Camera calibration for color research

Kobus Barnard and Brian Funt

School of Computing Science,

Simon Fraser University, Burnaby, BC.

ABSTRACT

In this paper we introduce a new method for determining the relationship between signal spectra and camera RGB which is required for many applications in color. We work with the standard camera model, which assumes that the response is linear. We also provide an example of how the fitting procedure can be augmented to include fitting for a previously estimated non-linearity. The basic idea of our method is to minimize squared error subject to linear constraints, which enforce positivity and range of the result. It is also possible to constrain the smoothness, but we have found that it is better to add a regularization expression to the objective function to promote smoothness. With this method, smoothness and error can be traded against each other without being restricted by arbitrary bounds. The method is easily implemented as it is an example of a quadratic programming problem, for which there are many software solutions available. In this paper we provide the results using this method and others to calibrate a Sony DXC-930 CCD color video camera. We find that the method gives low error, while delivering sensors which are smooth and physically realizable. Thus we find the method superior to methods which ignore any of these considerations.

Keywords: Camera calibration, color constancy, data linearization, constrained least squares, quadratic programming

1. INTRODUCTION

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

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

2. THE CAMERA MODEL

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

$$\rho^{(k)} = F^{(k)}(v^{(k)}) = \int L(\lambda) R^{(k)}(\lambda) d\lambda \tag{1}$$

where $R^{(k)}$ is a sensor sensitivity function for the k'th channel, and $F^{(k)}$ is a wavelength independent linearization function. This model has been verified as being adequate for computer vision over a wide variety of systems.[1-5] This model is also assumed for the human visual system, and forms the basis for the CIE colorimetry standard.

As we move around the image plane the signal is attenuated due to geometric effects, notably vignetting[6], and a fall-off proportional to the fourth power of the cosine of the off axis angle[7]. These effects can be considered to be absorbed into either $R^{(k)}$ or $F^{(k)}$. In our calibration experiments we deferred these considerations by using only the central portion of the image.

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

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

For the practical application of the above model, the continuous function of the wavelength, λ, are replaced by samplings of those functions. For example, our data is collected with a PhotoResearch PR-650 spectraradiometer, which measures data from 380nm to 780nm in 4nm steps. The function $L(\lambda)$ then becomes the vector $\mathbf{L}$, $R^{(k)}(\lambda)$ becomes the vector $\mathbf{R^{(k)}}$, and equation (1) becomes:

$$\rho^{(k)} = F^{(k)}(v^{(k)}) = \mathbf{L} \bullet \mathbf{R^{(k)}} \tag{2}$$

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

3. MOTIVATION FOR COLOUR CAMERA CALIBRATION

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

Determining the function $\mathbf{R^{(k)}}$ is also important for computational color constancy. Most algorithms, including all the ones we currently think are the most promising, require an estimate of what the camera may see when it is used in the real world with its many different surfaces and illuminations. Although it is conceivable to obtain camera responses for a large number of surfaces under a given illuminant, it is impractical to obtain this data for each camera. Some algorithms would further require this data for each possible illumination, including combinations of several sources. It is thus far more effective to first measure or obtain reflectance functions and illuminant spectra, and then to use a camera model to predict the wide range of camera responses required for these algorithms. If the camera model can be easily updated, then databases for reflectance and illuminant spectra can be continuously improved, shared, and re-used for color constancy or similar problems on many different cameras.

4. PREVIOUS WORK

Since $F^{(k)}$ is independent of wavelength, it can be determined by stimulating the camera with varying intensities of a single light source, obtained with neutral density filters, or by simply moving the source. An appropriate function can then be fitted to the data, or alternatively a smoothed version of the data can be used to generate a look up table. Vora et all[4] used this method to verify that a Kodak DCS-200 digital camera was linear over most of its operating range, and also to develop a linearization curve for a Kodak DCS-420 digital camera. They then determined $\mathbf{R^{(k)}}$ for those cameras by stimulating them with very narrow band illumination produced by a monochrometer.[5] This method is conceptually very simple, and if done carefully, it can be very accurate. However, the equipment required to produce sufficiently intense narrow band illumination at uniformly spaced wavelengths is expensive and not readily available. Hence various researchers have investigated methods for calibration which do not use such equipment.

