Estimating the Spectral Sensitivity of a Digital Sensor using Calibration Targets

Marc Ebner
Universitat Wurzburg, Lehrstuhl fur Informatik II
Am Hubland, 97074 Wurzburg, Germany
ebner@informatik.uni-wuerzburg.de

ABSTRACT

A digital sensor which is used inside a digital camera usually responds to a range of wavelengths. The response of the sensor is proportional to the product of the irradiance falling onto the sensor and the sensitivity of the sensor integrated over all wavelengths. Knowledge of the sensor's response function is important for colorimetry and the research area of color constancy. Such data may not always be available from the manufacturer of the camera. The sensitivity of the imaging device is a result of the hardware properties of the imaging chip, the lens and filters used, and the post-processing done by the processor contained inside the camera. We will be using an evolution strategy to obtain the sensor response curves of a camera given a single image of a calibration target.

Categories and Subject Descriptors

I.4.1 [**Digitization and Image Capture**]: Camera calibration; G.1.6 [**Optimization**]: Constrained optimization; G.1.6 [**Optimization**]: Global optimization

General Terms

Algorithms, Measurement, Performance, Experimentation

Keywords

Spectral Sensitivity, Evolution Strategies, Constraints, Camera Calibration, Colorimetry

1. MOTIVATION

A digital camera is equipped with a sensor array which measures the incident light in order to obtain an image of the scene viewed. Filters are used to make the sensor array respond to light in the red, green and blue parts of the spectrum. The sensors are usually arranged in a matrix as shown in Figure 1 which is called a Bayer pattern [1]. When the image is generated, data from adjacent sensors are interpolated in order to obtain the RGB color for each image pixel. Some imaging chips, e.g. the Foveon X3 sensor [16], measure the RGB components for each image pixel. Some cameras also use more than three sensors.

A sensor usually responds to a range of wavelengths. The response of the sensor is proportional to the product of the irradiance falling onto the sensor and the sensitivity of the sensor integrated over all wavelengths. We will formalize this below. The sensitivity of the sensor is not always available from the manufacturer of the camera. For instance, for our research we are using a Canon 10D. Unfortunately, the sensitivities of the imaging sensor could not be obtained from Canon.

We will be using an evolution strategy [25, 28, 12] to obtain the sensor response curves of a digital camera given a single image of a calibration target. The sensitivity curves are a result of the entire device which consists of the image sensor, the lens together with filters which may be used and also the internal processing done by the camera. Note that the goal of this is not color calibration. We want to obtain the sensor response curves given an imaging device and a calibration target. Of course, once the sensitivities are known, this information can also be used for colorimetry [33, 19]. Knowledge of the sensor's response function is also important in the area of color constancy [32, 9, 10, 11].

A number of authors have already used evolutionary algorithms for a variety of image processing applications. Some have used evolutionary algorithms for camera calibration, e.g Zhang and Ji [35] or Rodehorst and Hellwich [26], who used a genetic algorithm or Cerveri et al. [4] who worked with an evolution strategy. Johnson et al. [20] have used a genetic algorithm for projector calibration.

In these cases, the internal or external parameters of either a camera or a projector were determined using an evolutionary algorithm. More relevant to the research described here, is the research of Carvalho et al. [3] who used a genetic algorithm to maximize the prediction ability of an extended generalized cross-validation measure. In contrast to this work, we will be using an evolution strategy where the sensitivities of the sensor are coded into the genotype. Because of the use of integrated signals, the problem can only be solved if appropriate constraints are enforced. Our research focuses on how the necessary constraints such as positivity or smoothness can be applied.

There is a large body of literature with regard to evolutionary algorithms and constraint handling. Extensive surveys can be found in [29, 22, 23, 34, 6]. Kuri-Morales and

GECCO'07, July 7–11, 2007, London, England, United Kingdom.

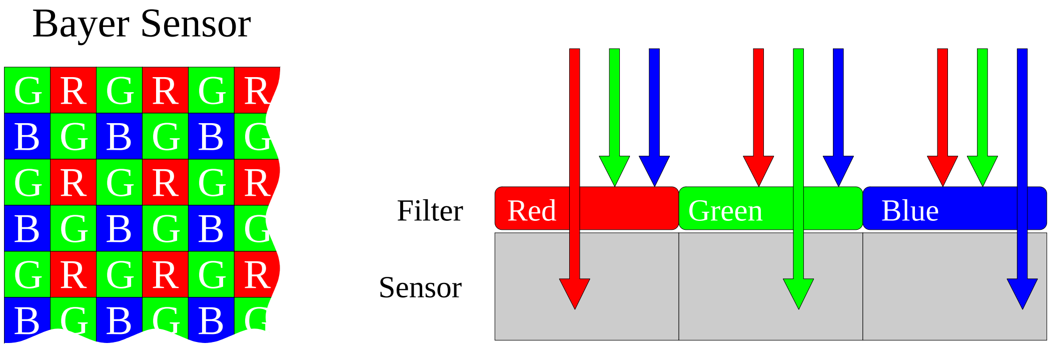

Figure 1: Bayer pattern (left). Filters are used to make the sensor respond in the red, green, and blue part of the spectrum (right).

