mathbf{e}^t\mathbf{S} - \mathbf{T}_\mathbf{e}^t\tilde{\mathbf{s}}(\mathbf{r})\mathbf{1}_{N_t}^t]^t\}^{-1}, \end{aligned} \tag{11a}$$

$$\boldsymbol{\phi}_A(\mathbf{r}) = \tilde{\mathbf{a}}(\mathbf{r}) - \mathbf{\Phi}_A(\mathbf{r})\mathbf{T}_\mathbf{e}^t\tilde{\mathbf{s}}(\mathbf{r}). \tag{11b}$$

The columns of the matrix $\mathbf{A}$ contain the adjustment vectors; the columns of $\mathbf{S}$ are the training data; the diagonal matrix $\mathbf{W}(\mathbf{r})$ contains the weights along its diagonal; $\mathbf{1}$ is

the vector of ones, where the subscript indicates the number of one elements; N_t is the number of training samples; $\tilde{\mathbf{a}}$ is the weighted mean of the adjustments $[\tilde{\mathbf{a}}(\mathbf{r}) = \mathbf{A}\mathbf{W}(\mathbf{r})^2\mathbf{1}_{N_t}/\mathbf{w}(\mathbf{r})^t\mathbf{w}(\mathbf{r})]$; and $\tilde{\mathbf{s}}$ is the weighted mean of the training response vectors $[\tilde{\mathbf{s}}(\mathbf{r}) = \mathbf{S}\mathbf{W}(\mathbf{r})^2\mathbf{1}_{N_t}/\mathbf{w}(\mathbf{r})^t\mathbf{w}(\mathbf{r})]$. Both $\mathbf{\Phi}_A(\mathbf{r})$ and $\boldsymbol{\phi}_A(\mathbf{r})$ are a function of the response, $\mathbf{r}$, because the weighting matrix, $\mathbf{W}(\mathbf{r})$, depends on $\mathbf{r}$. Hence $\mathbf{\Phi}_A(\mathbf{r})$ and $\boldsymbol{\phi}_A(\mathbf{r})$ must be recomputed for each measured response.

Finally, the value of the expected deviation function, $\boldsymbol{\delta}_A(\mathbf{r})$, is added to the linear estimate to produce the submanifold estimate [Eq. (7), step vi].

Figure 7 illustrates graphically the entire estimation algorithm for a measurement made with a single system sensor. As in previous figures, the gray circles represent the training reflectance data, and the black vector represents the system sensor responsivity. The distance from the origin to the square on the sensor line represents the value of the sensor measurement. The solid black circles on the sensor vector denote the sensor measurements to the training data that are the nearest neighbors to the current measurement. The samples producing these responses are identified by the open black circles. The estimate (denoted by the x) is calculated by a weighted sum of the training data; these weights are inversely related to the distance between the sensor responses to the training data and the current measurement.

B. Relative-Scale Estimation

Relative-scale estimation applies to measurement conditions when the local intensity of the illuminant is unknown. Examples of such conditions include imaging with a flash, consumer imaging, and studio imaging applications. In these situations the geometry of the scene influences the local illumination intensity as surfaces have various orientations with respect to the illumination or as intensity falls off with distance from a point light source.