The general approach of these methods is to first measure $F^{(k)}$, and then to measure a number of input spectra and the corresponding camera responses. Let $\mathbf{r}^{(k)}$ be a vector whose elements are the linearized camera responses $\rho^{(k)}$ and let L be a matrix whose rows are the corresponding sampled spectra. Then (2) becomes:

$$\mathbf{r}^{(k)} = L\mathbf{R}^{(k)} \tag{3}$$

Equation (3) can be solved by multiplying both sides by the pseudo-inverse of L. However, this does not work very well because L is invariably rank-deficient. L is rank-deficient because we are now trying to determine R using easily obtainable input spectra, and these tend to be of relatively low dimensionality. If L was of full rank, then we would have a method analogous to the monochrometer method. Given that L is rank deficient, the results using this method are very sensitive to noise (since it is mainly the noise that is being fitted), and the resulting sensor responses tend to have numerous large spikes, and have an abundance of non-negligible negative values (see Figure 4).

Sharma and Trussell[1] improved the prospects for a reasonable solution by introducing various constraints on $\mathbf{R}^{(k)}$. First, instead of solving (3) exactly, they constrained the maximum allowable error as well as the RMS error. In addition, they constrained a discrete approximation of the second derivative to promote a smooth solution. Finally, they constrained the response functions to be positive. They then observed that the constraint sets are all convex, and so they computed a resulting constraint set using the method of projection onto convex sets.

Hubel et al[8] also recognized that some form of smoothness was necessary for a good solution, and they investigated the Wiener estimation method, as described by Mancill and Pratt[9], as a method for finding a smooth fit. They found that the method gave generally good results. They note, however, that the method produced negative lobes in the response functions, and mention briefly using the projection onto convex sets method to remedy this problem.

Sharma and Trussell's contribution was the starting point for some of our own work on this problem.[3] Rather than constrain the absolute RMS error, we chose instead to minimize the relative RMS error. We then re-wrote Sharma and Trussell's other constraints so that the entire problem became a least squares fit with linear constraints for which there are standard numerical methods readily available. Once we had a fit for our camera sensors, we noted that they were essentially uni-modal, and that once the sensors dropped to a small value they remained small. On these grounds we also constrained the sensors to be zero outside a certain range on subsequent runs. In this particular case this forced the sensors curves to be uni-modal. This last step needs to be applied with care, as it is possible that the sensors are in fact non-zero beyond the points where the main peaks drop to zero.

Recently Finlayson and Hordley also used a similar approach.[10] They constrained smoothness by constraining the sensors to be linear combinations of the first 9-15 Fourier basis functions. They also introduced a modality constraint expressed in terms of the peak location. They then found the best fit for the proposed modality by stepping through all possible peak locations. This method also requires care, as the modality is often unknown.

5. THE FITTING APPROACH

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

In initial work[3] we minimized the relative RMS error in equation (3) subject to positivity constraints, smoothness constraints, and constraint on the maximum allowable error (and/or relative error). We have since found that it is better to replace the constraint on smoothness with a regularization term added to the objective function. Thus we minimize the relative error and the non-smoothness measure together. This allows deviations from one to be traded against deviations of the other. With the hard constraint used previously, there is no recourse in the case that making the sensor response slightly less smooth at a particular location can substantially reduce the error. Similarly, there is no recourse when a small increase in error beyond the hard limit can substantially increase the smoothness.

We choose to minimize the relative RMS error because the standard deviation of the R, G, and B values used for calibration increased with their mean (see Figure 1), making relative error most sensible. This choice is supported by our finding that

minimizing relative error better reduces the error in chromaticity, which is difficult to minimize directly, but is often of most interest, as discussed above. However, for some applications minimizing absolute error, or even a weighted combination of both, may make more sense. We have also found that it is not generally necessary to use Sharma and Trussell's constraint on maximum allowable error to get good results, but again, limiting either the absolute error or relative error may be called for in some cases, and is easily added to the method.