Gutiérrez-García [21] evaluate different constraint handling approaches. Many researchers use penalty functions in order to enforce constraints, e.g. [18, 17]. It has also be suggested to provide a search direction from infeasible solutions to feasible solutions [7]. A differential evolution approach for constrained multi-objective optimization problems was proposed by Sarker et al. [27]. Constraint handling via multi-objective optimization was proposed by Coello Coello [5]. Constraint handling via multi-objective optimization in a multi-objective fitness context is described by Vieira et al. [31]. Instead of using penalty functions, we can also enforce hard constraints on the genotype. In this case, the genotype is modified or repaired such that the constraint is enforced. We will also be looking at how to incorporate smoothness constraints. A new iterative method is proposed on how to incorporate a smoothness constraint iteratively.

Before we describe the type of evolution strategy used, we first formalize our model of color image formation.

2. THEORY OF COLOR IMAGE FORMATION

A digital sensor measures the light reflected from the objects around us. Consider a single light source which illuminates the scene. The radiance emitted by the light source falls onto an object patch. Part of the irradiance is absorbed, the remainder is reflected. Eventually, the reflected light will enter the lens of the camera or the ray of light will disappear into infinity. Inside the camera, the irradiance will be measured by a sensor array. At each image position we will have one image sensor. Each image position has one corresponding object patch. The correspondence will be determined by the type of lens used. Let $L(x, y, \lambda)$ be the irradiance at wavelength λ which is falling onto the object patch corresponding to image pixel (x, y). Let $R(x, y, \lambda)$ be the reflectance, i.e. the percentage of reflected light at wavelength λ. The reflectance also varies with the position (x, y). Let $S(x, y)$ be the response characteristic of the sensor. Then the energy measured by a single sensor I at position (x, y) of the sensor array is given by

$$I(x,y) = \int S(x,y)R(x,y,\lambda)L(x,y,\lambda)d\lambda \qquad (1)$$

where the integration is done over all wavelengths λ to which the sensor responds. This model is frequently used in colorimetry and color constancy [2, 8, 15].

A digital camera usually uses three different types of sensors which primarily respond to the red, green and blue parts of the spectrum. Let $S_i(\lambda)$ with $i \in \{r, g, b\}$ be the response curves of the three sensors, i.e

$$\mathbf{S}(\lambda) = [S_r(\lambda), S_g(\lambda), S_b(\lambda)]. \qquad (2)$$

In this case, the energy measured by the three sensors will be given by

$$\mathbf{I}(x,y) = \int \mathbf{S}(\lambda)R(x,y,\lambda)L(x,y,\lambda)d\lambda \qquad (3)$$

with $\mathbf{I} = [I_r(x,y), I_g(x,y), I_b(x,y)]$.

In case of a Lambertian surface which reflects light equally in all directions, the result will be scaled by the scalar product of the normal vector which points from the surface patch into the direction of the light source and the normal vector of the surface. Let $\mathbf{n}_S(x, y)$ be the normal vector of the surface patch and let $\mathbf{n}_L(x, y)$ be the normal vector which points into the direction of the light source. We use $G(x, y) = \mathbf{n}_S^T(x, y)\mathbf{n}_L(x, y)$ to denote the geometry factor. The geometry factor scales all color channels equally and we obtain

$$\mathbf{I}(x,y) \;=\; G(x,y)\int \mathbf{S}(\lambda)R(x,y,\lambda)L(x,y,\lambda)d\lambda. \qquad (4)$$

When the measured image data is saved into an image, a gamma correction is usually applied. Most often the sRGB standard [30] is used to store images. The sRGB standard uses the following transfer function [24].

$$\mathrm{gamma}_{\mathrm{sRGB}}(x) = \begin{cases} 12.92x & \text{if } x \leq 0.0031308 \\ 1.055x^{\frac{1}{2.4}} - 0.055 & \text{if } x > 0.0031308 \end{cases} \qquad (5)$$

This transform has a linear section for small intensities. The sRGB standard contains a power function with an exponent of 2.4, however, the overall curve is best described by the function

$$\mathrm{gamma}(x) = x^{1/2.2}. \qquad (6)$$

Modern cathode ray tubes have a gamma factor of 2.5, i.e. they have a transfer function of $\mathrm{gamma}(x) = x^{2.5}$ which results in an end-to-end gamma which is suitable for an office environment. The gamma correction together with the non-linearity of the display device ensures that the colors are accurately displayed. High quality flat panel displays allow choosing the sRGB color space for display. Digital

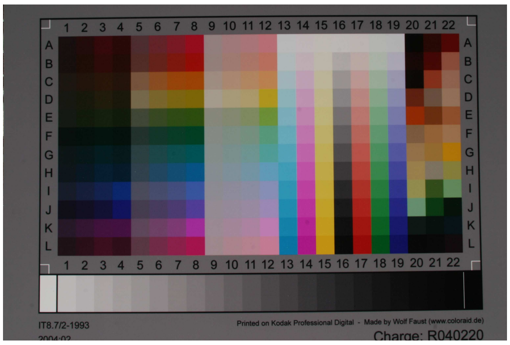

Figure 2: IT8 calibration target made by Wolf Faust. An IT8 target may be used to calibrate scanners or similar imaging devices.

images are frequently stored using the sRGB color space. When processing image data from a file we have to undo this gamma correction such that the image data is again linear.