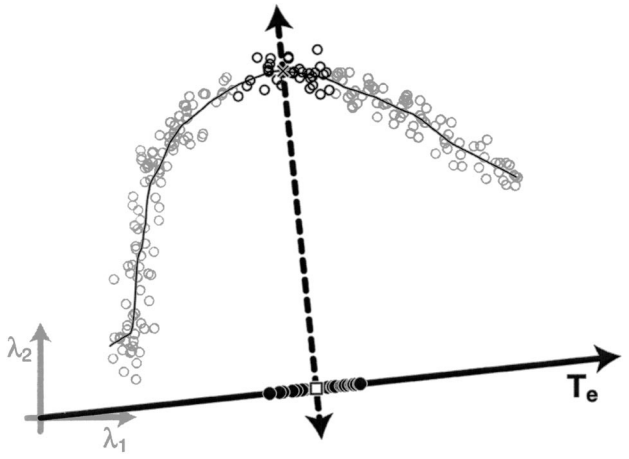

Fig. 7. Submanifold absolute-scale estimation method. The method estimates a spectral signal from a sensor response (open black square). Response values in the training data set that are similar to the measured response are identified (solid black circles). The samples producing these responses are identified in the training data (open black circles). The final estimate (black ×) is a weighted linear fit of these samples. Other details as in Fig. 1.

Under these imaging conditions, an estimation method cannot use absolute intensity information. If the response measured for a particular surface is $\mathbf{r}$, we must admit that the surface reflectance could be $\mathbf{s}$ or $\alpha\mathbf{s}$ depending on the local illumination intensity. Conversely, even if we know $\mathbf{s}$, depending on the local illumination intensity, the measured response could be any one of the values, $\alpha\mathbf{r}$.

Given this restriction, we should not seek to estimate a single reflectance from a response, but rather we must recognize that each measurement is consistent with a set of surface reflectance functions that are related by a scale factor. Equivalently, we should realize that responses related by a scale factor are associated with precisely the same set of possible surface reflectance functions. Hence the surface reflectance estimates derived from the responses $\alpha\mathbf{r}$, for any α, should be the same.

This difference between absolute-scale and relative-scale measurements has a simple geometric intuition. In the absolute-scale case, we measure response points, $\mathbf{r}$, and we estimate absolute surface reflectance functions, $\mathbf{s}$. In the relative-scale case, however, there is an equivalency along the response lines, $\alpha\mathbf{r}$, and from these response lines we estimate surface reflectance lines $\alpha\mathbf{s}$. In this relative-scale case, the estimation problem maps response lines into surface reflectance lines.

A consequence of the estimation restriction is that the entire mapping from measurement to estimate must obey homogeneity. Suppose that we estimate the surface reflectance function from the response $\hat{\mathbf{s}} = \boldsymbol{\rho}(\mathbf{r})$, where $\boldsymbol{\rho}(\cdot)$ is the complete estimation function. We have just described how any scaled version of the response must be associated with a scaled copy of the estimated surface reflectance function, $\boldsymbol{\rho}(\alpha\mathbf{r}) = \alpha\hat{\mathbf{s}}$. It follows that $\boldsymbol{\rho}(\alpha\mathbf{r}) = \alpha\boldsymbol{\rho}(\mathbf{r})$. The mapping function $\boldsymbol{\rho}(\mathbf{r})$ must be homogeneous, but it need not be linear.

We derive the relative-scale algorithm by converting the absolute-scale algorithm [Eq. (7)] to a homogeneous form. We describe the changes to the absolute-scale submanifold algorithm (Fig. 6) necessary to convert Eq. (7) to a homogeneous form.

First, the linear estimator derived in the absolute-scale case is an affine mapping; it includes the addition of a fixed vector (step i). For the relative-scale case, we must remove this fixed vector ($\boldsymbol{\gamma}_R = \mathbf{0}$). Consequently, the absolute-scale linear estimator matrix, $\mathbf{\Gamma}_A$, must be replaced by a new operator to account for the deletion of the fixed vector. This new linear estimator, $\mathbf{\Gamma}_R$, replaces the surface reflectance covariance matrix in the absolute-scale case with the surface reflectance sum-of-products matrix:

$$\mathbf{\Gamma}_R = \mathbf{S}\mathbf{S}^t\mathbf{T}_\mathbf{e}[\mathbf{T}_\mathbf{e}^t\mathbf{S}\mathbf{S}^t\mathbf{T}_\mathbf{e}]^{-1}. \quad (12)$$

Here the columns of the matrix $\mathbf{S}$ contain the training surface reflectance vectors and the columns of $\mathbf{T}_\mathbf{e}$ are the system responsivity functions defined in Section 2.

