Bayesian Methods for Radiometric Calibration in Motion Picture Encoding Workflows

By Ricardo R. Figueroa and Rui Li

Abstract

The accurate estimation of a camera response function (CRF) allows for proper encoding of camera exposures into motion picture post-production workflows like the Academy Color Encoding System (ACES), helping minimize noncreative manual adjustments. Although there are well-known standard CRFs implemented in typical video camera workflows, motion picture camera workflows and new high dynamic range (HDR) workflows have introduced new standard CRFs, as well as custom and proprietary CRFs. Current methods to estimate this function rely on the use of test charts, restrictive exposure and/or lighting conditions, or assume a simplistic model of the function's shape. All of these methods become problematic and tough to fit into motion picture production and post-production workflows, where the use of test charts and varying camera or scene setups becomes impractical. We propose a method initially based on the work of Lin, Gu, Yamazaki, and Shum that considers edge color mixtures in an image or image sequence that are affected by the nonlinearity introduced by a CRF. This feature is then used in a Bayesian framework to estimate a posterior probability distribution function of the CRF model parameters approximated by a Markov Chain Monte Carlo (MCMC) algorithm, allowing for a more robust description of the CRF over methods like maximum likelihood (ML) and maximum a posteriori (MAP).

We propose a method initially based on the work of Lin, Gu, Yamazaki, and Shum that considers edge color mixtures in an image or image sequence, that are affected by the nonlinearity introduced by a CRF.

Keywords

Camera response function, gamma correction, optoelectronic camera function, radiometric calibration

Introduction

Digital processes currently generate large amounts of data. These processes exist in a large number of applications, and, in most cases, the generated data exists by itself without any information or details about the process that created it. It is of interest to use computers to analyze the data, learn information about the process, and be able to predict future data from an estimation of the process that generated that data. This is currently known as machine learning.

One of such processes is the capture of images by a digital system. Different cameras have different ways of encoding the light being captured from a scene. This encoding converts the light falling onto the imaging sensor into digital pixel code values that are then manipulated in post-production. One of the most important steps in this encoding is the camera response function (CRF), which allows the camera to have more efficient encoding and to encode the light from the scene in a way that mimics the human visual system. Another important typical step following the implementation of the CRF is a color encoding transform that ensures subsequent appropriate display color reproduction. Although the main objective of the CRF is similar for all cameras, different cameras implement different types of functions due to different existing standards, advancements in camera technology, proprietary advantages, and image sensor characteristics, to name a few reasons. This means that the CRF is one of the reasons why time has to be spent making technical adjustments before images look uniform for subsequent creative manipulations.

These adjustments are a time-consuming endeavor that can be optimized by implementing methods to estimate the camera specific characteristics that created the images, like the CRF, without the need for specific camera analysis procedures that are not viable or practical in post-production environments. Typical television or motion picture productions involve different image manipulation workflows that are very challenging to integrate. Many of these productions can use various digital cameras and also include film in their production. Additionally, during post-production, images from all these sources will arrive at a facility in a number of different formats, color encodings, and, in many cases,

Digital Object Identifier 10.5594/JMI.2019.2917595
Date of publication: 23 July 2019

without helpful metadata. An example system to solve some of these issues is the Academy Color Encoding System (ACES) and its Input Device Transform (IDT), which aims at characterizing every possible input so it can be integrated into the ACES workflow. The CRF is a key camera characteristic that needs to be known for the calculation and/or implementation of the IDT since it relates camera code values to linear scene radiometry and is critical for these calculations to be accurate.

Our work focuses on developing a method to estimate a motion picture CRF from an image or a sequence of images and evaluate its impact in computing and implementing an accurate motion picture color encoding transformation. Although camera characteristics like the CRF are easily measurable in a research environment, they are very hard to measure during production or post-production where the time is short and other processes take priority. Furthermore, modern cameras offer the user a wide range of options to modify some of these characteristics, making the interchange of images even more challenging during post-production.