3. OPTIMIZATION OF SENSOR RESPONSE CURVES

In order to determine the response curves of the image sensors used by the imaging device we first take an image of a calibration target. Figure 2 shows an IT8 calibration target made by Wolf Faust. This calibration target consists of 22×12 different colored patches. A gray scale with 24 different gray tones is located at the bottom ranging from white to black. Along with the calibration target comes a complete set of reflectances covering wavelengths from 390nm to 700nm in steps of 10nm.

Before processing the data it is important to linearize the data as described above, i.e. the gamma correction is undone. For each image patch we then compute the average pixel value of the pixels which belong to the patch. Pixels from the boundary of the patch are not included in the average as such pixels are considered to be linear combinations of the two adjacent colors. This is because of the Bayer pattern of the sensor that is used frequently. Such a sensor is also used inside the camera model we experimented with. Thus, for each image patch we now have the color $\mathbf{I}$ which was measured by the sensor. For each patch we also know the reflectances $R(x, y, \lambda)$. The irradiance $L(x, y, \lambda)$ can be measured using a spectrometer. Alternatively, we can also use a standardized illuminant where the spectral power distribution is known. When we take an image of the calibration target using a digital camera we have the option of specifying the type of illuminant used. The illuminant can either be set to a specific color temperature or it can be set to one of sun, cloudy sky, neon light, light bulb or flash. The camera then adjusts for the type of illuminant used. Unfortunately, in most cases, it is not known what internal processing is done by the camera. Since we assume that the camera corrects for the illuminant $L(\lambda)$, we have to solve the following equation for $\mathbf{S}(\lambda)$

$$\mathbf{I} = G \int \mathbf{S}(\lambda) R(\lambda) d\lambda \tag{7}$$

where we have omitted the index (x, y) which refers to the current patch.

The geometry factor G scales all color channels equally. Thus, we can remove this factor by computing chromaticities.

$$\hat{\mathbf{I}} = \frac{1}{I_r + I_g + I_b} \mathbf{I} \tag{8}$$

We will use an evolution strategy [25, 28, 12] to find the sensor response curves of the digital camera (a Canon EOS 10D) which took the image of the calibration target. Reflectance data is available from 390nm to 700nm in steps of 10nm. Therefore we can optimize for the sensor response curves $\mathbf{S}(\lambda)$ with $\lambda \in \{390\text{nm}, 400\text{nm}, ..., 700\text{nm}\}$. Evolution strategies work with real valued genotypes. Since we have three sensors, the genotype will consist of 3×32 floating point values. Let x_i be the values which have to be optimized, i.e. we have $x_1 = S_r(390)$, $x_2 = S_r(400)$, ... $x_{32} = S_r(700)$, $x_{33} = S_g(390)$, $x_{34} = S_g(400)$, ..., $x_{64} = S_g(700)$, $x_{65} = S_b(390)$, $x_{66} = S_b(400)$, ..., $x_{96} = S_b(700)$.

Evolution strategies mutate a genotype by adding a vector with normally distributed values. One can either use a single standard deviation for all values or one can use separate standard deviations for each value which has to be optimized. We will be working with separate standard deviations. This has the advantage that the standard deviation may be tuned to the parameter x_i. Let σ_i be the standard deviation for parameter x_i. In this case, mutation of an individual is defined as

$$\sigma_i := \sigma_i e^{N(0,\sigma_b)} \tag{9}$$

$$x_i := x_i + N(0, \sigma_i) \tag{10}$$

where σ_b is the standard deviation which is used to mutate the standard deviations σ_i. The genotype which we will be using is shown in Figigure 3. Half of the genotype consists of the standard deviations which are used to mutate the parameters x_i, the other half consists of the parameters x_i.

4. ERROR MEASURE AND INCORPORATING CONSTRAINTS

Each individual i represents a set of sensor response curves $\mathbf{S}(\lambda)$. Using these response curves we can compute the intensities such a sensor would measure for a given patch p.

$$\mathbf{I}_{\mathrm{i}}(p) = \sum_{\lambda \in \{390, \ldots, 700\}} \mathbf{S}(\lambda) R_p(\lambda) \tag{11}$$

Let $\hat{\mathbf{I}}_{\mathrm{i}}(p)$ be the corresponding chromaticities. Let $\hat{\mathbf{I}}_{\mathrm{act}}(p)$ be the actual chromaticities which were obtained from the image of the calibration target. Our error measure E_{fit} is then defined as

$$E_{\mathrm{fit}} = \sum_p \left(\hat{\mathbf{I}}_{\mathrm{i}}(p) - \hat{\mathbf{I}}_{\mathrm{act}}(p) \right)^2. \tag{12}$$

This measures how good the estimated sensitivities explain the actual data values. In other words, we compute the squared differences between the output which would be obtained using the response curves of the individual and the actual output which was obtained over all 288 image patches. We want to minimize this error measure.