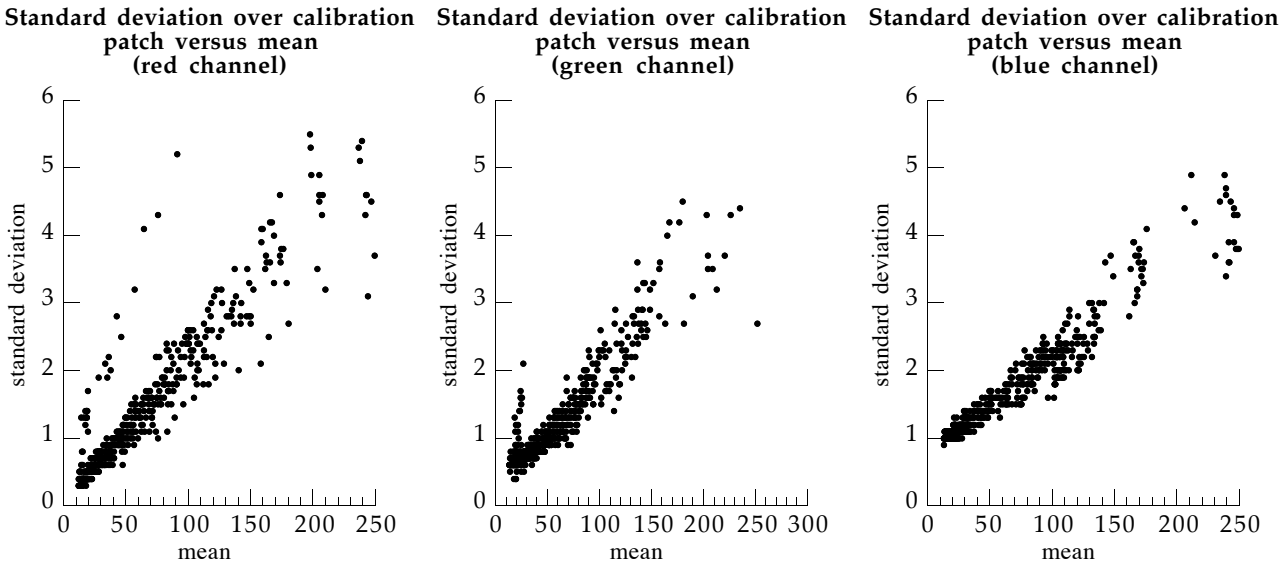

Figure 1: The standard deviation of the R, G, and B measurements used for the calibration experiments described in the text. The R, G, and B values used in those experiments were obtained by taking both spatial and time averages. The graphs here show how the standard deviation over these measurements depend on their magnitude. The fact that the standard deviation increases with the magnitude indicates that fitting based on relative error makes more sense than fitting based on absolute error.

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

$$S = \begin{vmatrix} -1 & 2 & -1 & & & & \\ & -1 & 2 & -1 & & & \\ & & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & & \\ & & & -1 & 2 & -1 & \\ & & & & -1 & 2 & -1 \end{vmatrix} \tag{4}$$

Then we solve

$$\left|\frac{L}{\lambda S}\right| \mathbf{R}^{(k)} = \left|\frac{\mathbf{r}^{(k)}}{\mathbf{0}}\right| \tag{5}$$

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

$$\sum_i \left(\mathbf{L}_i \bullet \mathbf{R}_i^{(k)} - \rho_i^{(k)}\right)^2 \tag{6}$$

and the lower part forms a term for smoothness. The coefficient λ specifies the relative weight attributed to the two terms. If λ is zero and there are no constraints, then this becomes the pseudo inverse method. A serviceable value for λ is easily found by trial and error. To ensure positivity, we use the constraint:

$$\mathbf{R}^{(k)} \geq \mathbf{0} \tag{7}$$

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

$$R_i^{(k)} \le 0 \quad \text{for i<min, i>max} \tag{8}$$

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

$$\rho_i^{(k)} - \delta \le L\mathbf{R}^{(k)} \le \rho_i^{(k)} + \delta \tag{9}$$

To minimize the relative error we need to replace (6) by:

$$\sum_i \left(\frac{\mathbf{L}_i \bullet \mathbf{R}_i^{(k)} - \rho_i^{(k)}}{\rho_i^{(k)}} \right)^2 = \sum_i \left(\frac{\mathbf{L}_i \bullet \mathbf{R}_i^{(k)}}{\rho_i^{(k)}} - 1 \right)^2 \tag{10}$$

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

$$L_{rel}^{(k)} = (\text{diag}(\mathbf{r}^{(k)}))^{-1} \bullet L \tag{11}$$

We then replace (5) with:

$$\left| \frac{L_{rel}^{(k)}}{\tilde{\lambda}\tilde{S}} \right| \mathbf{R}^{(k)} = \left| \frac{\mathbf{1}}{\mathbf{0}} \right| \tag{12}$$