Previous Works

Aside from measuring the CRF through direct photography of a variable reflectance or transmittance test chart, some of the most popular methods for estimating the CRF involve taking multiple registered images of a static scene while varying the camera exposure.[1,2] Work has also been done by Grossberg and Nayar[3] to relax this dependence on multiple spatially registered images by using histograms of images at different exposures. Manders et al.[4] estimate the response by capturing registered images of a static scene illuminated by different combinations of light sources and locking the exposure instead of varying it. Similar work was presented by Kim et al.[5] where the estimation is done based on a video sequence with varying exposure. All of these methods require significant effort to obtain a series of registered, static scene, or exposure-dependent images. Other approaches, like Farid,[6] assume that the response function has the form of a gamma curve to estimate it from one single image by exploiting the fact that the nonlinearity introduced by the gamma correction also introduces higher order correlations in the frequency domain that can be detected with tools from the polyspectral analysis. However, many cameras in use today in the motion picture industry have CRFs that differ significantly from a gamma curve, especially higher end motion picture cameras where their design deviates from the typical video imaging paradigm. Lin et al.[7] estimate the response by looking at the edge color distributions in a single image. This work serves as a basis for our research as we look to expand its methodology by analyzing edge color distributions in a sequence of motion images and expand their maximum a posteriori (MAP) approach to a complete Bayesian framework.

More recent work has also focused on using multiple images, especially photo collections. Diaz and Sturm[8] use a set of images from an internet collection to recover the camera's geometric calibration and a 3D scene model, which in turn are used as input to determine the camera's radiometric response function. The requirement to obtain the 3D scene model would be an undesired workflow step for motion picture post-production workflows focused on image integration and color correction. Shafique and Shah[9] introduce a method that uses differently illuminated images and estimates the response function by assuming that the properties of materials in a scene should remain constant and use cross-ratios of image values in the different color channels to compute the response function. They also model the response function as a gamma curve. This again excludes the modeling of cameras that have CRFs that differ from a gamma curve and their algorithm was only verified by synthetic experiments. Kuthirummal et al.[10] found priors for statistics in large photo collections and used them to estimate the response function of generic camera models assuming that all instances of a camera model have the same properties and that many images are available for that specific camera. A major disadvantage of their method is that it does not allow modeling of cameras with interchangeable lenses. This is also an impractical approach for the motion picture industry.

In summary, the existing methods described suffer from the following general practical limitations in their application to the motion picture industry: they are restrictive in the position or movement of the capture device, they are restrictive in the exposure required for the camera sensor, and they wrongly assume a simplistic mathematical formulation of the CRF that does not characterize all possible CRFs or they require co-capture of images and some additional scene information. Our proposed approach explained in the following sections again looks to expand on the work of Lin et al. to avoid all of these limitations and provide a solution based on unrestricted captured images that are typical of the motion picture industry.

Background

The radiometric camera response function (CRF) f relates captured scene radiance I, or proportional image irradiance, to its measured intensity M in the image, represented by the pixel code value:

$$M = f(I). \tag{1}$$

Typically, radiometric calibration methods solve for the inverse response function $g = f^{-1}$, which is invertible since the sensor output increases monotonically with respect to I. Popular previous works relying on multiple exposures compute the inverse response function based on the relationship:

$$g(m_A) = kg(m_B). \tag{2}$$

where m_A and m_B represent image intensities in images A and B, respectively, for corresponding points, and k denotes the exposure ratio between the two images. Requirements like precise registration of images or known exposure ratios have been worked around with the use of histogram equalization and the use of iterative methods, referenced in the previous section, to solve for k and g.