In addition, it is clear from the problem statement that a sensor has a sensitivity larger than zero, i.e. $S(\lambda) \geq 0$ for all wavelengths λ. We could either try to enforce these constraints by augmenting the fitness function such that we also optimize for positive sensitivities. However, we can also

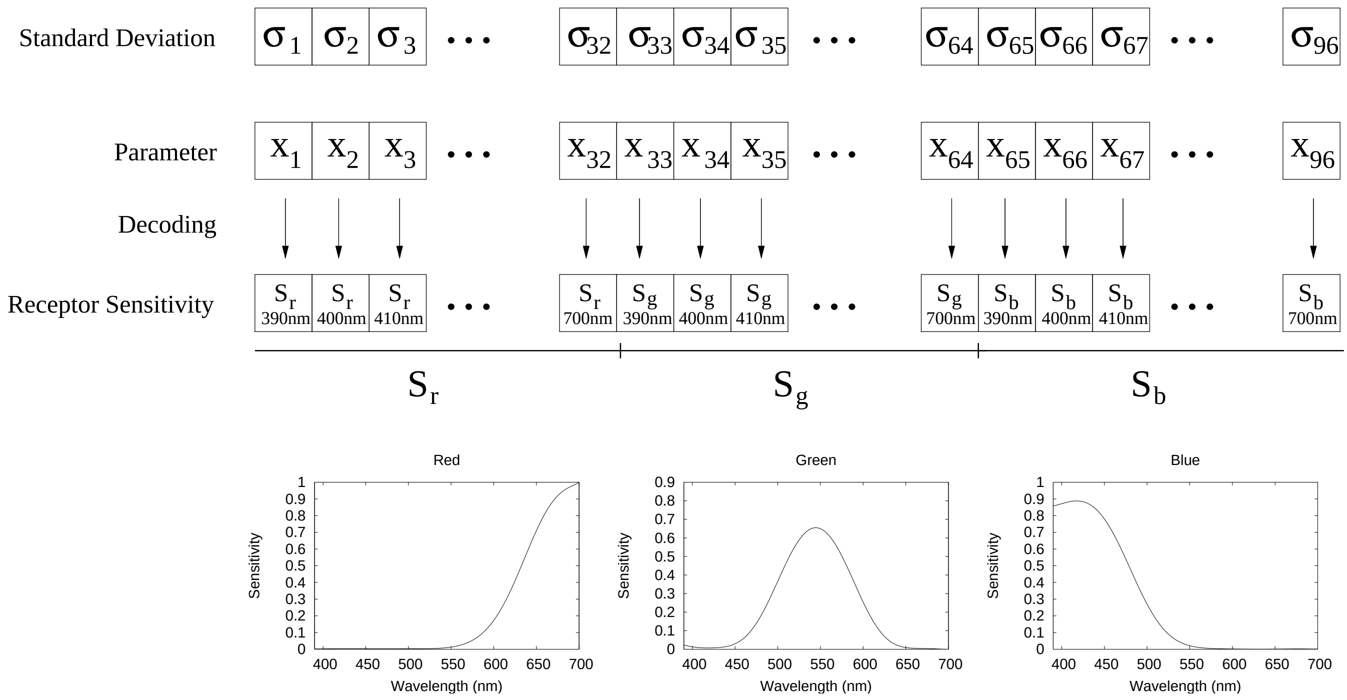

Figure 3: Each genotype represents three response curves for the three channels red, green, and blue. The genotype consists of 96 parameters x_i with $i \in \{1, ..., 96\}$ (32 per channel) and 96 standard deviations σ_i, which are used as mutation step sizes, i.e. we have a standard evolution strategy with automatic step size adaptation. Decoding of the individual is also shown. We have $x_1 = S_r(390)$, $x_2 = S_r(400)$, ..., $x_{96} = S_b(700)$.

enforce this constraint directly. In this case, we set a parameter to zero should it become negative.

$$x_i := \begin{cases} x_i & \text{if } x_i \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad \sigma_i := \begin{cases} \sigma_i & \text{if } x_i \geq 0 \text{ or } \sigma_i \leq \sigma_{\min} \\ \sigma_{\min} & \text{otherwise} \end{cases} \tag{13}$$

Since we compare only chromaticities, the result is invariant to uniform scaling of all sensitivities. This gives us the constraint that the maximum sensitivity can be normalized to 1. Again this could be incorporated into the fitness function or we could enforce it directly. In the latter case, we could - after mutation of individuals - normalize the sensitivities of the individual such that the maximum sensitivity is 1.

$$x_i := x_i / s \qquad \text{where } s = \max\{x_i | i \in \{1, ..., 96\}\} \tag{14}$$

We may also try to enforce a smoothness constraint, assuming that the sensitivities of the sensors vary smoothly. This could also be incorporated into the fitness function. Alternatively, we could enforce the smoothness constraint by smoothing neighboring parameters of the individual. Let p be a smoothing parameter. Then we could smooth neighboring parameters as follows.

$$x_i := \begin{cases} (1 - \frac{1}{2}p)x_i + \frac{1}{2}px_{i+1} & \text{if } i = 1, 33 \text{ or } 65 \\ (1 - \frac{1}{2}p)x_{i-1} + px_i + (1 - \frac{1}{2}p)x_{i+1} & \text{otherwise} \\ \frac{1}{2}px_{i-1} + (1 - \frac{1}{2}p)x_i & \text{if } i = 32, 64 \text{ or } 96 \end{cases} \tag{15}$$