Next, the adjustment vectors (step ii) are derived by using the absolute-scale equation with the relative-scale linear estimator, $\mathbf{\Gamma}_R$.

To compute the distances between the measured response and the training data responses (step iii), we must take into account that the intensity information is irrelevant. Hence we convert the responses to chromaticity coordinates, $\mathbf{r}^c = \mathbf{r}/(\mathbf{1}_{N_r}^t\mathbf{r})$. Here N_r is the number of system sensors, which is the number of elements in a response vector. The new distance formula is $d_j(\mathbf{r}) = \|\mathbf{r}^c - \mathbf{T}_e^t\mathbf{s}_j/(\mathbf{1}_{N_r}^t\mathbf{T}_e^t\mathbf{s}_j)\|$. The weights are calculated from these distances that have been computed in chromaticity coordinates (step vi).

To compute the relative-scale expected deviation function, $\boldsymbol{\delta}_R(\mathbf{r})$, we need to make two changes to the absolute-scale piecewise linear function [Eq. (10); step v]. First, we eliminate the offset term, $\boldsymbol{\phi}_R(\mathbf{r}) = \mathbf{0}$; second, we recompute the linear estimator matrix, $\mathbf{\Phi}_R(\mathbf{r})$, to compensate for the loss of the offset vector. The new formula is

$$\mathbf{\Phi}_R(\mathbf{r}) = \mathbf{A}\mathbf{W}(\mathbf{r})^2\mathbf{S}^t\mathbf{T}_e[\mathbf{T}_e^t\mathbf{S}\mathbf{W}(\mathbf{r})^2\mathbf{S}^t\mathbf{T}_e]^{-1}. \quad (13)$$

Here the quantities are the same as the absolute-scale case except for the weighting matrix, $\mathbf{W}(\mathbf{r})$. For the relative-scale algorithm, the weights are computed from distances based on chromaticity coordinates of the sensor responses rather than the responses themselves.

Finally, the relative-scale linear estimate and the expected deviation function are added together (step vi). Because both the linear estimate portion and the expected deviation function obey homogeneity, the entire estimation function for the relative-scale application is homogeneous, as required.

Figure 8 shows the geometric intuition behind the relative-scale estimation scale. For graphical simplicity, we assume reflectance functions are defined by three wavelength samples, and the system has two sensors. The short gray vectors represent the three wavelengths, the small gray spheres represent the reflectance functions, and the long black vectors represent the system sensor responsivity functions. The gray planar surface is the system sensor subspace. When a measurement is acquired, reflectance functions are projected onto this subspace (see Section 2). The small black spheres on the planar surface represent the sensor measurements of the gray reflectance functions. The semitransparent surface shows the relative-scale submanifold estimation surface, which is a homogeneous nonlinear surface that passes near the reflectance functions (gray spheres).

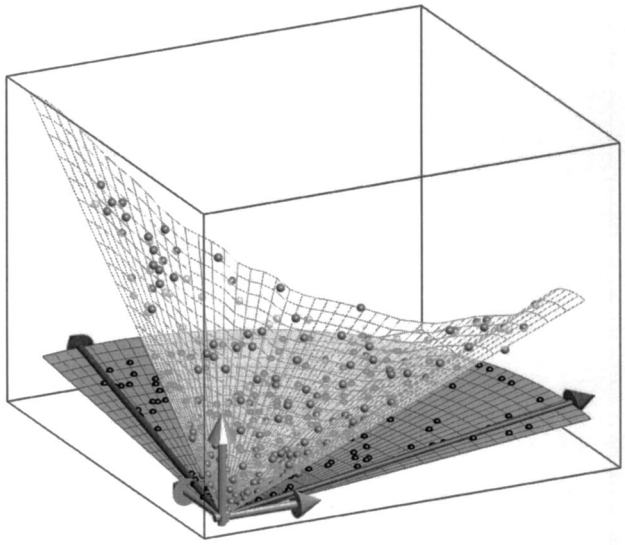

Fig. 8. Submanifold relative-scale estimation example. The gray spheres represent reflectance functions with three wavelength samples. The two black vectors denote sensor responsivity functions that define the gray sensor response plane. The black spheres represent the sensor responses associated with each of the reflectance functions (gray spheres). The semitransparent surface is a homogeneous submanifold estimation surface. See text for details.