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

$$1 - \zeta < L_{rel}^{(k)} \mathbf{R}^{(k)} < 1 + \zeta \tag{13}$$

We note that minimizing the relative error may need to be modified slightly to deal with very small $\rho^{(k)}$. Such data is likely to be inaccurate for other reasons. For example, if there is a camera offset due to camera black, then small values of $\rho^{(k)}$ include error from $F^{(k)}$. Thus we need to either ignore small values of $\rho^{(k)}$ or give the corresponding data row less weight in the fitting process. Equation (12) can be interpreted as a weighted version of equation (5), with the weighting being inversely proportional to $\rho^{(k)}$. Thus it is natural and easy to put an upper bound on this weighting, and this is how we safeguard against small $\rho^{(k)}$ when we do not want to exclude them outright.

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

The Sony DXC-930 camera which we used for our experiments is quite linear for most of its range, provided it is used with gamma disabled. However, it has a non-negligible offset and a slight non-linearity for small pixel values. Due to this non-linearity, the offset is not the camera black, and using the camera black for the offset leads to errors in chromaticity. Therefore the non-linearity must be taken into account, even if it is not explicitly fitted. Figure 2 shows the slight non-linearity for the red channel. The other two channels are similar.

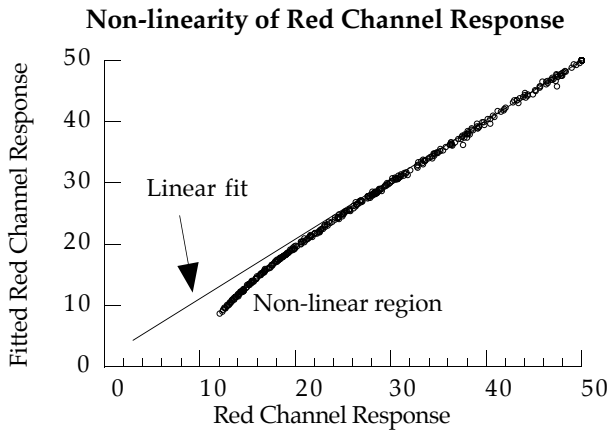

Figure 2: The non-linearity of the red channel response for the Sony DXC-930 camera used for the sensor fitting experiments. The other two channels have similar curves.

The linear fit shown in Figure 2 was found using a simple linear fit on the calibration data using pixel values greater than 30. This illustrates the point that linearization information is available in the data set which is to be used to find $\mathbf{R}^{(k)}$. If we forced a linearization function found using less data onto the fit for $\mathbf{R}^{(k)}$, then this information would be disregarded. To proceed with our strategy, we need to parameterize the non-linearity. The particular form of the parameterization is somewhat arbitrary and will vary substantially from case to case. With a little experimentation we found that the non-linearity for our camera could be approximated by:

$$F^{(k)}(x) = x - a_0^{(k)} - a_1^{(k)} e^{-C_k(x-b_k)} \tag{14}$$

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

$$F^{(k)}(x) = x - a_0^{(k)} \tag{15}$$

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

To find the parameters for the approximation of $F^{(k)}(x)$ simultaneously with $\mathbf{R}^{(k)}$ when fitting for absolute error, we replace equation (5) with:

$$\left|\begin{array}{c|cc} L & \mathbf{1} & e^{-C_k(\mathbf{r}^{(k)}-b_k)} \\ \hline \lambda S & 0 & 0 \end{array}\right| \bullet \left|\begin{array}{c} \mathbf{R}^{(k)} \\ \hline a_0^{(k)} \\ a_1^{(k)} \end{array}\right| = \left|\begin{array}{c} \mathbf{r}^{(k)} \\ \hline \mathbf{0} \end{array}\right| \tag{16}$$

where the arithmetic in the upper right block of the matrix is done element-wise as needed. Similarly, in the case of fitting for relative error, we replace (12) with:

$$\left|\begin{array}{c|cc} L_{rel} & \dfrac{\mathbf{1}}{\mathbf{r}^{(k)}} & \dfrac{e^{-C_k(\mathbf{r}^{(k)}-b_k)}}{\mathbf{r}^{(k)}} \\ \hline \lambda S & 0 & 0 \end{array}\right| \bullet \left|\begin{array}{c} \mathbf{R}^{(k)} \\ \hline a_0^{(k)} \\ a_1^{(k)} \end{array}\right| = \left|\begin{array}{c} \mathbf{r}^{(k)} \\ \hline \mathbf{0} \end{array}\right| \tag{17}$$

where again, the arithmetic in the upper right block of the matrix is done element-wise as needed. Note that the response vectors $\mathbf{r}^{(k)}$ now correspond to the observed camera responses $v^{(k)}$ in equation (2) in contrast to the earlier formulation where $\mathbf{r}^{(k)}$ corresponded to the linearized camera responses, $\rho^{(k)}$.