Instead of enforcing the constraints by modifying the genotype, we could try to enforce these constraints via the fitness function. We have already defined the main error measure E_{fit} above. We could augment this error measure by incorporating additional terms which measure the deviation below zero E_{zero}, normalization E_{norm} and the smoothness of the sensitivity E_{smooth}. These components can be defined as

$$E_{\text{zero}} = \sum_i y_i^2 \qquad \text{with } y_i = \begin{cases} x_i & \text{if } x_i < 0 \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

$$E_{\text{norm}} = (1 - s)^2 \tag{17}$$

$$\text{where } s = \max\{x_i | i \in \{1, ..., 96\}\}$$

$$E_{\text{smooth}} = \sum_{b \in \{0,1,2\}} \sum_{i \in \{2,...,32\}} (x_{32b+i-1} - x_{32b+i})^2 \tag{18}$$

The overall error measure would then be given by

$$E = w_1 E_{\text{fit}} + w_2 E_{\text{zero}} + w_3 E_{\text{norm}} + w_4 E_{\text{smooth}} \tag{19}$$

where the weights w_i may be set to influence the relative importance of the constraints.

5. EXPERIMENTAL RESULTS

For our experiments, we have used a $(100, 500)$ evolution strategy [25, 28, 12], i.e. 100 parents produce 500 offspring. For each offspring, two parents are randomly selected and a 2 point crossover operator is applied. Only one offspring is actually used. Then the mutation operator is applied as described above. Parameters of x_i were initialized uniformly from the range $[0, 1]$. The standard deviations were initialized randomly from the range $[0.01, 0.0001]$ and σ_b was set to 0.01. We are using a comma evolution strategy. The best 100 are selected among the offspring as parents for the next generation. Using this scenario we ran 7 experiments.

Table 1: List of experiments. Different types of fitness functions and constraints on the genotype of the individual were used. For each experiment, 10 individual evolutionary runs of 1000 generations were made.

Exp.	Fitness	Directly Enforced Constraints
A	$E_A = E_{\text{fit}}$	none
B	$E_B = E_{\text{fit}} + E_{\text{zero}} + E_{\text{norm}} + E_{\text{smooth}}$	none
C	$E_C = w_1 E_{\text{fit}} + w_2 E_{\text{zero}} + w_3 E_{\text{norm}} + w_4 E_{\text{smooth}}$	none
D	Pareto optimization using $(E_{\text{fit}}, E_{\text{zero}}, E_{\text{norm}}, E_{\text{smooth}})$	none
E	$E_E = E_{\text{fit}}$	Eqn. 13
F	$E_F = E_{\text{fit}}$	Eqn. 13, Eqn. 14
G	$E_G = E_{\text{fit}}$	Eqn. 13, Eqn. 14, Eqn. 15

The different experiments are summarized in Table 1. For experiment A, the error measure was solely determined by the fit to the data. For experiment B, the error measure was set to the sum of the error measures describing the fit to the data, the positivity of the parameters, the normalization, and the smoothness. For experiment C, we also used the sum over the different error measures, however this time each term is weighted such that the sum of a single term over all individuals of the population is equal to 1. In other words, each term is equally important. Experiment D uses pareto optimization [14, 13]. Each individual is assigned a rank according to the number of individuals which dominate that individual. Best individuals are assigned rank 0. Individuals which are dominated by one other individual are assigned rank 1 and so on. Note that, for each generation, we are selecting 100 parents among 500 offspring. It may well be that more than 100 individuals have rank 0. In this case, we have to decide which individuals are actually selected. We resolve this problem by augmenting the rank r_i of the individual i with a percentage p_i which is computed as follows

$$p_i = \frac{E_C(i)}{4} \tag{20}$$

where $E_C(i)$ is the error measure computed for individual i of the population as described in experiment C. Since each term of E_C is less than or equal to one, we have $0 \leq p_i \leq 1$. Then the augmented rank $\hat{r}_i$ is given by $\hat{r}_i = r_i + p_i$. Selection is done using $\hat{r}$.

Experiments E through G all use constraints on the genotype. For experiment E, we have limited all parameters to positive values using Eqn. 13. For experiment F, we have used the positivity constraint by applying Eqn. 13 and we also used the normalization by applying Eqn. 14. For experiment G, we have used the constraints of Eqn. 13 and Eqn. 14 and in addition also used the smoothness constraint of Eqn. 15.

For each experiment, 10 runs with different seeds for the random number generator were carried out. Figure 4 shows the minimum, average and maximum of the error measure E_{fit} over all generations shown for the run which produced the lowest error measure E_{fit}. Figure 5 shows the results we obtained. For each experiment, the best individual (according to E_{fit}) from all 10 runs is shown. For experiment D, we chose the individual having minimum distance to the origin using the same weighting as was used for experiment C among the individuals of the pareto front. The standard deviation of the minimum obtained error measure E_{fit} was 0.095, 0.169, 4.209, 3.583, 0.210, 0.116, and 0.412 for experiments A, B, C, D, E, F, and G respectively. Experiments C, and D have a larger standard deviation in the results obtained. This is of no surprise as for these experiments, multi-objective optimization was used. The minimum error measure E_{fit} for the data over all 10 runs was 7.092, 8.674, 52.764, 22.258, 10.623, 10.334, 15.441 for experiments A through G. It is clear that the results obtained in experiment A do not describe a physical sensor which responds positively to light. There are a large number of wavelengths for which the sensor responds negatively. Incorporating additional constraints into the fitness function provides only limited success. Constraining the genotype as done in experiment G works best. We now obtain a smooth sensor response function and see how the sensors actually respond to light in the red, green and blue parts of the spectrum.