4. EXPERIMENT

We illustrate the submanifold estimation in a relative-scale application. Specifically, we use the method to estimate the reflectance functions in a scene captured under a known light source by a three-channel color sensor. This experiment is relevant to typical color cameras when the illuminant is known; in this case the lighting intensity varies owing to the scene geometry. A more extensive analysis of this application, as well as an analysis of an absolute-scale application, will be described elsewhere.[15]

A. Data Collection

The training data were natural and man-made surface reflectance data sets obtained from four sources[36]: the Macbeth Color Checker (24 samples), Munsell chips (1269 samples), Dupont paints (120 samples), and the Vrhel[25] natural surfaces data set (170 samples).

The test data were simulated by using hyperspectral image data[37] and a model camera. We used four hyperspectral test images sampled from 400 to 700 nm every 10 nm. The four extracted test images, rendered for printing, are shown in Fig. 9. They consist of (a) books and an apple, (b) the Macbeth Color Checker, (c) a bear and soft drink can, and (d) fruit.

The simulated camera responses were based on the properties of the QImaging Retiga 1300 camera. The sensor spectral responsivities of this camera were measured with a calibrated light source and an Oriel monochrometer. In the simulations, photon noise and read noise were added on the basis of the camera specifications: (a) an 8-bit camera with an electron well capacity of 16,000 electrons and (b) read noise of 20 electrons.

B. Results

Figure 10 compares the performance of relative-scale submanifold estimation with the best linear estimator. The abscissa represents the linear estimator error, and the ordinate represents the fractional change in error by use of the relative-scale submanifold method. The gray shading indicates the number of image pixels with a given linear error and a given submanifold error change. The horizontal black line indicates fractional change of one: In this case the linear and submanifold errors are equal. Points plotted above this line indicate smaller linear error, and points plotted below indicate smaller submanifold error.

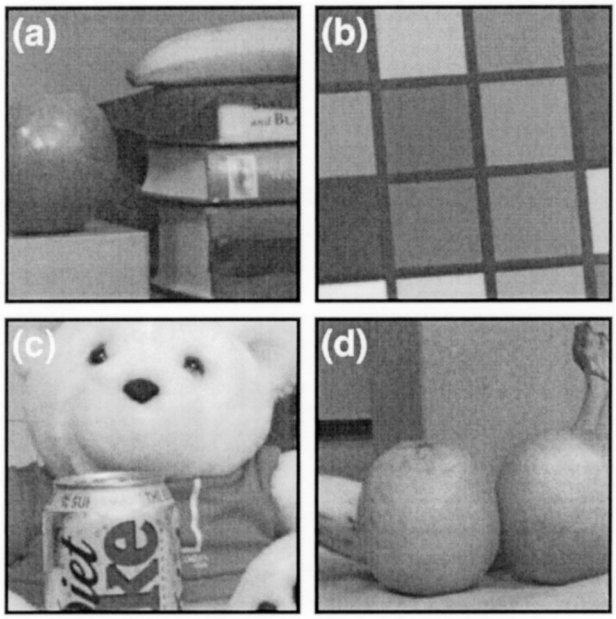

Fig. 9. Hyperspectral images used to evaluate the submanifold algorithms.