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

6. EXPERIMENTAL METHOD

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

We took additional steps to reduce the error. As indicated above, it is important that the camera and the spectrometer are exposed to the same signal. In order to minimize the effect of misalignment, we made the illumination as uniform as

possible. We extracted a window from the image which corresponded as closely as possible to the area used by the spectrometer. The pixels in this window were averaged. The camera measurements were further averaged over 50 frames. The spectrometer measurements were averaged over 20 capture cycles. Finally, in order to reduce the effect of flare, the entire setup was viewed through a hole in a black piece of cardboard, exposing the region of interest, but as little else as was practical. Some of the periphery was exposed to aid alignment.

7. RESULTS

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

The sensors computed using the new fitting method are shown in Figure 3. To compute the sensor curves shown, the range constraint in equation (8) was not used. If we use the range constraint to force the curves to be zero once they drop to zero, then we get sensor curves which are visually very close to the ones shown, except of course, they are exactly zero outside the main peak. Adding these range constraints increase the RMS relative error from 1.02% to 1.24% which is more than we expected. Thus either the camera really does have small responses outside the main peaks, or, more likely, there is some source of error which we have not yet isolated. Two possibilities to consider are flare and fluorescence in some of the surface reflectances.

The fitting process produces both the sensor curves, and the linearization function. We can use this linearization function to linearize the data, and further compare fitting methods. Figure 4 shows the sensor curves obtained using the pseudo inverse method on the linearized data (lambda=0 and no constraints). As expected, the result has a large number of extreme spikes. Figure 5 shows the sensor curves obtained using the pseudo inverse method with positivity constraints. Table 1 reports the RMS relative errors of the various methods.

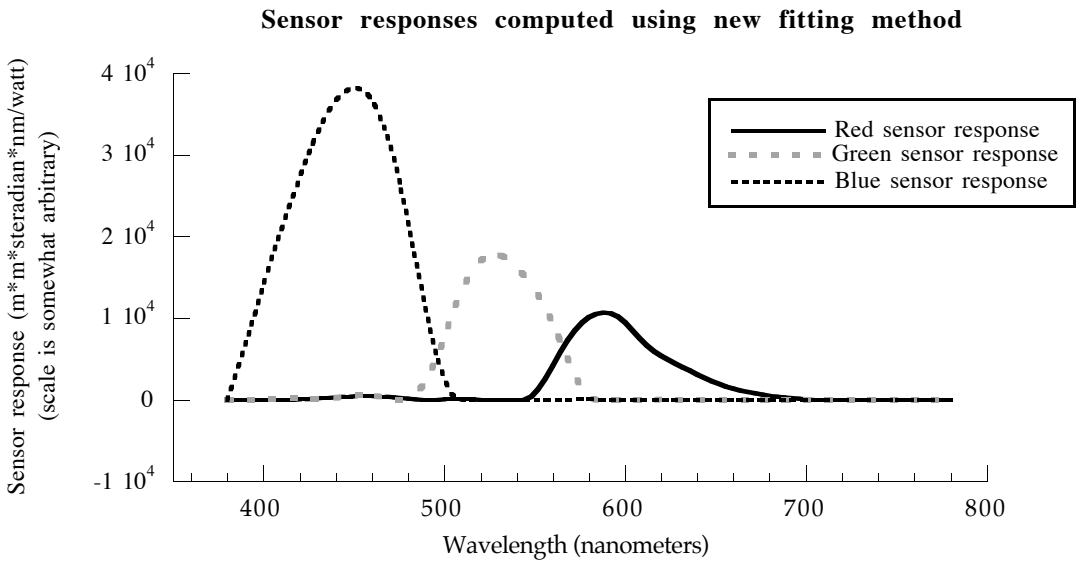

Figure 3: The sensor response found using the fitting method introduced in this paper without the range constraints in equation (8). Tow of the sensors have small responses outside the main peaks that can be removed by including the range constraints set to the obvious boundaries of the main peaks. The resulting sensors are otherwise very similar.