6. CONCLUSION

We have used an evolution strategy to find the response curves of the sensors used inside a digital camera. Individuals were represented as floating point vectors describing the sensitivity of the sensors for each wavelength. Obtaining sensor response curves is not as straight forward as our experiments have shown. Although a very good fit to the data could be obtained, this was not enough. Constraints had to be enforced on the genotype in order to obtain plausible response curves.

7. REFERENCES

[1] B. E. Bayer. Color imaging array. *United States Patent No. 3,971,065*, July 1976.

[2] G. Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin Institute*, 310(1):337–350, July 1980.

[3] P. Carvalho, A. Santos, A. Dourado, and B. Ribeiro. On the estimation of spectral data: a genetic algorithm approach. In *Proceedings of the IEEE International Conference on Image Processing, Thessaloniki, Greece, October 7-10*, pages 866–869. IEEE, 2001.

[4] P. Cerveri, A. Pedotti, and N. A. Borghese. Combined evolution strategies for dynamic calibration of video-based measurement systems. *IEEE Transactions on Evolutionary Computation*, 5(3):271–282, 2001.

[5] C. A. C. Coello. A survey of constraint handling techniques used with evolutionary algorithms. Technical Report Lania-RI-99-04, Laboratorio Nacional de Informática Avanzada, Xalapa, Veracruz, Mexico, 1999.

[6] C. A. C. Coello. The use of a multiobjective optimization technique to handle constraints. In A. A. O. Rodriguez, M. R. S. Ortiz, and R. S.

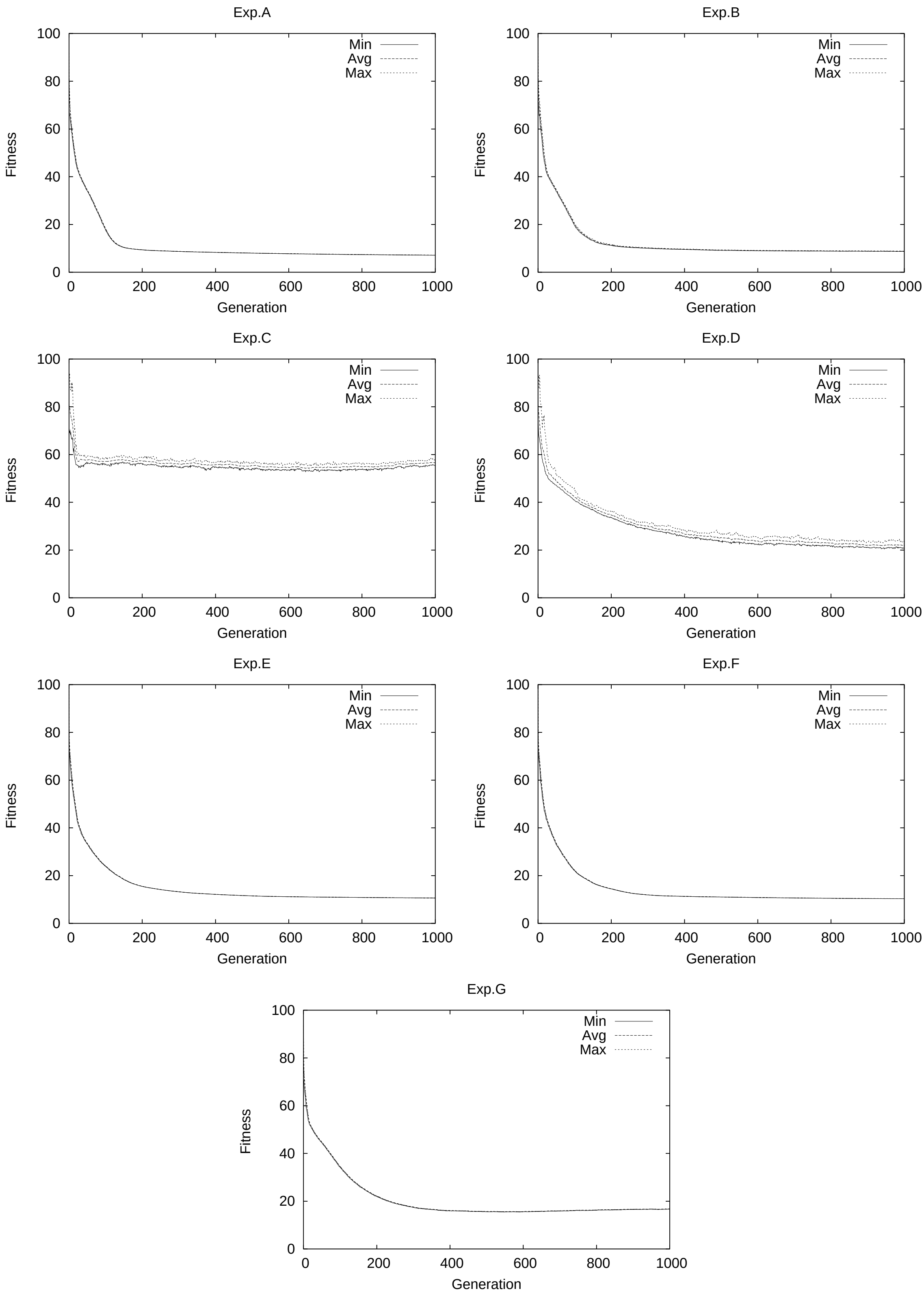

Figure 4: Minimum, average and maximum of the error measure E_{fit} over all generations shown for the run which produced the lowest error measure.