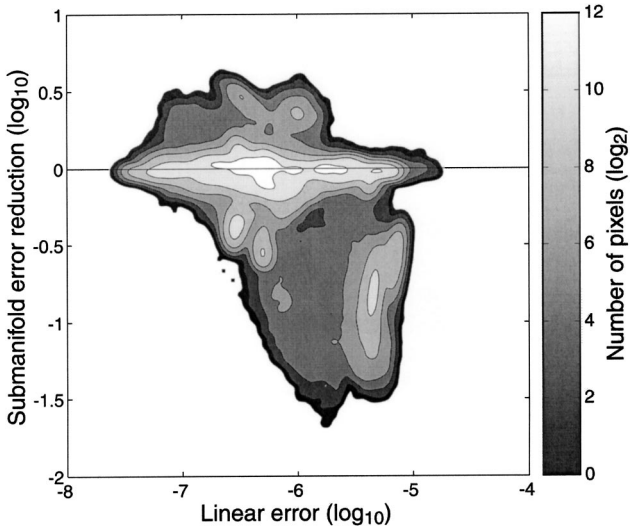

Fig. 10. Submanifold estimation performance. The abscissa represents the linear estimator error, and the ordinate represents the fractional change in error produced by using the relative-scale submanifold method. The horizontal black line indicates fractional change of 1; in this case the linear and submanifold errors are equal. Points plotted above this line indicate smaller linear error, and points plotted below indicate smaller submanifold error. The gray shading indicates the number of image pixels with fractional change in error for each level of linear error.

The average fractional change in error produced by using the submanifold method was 0.88, a 12% reduction in the size of the error. The nature of the reduction, however, is more important than the grand average. In this example, the submanifold method reduces the error for pixels that have a particularly large linear error. As we explain more fully elsewhere,[15] the submanifold method improves the estimates in identifiable local regions of the space. The value of the submanifold routine is mainly in reducing these large deviations, not reducing average error.

5. DISCUSSION

A. Limitations

Submanifold estimation as well as linear estimation assumes that the reflectance estimates are a function of the sensor responses. More specifically, given a sensor response, the methods produce one and only one reflectance estimate. The methods do not adjust their estimates on the basis of other factors, e.g., past measurements or local measurements. This property may limit the estimation results for reflectance functions with complex statistics.

Figure 11 illustrates this limitation for submanifold and linear estimation methods. The figure shows an example of possible reflectances that are not a function of the sensor responses. The reflectance information wraps around itself for the lower sensor responses. The solid curve shows the submanifold estimation function, and the dashed line shows the linear estimation function. Both methods fail to produce estimates that are embedded in the reflectance data because the data cannot be represented as a function. Instead, both methods average the reflectance data (locally or globally) and produce estimates that do not exist. For larger sensor responses in the figure, the reflectance data can be represented as a function. Here submanifold estimation improves its estimates because the technique works locally.

Another limitation of submanifold estimation is that it requires many reflectance measurements to build a high-quality training data set. Submanifold estimation requires data that uniformly sample the likely targets; the method's performance improves as we obtain better information about likely errors by using local values. The number of training data samples needed reflects how well behaved the data are; if the typical deviation from the linear estimate varies slowly across the input targets, then a small number of training data samples will suffice. If the data change rapidly, it will be necessary to sample the data more finely.

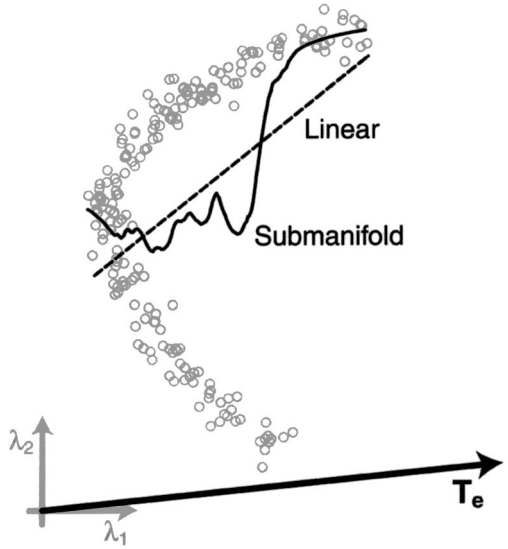

Fig. 11. Estimation limitation. The reflectance data are not a function of the sensor responses. The solid black line represents the submanifold estimation function; The dashed black line represents the linear estimation function. Other details as in Fig. 1.