One important obstacle in computing the camera radiometric response function from Eq. 2 is the exponential ambiguity. From Eq. 2, it can be seen that, if g and k are solutions for a specific image set, then g^{μ} and k^{μ} could also be solutions. To deal with this ambiguity, these prior methods require a good initial estimate of the exposure ratio if unknown, and assumptions on the structure of the radiometric function model as presented by Grossberg and Nayar.[11] Their results present a relationship between image intensity $M = f(I)$ and image irradiance I that can be expressed as a linear combination of an average CRF h_0 and principal components h_n:

$$f(I) = h_0(I) + \sum_{n=1}^{N} w_n h_n(I). \tag{3}$$

This principal component analysis (PCA) approach is utilized to model the CRF in combination with image features like measured image intensity characteristics that get affected by a nonlinear radiometric response, edge color distributions being an example as presented by Lin et al.[7] In **Fig. 1**, we can see the measured nonlinear distributions of edge colors produced by the nonlinear CRF. This warping of image irradiance colors into a nonlinear measured color distribution provides us with an image feature that can be used to estimate the inverse CRF as it is explained in the next section.

Edge Color Distributions

Single sensor (charged coupled device or CMOS) cameras, like most used in the motion picture industry, require a color filter array to capture color images. Each array element, or pixel x, images a solid angle of the scene and we can denote the whole array of pixels as $S(x)$. For each x, the image irradiance I depends on the sensitivity q_k of the pixel and filter pair, where k is the color of the filter, and the incoming scene radiance R incident on the image plane points p included in $S(x)$:

$$I(x,k) = \int_{\lambda}\int_{s(x)} R(p,\lambda)\, q_k(\lambda)\, dp\, d\lambda. \tag{4}$$

Here, λ represents the range of wavelengths that are transmitted by the color filter k. It is assumed that all three colors, i.e., red, green, and blue, are measured for each pixel position because of demosaicing in the camera.

To analyze edge color distributions, we start by considering two regions of distinct but uniform color like those in **Fig. 1**. A pixel x imaging a scene area where the two regions come together on an edge will be receiving scene radiances $R_1(\lambda)$ and $R_2(\lambda)$, respectively, from region 1 and region 2, and the overall radiance incident on it can be expressed as:

$$\begin{aligned}\int_{S(x)} R(p,\lambda)dp &= \int_{S_1(x)} R_1(\lambda)dp + \int_{S_2(x)} R_2(\lambda)dp \\ &= \alpha R_1(\lambda) + (1-\alpha) R_2(\lambda),\end{aligned} \tag{5}$$

where $\alpha = \int_{S_1(x)} dp$ and $S(x)$ is of unit area. Looking back at Eq. 1, if we substitute Eq. 4 and Eq. 5 into it, the measured color at pixel x will be:

$$m(x,k) = f\left[\alpha I_1(x,k) + (1-\alpha) I_2(x,k)\right]. \tag{6}$$

For a linear relationship between the image irradiance I and the measured color $f(I)$ to exist, the following property would hold:

$$f[\alpha I_1 + (1-\alpha) I_2] = \alpha f(I_1) + (1-\alpha) f(I_2).$$

This means that the measured colors of pixels that lie on an edge would fall on a straight line in the RGB color space between both regions. However, since f is typically nonlinear, the measured edge colors form a curve, rather than a straight line, that provides information on the CRF. We use this nonlinearity of the measured edge color distributions to estimate the CRF, based on the assumption that the inverse CRF should transform the measured values into values linearly related to image irradiance. For an edge region with measured colors M_1 and M_2, the inverse CRF g should map the measured color M_p of each point p in the region to a line defined by $g(M_1)$ and $g(M_2)$. **Figure 2** shows this graphically. A function g satisfies this property if the distance from $g(M_p)$ to line $g(M_1)$ $g(M_2)$ is zero as formulated by Eq. 7, where × is the cross-product operation between the two vectors.

$$\frac{\left|\left[g(M_1) - g(M_2)\right] \times \left[g(M_p) - g(M_2)\right]\right|}{\left|g(M_1) - g(M_2)\right|} = 0 \tag{7}$$

We collect all obtained edge color triplets into a set $\Omega = \{M_1, M_2, M_P\}$ from all images considered and then define the total distance as:

$$D(g;\Omega) = \sum_{\Omega} \frac{\left|\left[g(M_1) - g(M_2)\right] \times \left[g(M_p) - g(M_2)\right]\right|}{\left|g(M_1) - g(M_2)\right|}. \tag{8}$$