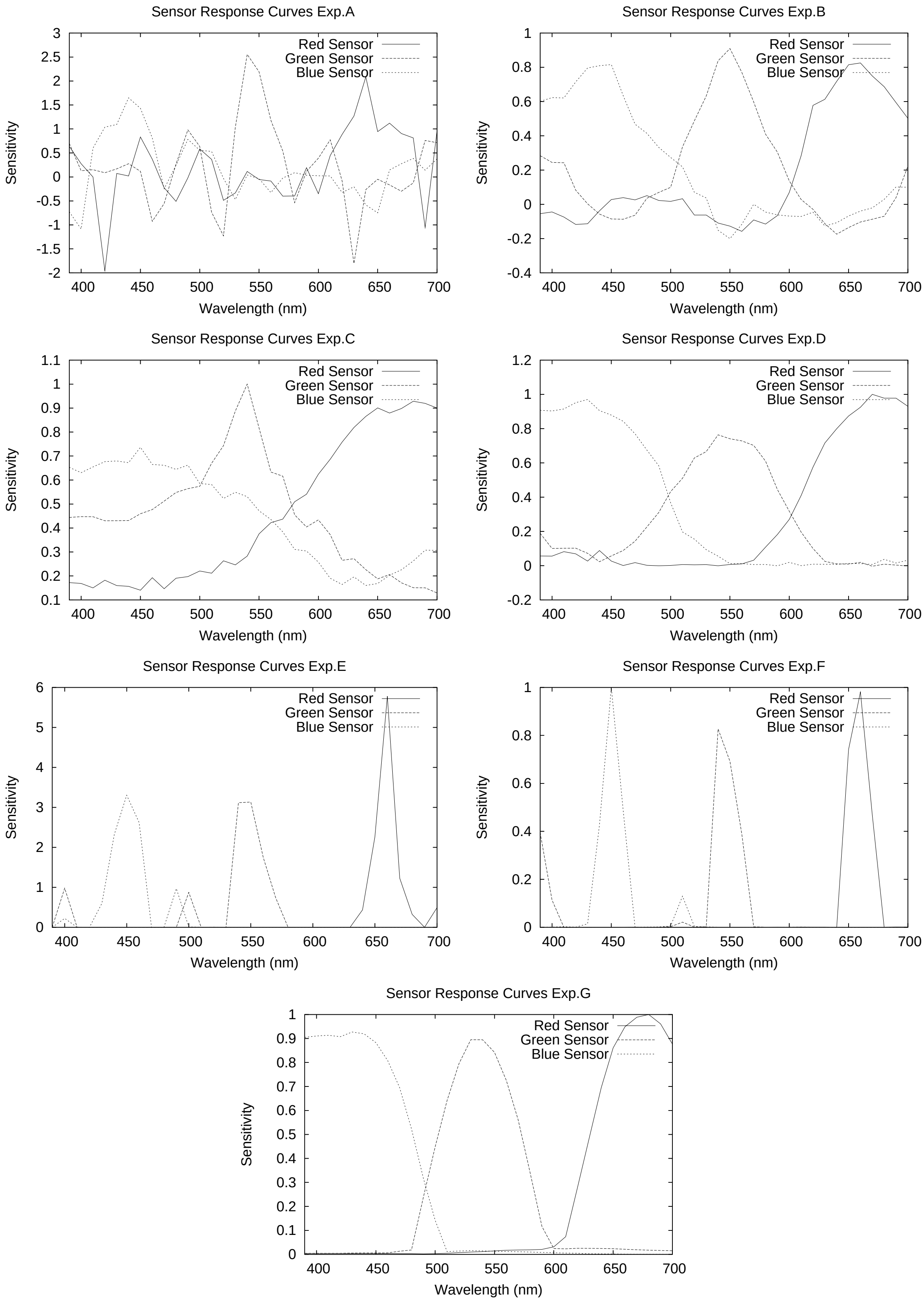

Figure 5: Results for experiments A through G.

Hermida, editors, *Proceedings of the Second International Symposium on Artificial Intelligence (Adaptive Systems), Habana, Cuba*, pages 251–256. Institute of Cybernetics, Mathematics and Physics, Ministry of Science, Technology and Environment, July 1999.

[7] K. Deb. An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering*, 186(2/4):311–338, 2000.

[8] G. D. Finlayson. Color in perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(10):1034–1038, Oct. 1996.

[9] G. D. Finlayson, M. S. Drew, and B. V. Funt. Color constancy: generalized diagonal transforms suffice. *Journal of the Optical Society of America A*, 11(11):3011–3019, Nov. 1994.