B. Implementation

Most of the values needed to apply the submanifold method can be computed ahead of time, making the method very efficient. The linear transformation matrix can be precomputed, and the nonlinear mapping can be precomputed and stored in a piecewise linear look-up table. By use of a look-up table, a measured response can be quickly mapped into the appropriate nonlinear estimate. At run time, each measured response is (a) multiplied by the linear transformation matrix and (b) run through a look-up table. The results of these two operations are multiplied by their respective principal components, and, finally, these outputs are summed.

C. Related Research

There has been a considerable amount of research on the estimation of spectral information for various input devices. Hardeberg and Schmitt[38] divide the research into two groups: physical models and empirical models. Algorithms based on physical models use the properties of surface, lights, and sensors to perform the estimation.[39–41] When these physical models are too complex or the physical processes are unknown, estimation algorithms based on empirical models and signal processing are a good alternative. Empirical models are constructed on the basis of training data that are obtained by testing various input values and recording the output values of the system. Most spectral estimation algorithms and color-conversion algorithms are based on empirical models.

Algorithms using empirical models also can be divided into two groups: linear and nonlinear models. Linear models are by far the most common; these are compact descriptions of the input signals, and the linear format works smoothly with the mathematics of sensors and image capture.[1,3–7,17–19] In Section 2 we describe several of the most widely used algorithms. Although these can be useful, there are cases when the underlying physical process is highly nonlinear. Nonlinear models, such as those in this paper, have been developed for these cases. Examples of nonlinear models include polynomial, tetrahedral lookup, and neural networks. Hong *et al.*[42] evaluated the performance of various polynomial models. These models assume that the physical process behind the creation of the spectral information can be described by a polynomial. Tetrahedral models do not assume any global structure; instead, the models assume that the data are locally linear.[38,43] Finally, different types of neural networks have been developed for color calibration.[14,44]

All of these nonlinear models (polynomial, tetrahedral models and neural networks) can be used for spectral estimation in the absolute-scale case. The differences in performance levels between the methods will depend on how well they capture the structure of the data. The polynomial methods make the strongest assumptions about the data submanifold; to the extent that the data deviate from a polynomial surface, the estimates will be incorrect. Neural networks cannot be characterized as a single class because there are so many types of networks. To the extent that they approximate Bayesian networks, however, their performance will be similar to the methods described here or to tetrahedral models.

The main weakness of these nonlinear models is their failure to satisfy homogeneity: Hence they are inappropriate for the important case of relative-scale estimation. Submanifold estimation methods can be used for relative-scale estimation when the intensity of the illuminant is unknown, making the method useful for camera applications as well as scanner applications.

6. CONCLUSIONS

We introduce several ideas for the first time, to our knowledge, in the context of spectral estimation for color reproduction: (a) the explanation of spectral estimation based on submanifold geometry, (b) the accompanying account of the conditions under which training data will succeed in improving on the estimation derived from linear methods, (c) the application of the kernel method for incorporating the training data information, and (d) the method for analyzing the case in which there is an unknown scaling of the data. The submanifold methods are beyond linear but before Bayesian.

ACKNOWLEDGMENTS

We thank Feng Xiao, Peter Catrysse, and Ulrich Barnhoefer for their help. This work was supported by the Programmable Digital Camera project at Stanford, whose founding members are Agilent Technologies, Canon, Kodak, and Hewlett-Packard. Jeffrey DiCarlo is a Kodak Fellow.