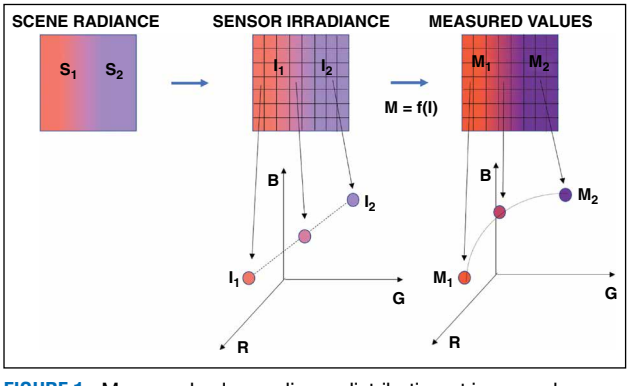

FIGURE 1. Measured color nonlinear distribution at image edges (adapted from Lin et al.[7]).

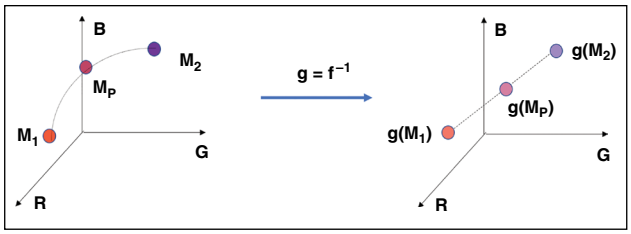

FIGURE 2. Nonlinear to linear distribution transformation (adapted from Lin et al.[7]).

The best inverse response g is one that results in the smallest total distance among all triplets selected.

Bayesian Framework

Probability-based estimation methods have been used for many years and in many applications.[12] Two popular solution approaches for these estimation methods are maximum likelihood (ML) and maximum a posteriori (MAP).[13,14] In the case of ML, the result produces the choice most likely to have generated the available data. MAP produces the choice that is most likely, given the available data. An important difference is that MAP estimation applies Bayes' Rule, shown in Eq. 9, so that the estimate can consider prior parameter knowledge in the form of a prior probability distribution, $p(g)$, while evaluating the likelihood of the data, $p(\Omega|g)$, generated by those parameters. MAP can be considered an improvement over ML, but both ML and MAP give only single point best estimates and not a distribution of the parameters in question, although the benefit is that these methods typically compute this single best estimate in a fast and efficient manner:

$$p(g|\Omega) = \frac{p(\Omega|g)p(g)}{p(\Omega)} = \frac{p(\Omega|g)\ p(g)}{\int_g p(\Omega|g)\ p(g)dg}. \quad (9)$$

Prior Probability Distribution

The prior model was created from the Database of Response Functions (DoRF) compiled by Grossberg and Nayar.[11] This database contains 201 inverse response functions from a variety of digital cameras and films up to the year 2003. The prior probability distribution based on this database, $p(g)$, describes prior knowledge about the mathematical description and space of CRFs and is modeled as a Gaussian Mixture Model (GMM), with means and covariances μ_i, Σ_i for each mixture component i and mixture proportion α_i. In our implementation, the prior probability distribution was precomputed using the Expectation Maximization (EM) algorithm:

$$p(g) = \sum_{i=1}^{K} \alpha_i N(g;\mu_i,\Sigma_i). \quad (10)$$

Likelihood Probability Distribution

From Eq. 8, the estimated inverse CRF should yield a low total distance for the right inverse CRF. The likelihood probability distribution, $p(\Omega|g)$, can then be modeled by incorporating this distance calculation into an exponential distribution with λ set empirically and Z being a normalization constant:

$$p(\Omega|g) = \frac{1}{Z} exp\left(-\lambda D(g;\Omega)\right). \quad (11)$$

Bayesian MAP Solution

After modeling the prior and the likelihood functions as described in the previous section, the problem can be formulated with a MAP solution approach as described in Eqs. 12 and 13. The optimal response function for the data set Ω is then described as:

$$g = \text{argmax}\ p(g|\Omega) = \text{argmax}\ p(\Omega|g)p(g). \quad (12)$$

Or taking the log of (12), g can also be written as:

$$g = \text{argmin}\ \lambda D(g;\Omega) - \log p(g). \quad (13)$$

This again represents a single point best estimate of the inverse CRF, g, and serves as the main improvement comparison of our methodology.