[10] G. D. Finlayson and S. D. Hordley. Color constancy at a pixel. *Journal of the Optical Society of America A*, 18(2):253–264, Feb. 2001.

[11] G. D. Finlayson, S. D. Hordley, and M. S. Drew. Removing shadows from images. In *Proceedings of the European Conference on Computer Vision*, pages 823–836, Berlin, 2002. Springer-Verlag.

[12] D. B. Fogel. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. IEEE Press, New York, 2000.

[13] C. M. Fonseca and P. J. Fleming. Multiobjective genetic algorithms made easy: Selection, sharing and mating restriction. *First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications*, 414:45–52, 1995.

[14] C. M. Fonseca and P. J. Fleming. An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation*, 3(1):1–16, 1995.

[15] D. A. Forsyth. A novel approach to colour constancy. In *Second International Conference on Computer Vision (Tampa, FL, Dec. 5-8)*, pages 9–18. IEEE Press, 1988.

[16] Foveon Inc. X3 technology. why X3 is better, 2002.

[17] S. B. Hamida and M. Schoenauer. An adaptive algorithm for constrained optimization problems. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. J. Merelo, and H.-P. Schwefel, editors, *Proceedings of the 6th Parallel Problem Solving From Nature (PPSN VI), Paris France*, pages 529–538, 2000.

[18] F. Hoffmeister and J. Sprave. Problem-independent handling of constraints by use of metric penalty functions. In L. J. Fogel, P. J. Angeline, and T. Bäck, editors, *Proceedings of the Fifth Annual Conference on Evolutionary Programming (EP'96), San Diego, CA*, pages 289–294. The MIT Press, 1996.

[19] International Commission on Illumination. Colorimetry, 2nd edition, corrected reprint. Technical Report 15.2, International Commission on Illumination, 1996.

[20] C. M. Johnson, A. Bhat, and W. C. Thibault. An evolutionary approach to camera-based projector calibration. In *Proceedings of the Genetic and Evolutionary Computation Conference 2006, Seattle, Washington, July 8-12*, pages 1871–1872. ACM, 2006.

[21] A. F. Kuri-Morales and J. Gutiérrez-García. Penalty function methods for constrained optimization with genetic algorithms: A statistical analysis. In C. A. C. Coello, A. de Albornoz, L. E. Sucar, and O. C. Battistutti, editors, *Proceedings of the 2nd Mexican International Conference on Artificial Intelligence (MICAI 2002), Merida, Yucatan, Mexico*, pages 108–117, Heidelberg, 2002. Springer-Verlag.

[22] Z. Michalewicz. A survey of constraint handling techniques in evolutionary computation methods. In J. R. McDonnell, R. G. Reynolds, and D. B. Fogel, editors, *Proceedings of the 4th Annual Conference on Evolutionary Programming*, pages 135–155, Cambridge, Massachusetts, 1995. The MIT Press.

[23] Z. Michalewicz and M. Schoenauer. Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary Computation*, 4(1):1–32, 1996.

[24] C. Poynton. *Digital Video and HDTV. Algorithms and Interfaces*. Morgan Kaufmann Publishers, San Francisco, CA, 2003.

[25] I. Rechenberg. *Evolutionsstrategie '94*. frommann-holzboog, Stuttgart, 1994.

[26] V. Rodehorst and O. Hellwich. Genetic algorithm sample consensus (gasac) - a parallel strategy for robust parameter estimation. In *International Workshop '25 Years of RANSAC', New York, USA*. IEEE, June 2006.

[27] R. Sarker, H. A. Abbass, and S. Karim. An evolutionary algorithm for constrained multiobjective optimization problems. In *The 5th Australasia-Japan Joint Workshop on Intelligent and Evolutionary Systems (AJWIS 2001), Dunedin, New Zealand*, pages 113–122, 2001.

[28] H.-P. Schwefel. *Evolution and Optimum Seeking*. John Wiley & Sons, New York, 1995.

[29] A. E. Smith and D. W. Coit. Penalty functions. In T. Bäck, D. Fogel, and Z. Michalewicz, editors, *Handbook of Evolutionary Computation*, page Section C 5.2. Oxford University Press and Institute of Physics Publishing, 1997.

[30] M. Stokes, M. Anderson, S. Chandrasekar, and R. Motta. A standard default color space for the internet - sRGB. Technical report, Version 1.10, 1996.

[31] D. A. G. Vieira, R. L. S. Adriano, L. Krähenbühl, and J. A. Vasconcelos. Handling constraints as objectives in a multiobjective genetic based algorithm. *Journal of Microwaves and Optoelectronics*, 2(6):50–58, 2002.

[32] B. A. Wandell. The synthesis and analysis of color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(1):2–13, Jan. 1987.

[33] G. Wyszecki and W. S. Stiles. *Color Science. Concepts and Methods, Quantitative Data and Formulae*. John Wiley & Sons, Inc., New York, second edition, 2000.

[34] Ö. Yeniay. Penalty function methods for constrained optimization with genetic algorithms. *Mathematical and Computational Applications*, 10(1):45–56, 2005.

[35] Y. Zhang and Q. Ji. Camera calibration with gentic algorithms. In *Proceedings of the 2001 IEEE International Conference on Robotics & Automation, Seoul, Korea, May 21-26*, pages 2177–2182. IEEE, 2001.