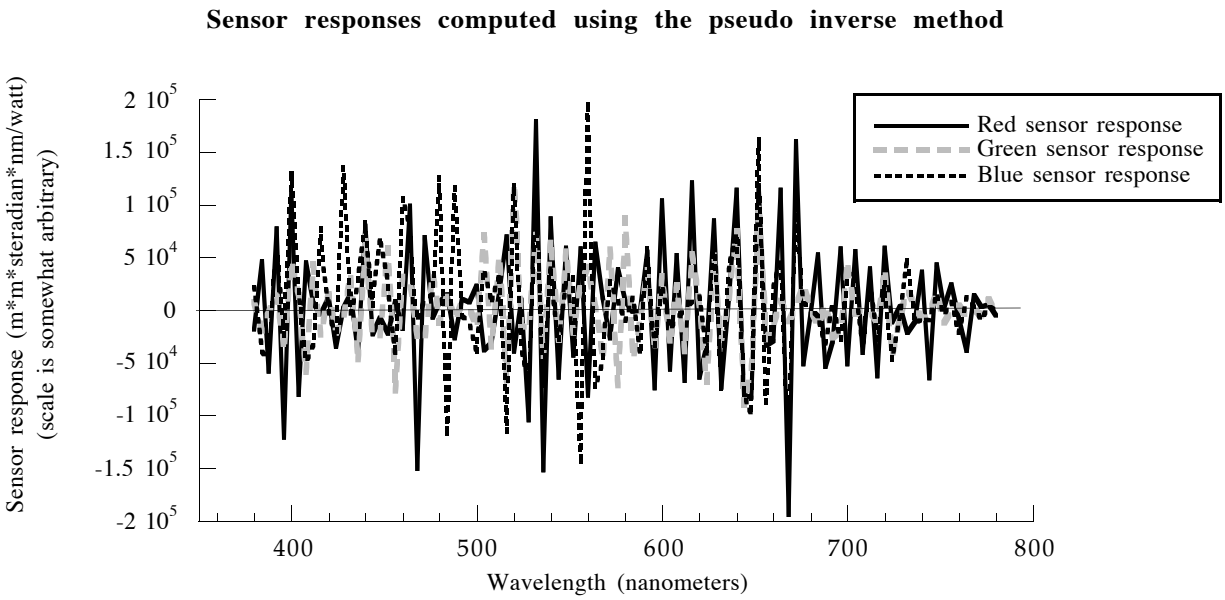

Figure 4: The sensor response found using the pseudo inverse method applied to linearized data. This method minimizes the error, but takes nothing else into account. Since the matrix is rank-deficient, the method mainly fits the noise in the data. The resulting sensors are clearly incorrect, and if they are used on different data, the error will be very large.