Jeffrey M. DiCarlo, the corresponding author, can be reached at Mail Stop 1158, 1501 Page Mill Road, Palo Alto, California 94304; phone, 650-857-4617; e-mail, jeffrey.dicarlo@hp.com

REFERENCES

1. D. B. Judd, D. L. MacAdam, and G. Wyszecki, "Spectral distribution of typical daylight as a function of correlated color temperature," J. Opt. Soc. Am. **54**, 1031–1040 (1964).
2. E. L. Krinov, "Surface reflectance properties of natural formations," Technical Translation **TT-439** (National Research Council of Canada, Ottawa, 1947).
3. J. Cohen, "Dependency of the spectral reflectance curves of the Munsell color chips," Psychonomic Sci. **1**, 369–370 (1964).
4. G. Buchsbaum, "A spatial processor model for object color perception," J. Franklin Inst. **310**, 1–26 (1980).
5. L. T. Maloney and B. A. Wandell, "Color constancy: a method for recovering surface spectral reflectance," J. Opt. Soc. Am. A **3**, 29–33 (1986).
6. B. K. P. Horn, "Exact reproduction of colored images," Comput. Vision Graph. Image Process. **26**, 135–167 (1984).
7. F. O. Huck, D. J. Jobson, S. K. Park, S. D. Wall, R. E. Arvidson, W. R. Patterson, and W. D. Benton, "Spectrophotometric color estimates of the Viking Lander sites," J. Geophys. Res. **82**, 4401–4411 (1977).
8. D. A. Forsyth, "A novel algorithm for color constancy," Int. J. Comput. Vision **5**, 5–36 (1990).
9. D. Brainard and W. Freeman, "Bayesian color constancy," J. Opt. Soc. Am. A **14**, 1393–1411 (1997).
10. G. D. Finlayson, P. M. Hubel, and S. Hordley, "Color by cor-