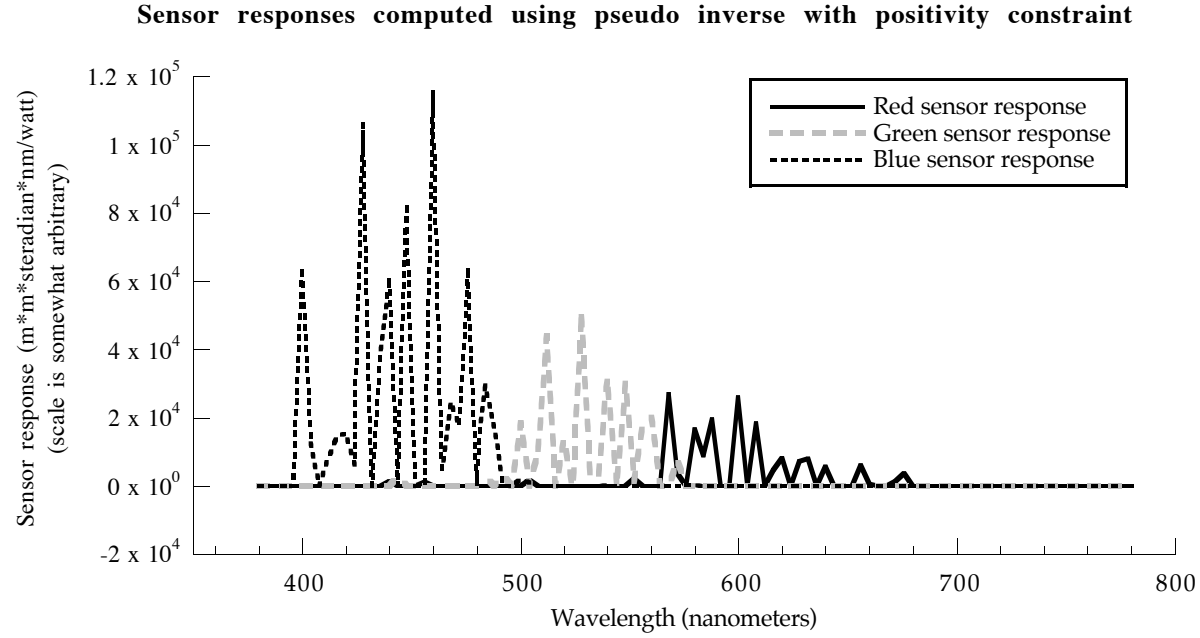

Figure 5: The sensor response found using the pseudo inverse method with positivity constraints applied to linearized data. The spikes conform to the rough outline of the sensors, and the result is surprisingly robust (see Figure 6). Nonetheless, there is no reason to use these sensor responses, as they are clearly incorrect, and the results plotted in Figure 6 indicate that the error on different data will be larger than the other methods.

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

Table 1: Error obtained on 612 data points using various fitting methods. The first two methods fit for both linearity and the sensor curves. The last five methods are applied to linearized data.

As expected, for the five results on the same (linearized) data, as constraints are added, the error increases. In the extreme case, the pseudo inverse method has the lowest error, but the sensor curves are obviously not correct. As mentioned above, because the matrix is rank-deficient, this method fits the noise of the specific data set. This becomes evident when we look at how well the fitting methods perform when run on a subset of the data. To test this, we ran each of the methods on sub-sets of the data of sizes 400, 200, 100, 50, 25, and 15. The RMS error was computed using all 612 measurements. The result for each sample size was the average of the results for several hundred randomly selected samples. The results are shown in Figure 6. The pseudo inverse method degrades very rapidly with decreasing sample size, verifying that the fits are largely an artifact of the chosen data set. If positivity constraints are added to the pseudo inverse method, then the results are much better and surprisingly robust. However, the error quickly goes from slightly less than that for the more reasonable smooth fit, to somewhat more, and thus there is no reason to use such a method. The other methods tested are quite robust, degrading slowly and reasonably with decreasing sample size.

Figure 6 also shows that the addition of the range constraint also aids robustness. This result is somewhat artificial, as the range constraints used were those determined from the full data fits, but nonetheless the graph illustrates how constraints become more critical as the amount of available data decreases.

8. CONCLUSIONS

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

ACKNOWLEDGMENTS

We are grateful for the support of Hewlett-Packard Corporation and the Natural Sciences and Engineering Council of Canada. In addition we acknowledge the efforts of Lindsay Martin who helped greatly with the data collection.