relation," in *Proceedings of the Fifth Color Imaging Conference* (Society for Imaging Science and Technology, Springfield, Va., 1997), pp. 6–11.
11. K. Barnard, L. Martin, and B. Funt, "Colour by correlation in a three-dimensional colour space," in *Sixth European Conference on Computer Vision* (Springer-Verlag, Berlin, 2000), pp. 275–289.
12. S. Tominaga, S. Ebuisi, and B. A. Wandell, "Scene illumination classification: brighter is better," J. Opt. Soc. Am. A **18**, 55–64 (2001).
13. P. Catrysse, A. E. Gamal, and B. Wandell, "Color architectures for CMOS sensor imaging," in *Sensors, Cameras, and Applications for Digital Photography*, N. Sampat and T. Yeh, eds., Proc. SPIE **3650**, 26–35 (1999).
14. J. Chen and K. Huang, "Adaptive color correction by high-order CMAC neural network," in *Proceedings of the Fifth Color Imaging Conference* (Society for Imaging Science and Technology, Springfield, Va., 1997), pp. 182–186.
15. J. M. DiCarlo and B. A. Wandell, "Spectral estimation examples: beyond linear but before Bayesian," (manuscript in preparation), contact authors for information: dicarlo@white.stanford.edu.
16. T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation* (Prentice-Hall, Englewood Cliffs., N.J., 2000), p. 854.
17. B. A. Wandell, "The synthesis and analysis of color images," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-9**, 2–13 (1987).
18. M. Vrhel and H. Trussell, "Color correction using principal components," Color Res. Appl. **17**, 328–338 (1992).
19. M. S. Drew and B. V. Funt, "Natural metamers," CVGIP: Image Understand. **56**, 139–151 (1992).
20. F. H. Imai and R. S. Berns, "Spectral estimation using trichromatic digital cameras," in *Proceedings of the International Symposium on Multispectral Imaging and Color Reproduction for Digital Archives* (Chiba, Japan, 1999), pp. 42–49.
21. M. Vrhel and J. Trussell, "Color device calibration: a mathematical formulation," IEEE Trans. Image Process. **8**, 1796–1806 (1999).
22. B. A. Wandell and J. E. Farrell, "Water into wine: converting scanner RGB to tristimulus XYZ," in *Device-Independent Color Imaging and Imaging Systems Integration*, R. J. Motta and H. A. Berberian, eds., Proc. SPIE **1909**, 92–100 (1993).
23. K. V. Mardia, J. T. Kent, and J. M. Bibby, *Multivariate Analysis* (Academic, London, 1979).
24. R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis* (Prentice-Hall, Upper Saddle River, N.J., 2002).
25. M. Vrhel, R. Gershon, and L. Iwan, "Measurement and analysis of object reflectance spectra," Color Res. Appl. **9**, 4–9 (1994).
26. G. Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae* (Wiley, New York, 1982).
27. G. Finlayson and M. Drew, "The maximum ignorance assumption with positivity," in *Proceedings of the Fourth Color Imaging Conference* (Society for Imaging Science and Technology, Springfield, Va., 1996), pp. 202–205.
28. J. A. S. Viggiano, "Minimal-knowledge assumptions in digital still camera characterization. I.: Uniform distribution, Toeplitz correlation," in *Proceedings of the Ninth Color Imaging Conference* (Society for Imaging Science and Technology, Springfield, Va., 2001), pp. 332–336.
29. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (Springer-Verlag, Berlin, 2001).
30. J. M. DiCarlo and B. A. Wandell, "Illuminant estimation: beyond the bases," in *Proceedings of the Eighth Color Imaging Conference* (Society for Imaging Science and Technology, Springfield, Va., 2000), pp. 91–96.
31. M. Shi and G. Healey, "Using reflectance models for color scanner calibration," J. Opt. Soc. Am. A **19**, 645–656 (2002).
32. N. Tsumura, M. Kawabuchi, H. Haneishi, and Y. Miyake, "Mapping pigmentation in human skin by multi-visible-spectral imaging by inverse optical scattering technqiue," in *Proceedings of the Eighth Color Imaging Conference: Color Science, Systems and Applications* (Society for Imaging Science and Technology, Springfield, Va., 2000), pp. 81–84.
33. H. Poincaré, "Analysis situs," J. Ec. Polytech Series 2 **1**, 1–123 (1895).
34. V. I. Arnold, "On teaching mathematics," http://pauli.uni-muenster.de/~munsteg/arnold.html (1997).
35. C. S. McCamy, H. Marcus, and J. G. Davidson, "A color-rendition chart," J. Appl. Photogr. **48**, 777–784 (1976).
36. K. Barnard, L. Martin, B. Funt, and A. Coath, "A data set for colour research," Color Res. Appl. **27**, 147–151 (2002).
37. D. H. Brainard, Hyperspectral image data, http://color.psych.ucsb.edu//hyperspectral/,1998.
38. J. Hardeberg and F. Schmitt, "Color printer characterization using a computational geometry approach," in *Proceedings of the Fifth Color Imaging Conference* (Society for Imaging Science and Technology, Springfield, Va., 1997), pp. 97–99.
39. H. R. Kang, *Color Technology for Electronic Imaging Devices* (SPIE Press, Bellingham, Wash., 1997).
40. P. Kubelka and F. Munk, "Ein Beitrag sur Optik der Farbanstrche," Z. Tech. Phys. **12**, 593–601 (1931).
41. H. E. J. Neugebauer, "Die theoretischen Grundlagen des Mehrfarbendruckes," Z. Wiss. Photogr. **36**, 73–89 (1937).
42. G. Hong, M. R. Luo, and P. A. Rhodes, "A study of digital camera colorimetric characterization based on polynomial modeling," Color Res. Appl. **26**, 76–84 (1991).
43. P. Hung, "Colorimetric calibrations in electronic imaging devices using a look-up-table model and interpolations," J. Electron. Imaging **2**, 53–61 (1993).
44. A. Ribes and F. Schmit, "Reconstructing spectral reflectances with mixture density networks," in *Proceedings of the CGIV* (Poitiers, France, 2002), pp. 486–491.