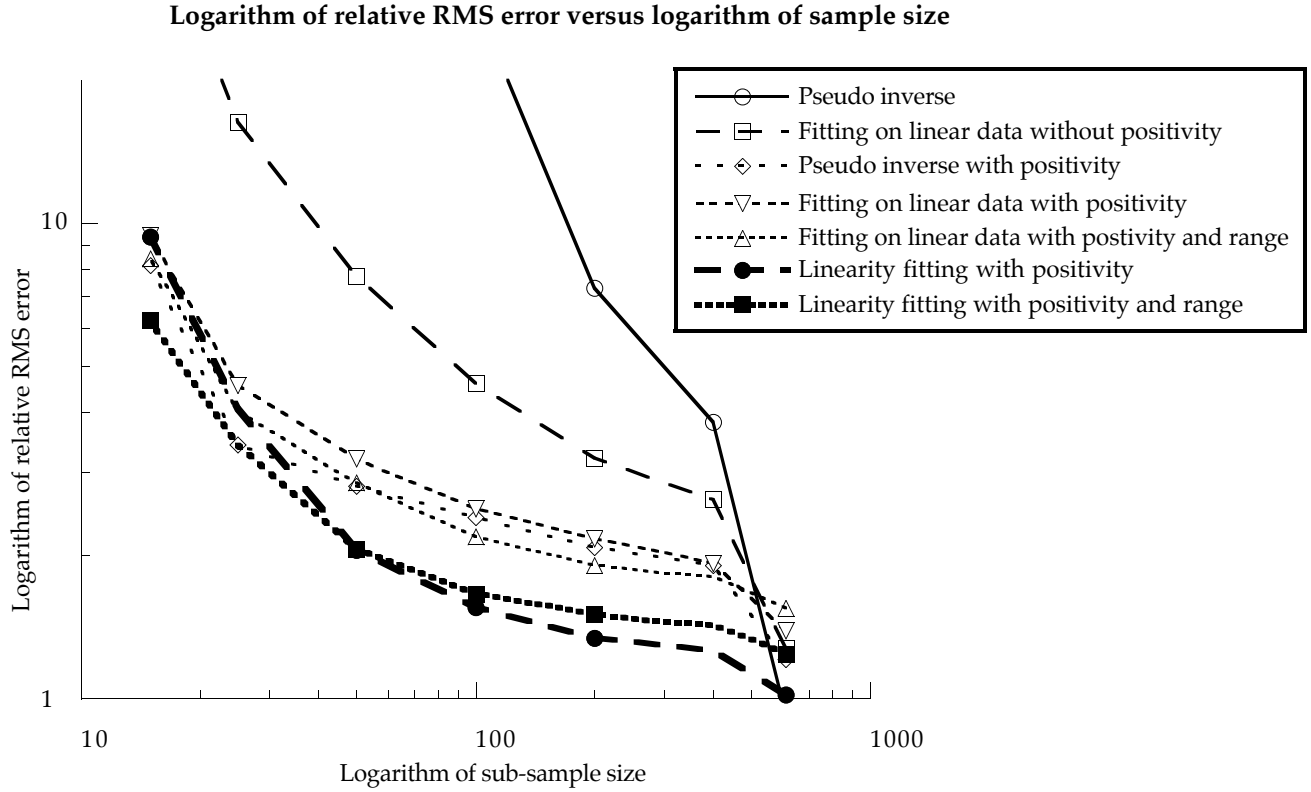

Figure 6: RMS relative fitting error versus sample size (log scales). The methods which are more constrained are more robust. The unconstrained pseudo inverse method (first curve), degrades rapidly with decreasing sample size. A fit with smoothness, but without positivity (second curve) also degrades quite rapidly. This indicates that promoting smoothness alone cannot ensure a robust result. Interestingly enough, positivity without smoothness gives a fairly robust result (third curve). This graph also shows the improved performance of the fitting method which includes fitting the linearization function (the two darker curves).

REFERENCES

1. G. Sharma and H. J. Trussell, "Characterization of Scanner Sensitivity," In *IS&T and SID's Color Imaging Conference: Transforms & Transportability of Color*, pp. 103-107, 1993.
2. Glenn E. Healey and Raghava Kondepudy, "Radiometric CCD camera calibration and noise estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**(3), pp. 267-276, 1994.
3. Kobus Barnard, "Computational color constancy: taking theory into practice," MSc thesis, Simon Fraser University, School of Computing, 1995.
4. P. L. Vora, J. E. Farrell, J. D. Tietz, D. H. Brainard, "Digital color cameras—1—Response models," Available from http://color.psych.ucsb.edu/hyperspectral/
5. P. L. Vora, J. E. Farrell, J. D. Tietz, D. H. Brainard, "Digital color cameras—2—Spectral response," Available from http://color.psych.ucsb.edu/hyperspectral/
6. B. K. P. Horn, *Robot Vision*, MIT Press, 1986, page 26.
7. B. K. P. Horn, *Robot Vision*, MIT Press, 1986, page 208.
8. P. M. Hubel, D. Sherman, and J. E. Farrell, "A comparison of method of sensor spectral sensitivity estimation," *IS&T/SID 2nd. Color Imaging Conference: Color Science, Systems and Applications*, pp. 45-48, 1994.
9. W. K. Pratt and C. E. Mancill, "Spectral estimation techniques for the spectral calibration of a color image scanner," *Applied Optics*, **15**(1), pp. 73-75, 1976.
10. Graham Finlayson and Steven Hordley, "Recovering device sensitivities with quadratic programming," *IS&T/SID Sixth Color Imaging Conference: Color Science, Systems and Applications*, pp. 90-95, 1998