Methodology

Posterior Probability Distribution

As mentioned previously, the benefit of ML and MAP is that these methods compute a single best estimate that can be easily and efficiently calculated, but this comes at the cost of throwing away information included in the posterior probability distribution, $p(g|\Omega)$ in Eq. 9, due to the cost of computing the integral in the denominator over a high-dimensional space. To do this, we realize that the integral in the denominator in Eq. 9 is an expected value calculation that can be approximated, allowing us to estimate the posterior distribution of g rather than just a single best estimate. This expectation is shown in Eq. 14 for the general continuous case:

$$E[f(z)] = \int f(z)\ p(z)\ dz. \quad (14)$$

Here, z is the random variable, and $p(z)$ is the probability distribution over possible values of z. In our case, we are interested in calculating the expected value of $f(z) = p(\Omega|g)$, which is the probability of the data given g, the likelihood. This expectation is taken over the whole distribution of possible values of g, $p(g)$, the prior probability density function.

Markov Chain Monte Carlo Sampling

The problem in calculating the expected value in Eq. 14 arises from the fact that computing the integral in the denominator of Eq. 9 is in many cases an intractable problem.[13] Conceptually, this integral sums up $p(\Omega|g)$ $p(g)$ over all values of g in a highly dimensional space. Since we are interested in approximating this integral because of the high dimensionality, another way to think about the calculation is to sample N points z (1), $z(2)$, $z(3)$ ... $z(N)$ from the distribution $p(\Omega|g)$ at random, with respect to the prior probability density

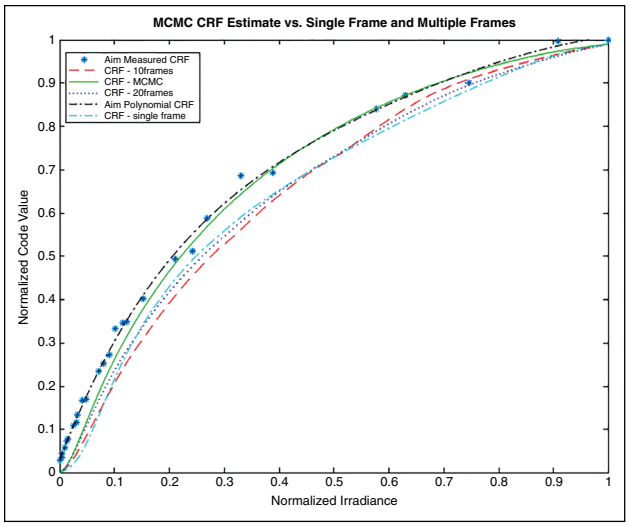

FIGURE 3. Camera response function estimation results comparison.

function $p(g)$, giving us the expected value in the following form:

$$E_{p(z)}[f(z)] = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} f(z^{(t)}). \quad (15)$$

By thinking of the problem in this way, we can implement a Markov Chain Monte Carlo (MCMC) sampling algorithm, like the Metropolis–Hastings algorithm, [13,14] to approximate the posterior probability distribution for g given a set of image pixel data Ω.

Results

We implemented a Bayesian MCMC algorithm to estimate the posterior distribution of the CRF following the methodology presented in the previous section for images captured with an ARRI digital motion picture camera. A Metropolis–Hastings algorithm was implemented to sample the space of the inverse CRF, g, and calculate the posterior probability distribution. Five PCA components were used in the PCA CRF model, making the sampling approach a 5D sampling problem. In this preliminary implementation of the algorithm, it was optimized by looking at the range covered by the PCA coefficients based on the DoRF to optimize the PCA CRF model sampling space. We assumed equal CRFs for the red, green, and blue channels. **Figure 3** shows the CRF (inverse of g) expected value for the green channel after three million sampling iterations, compared with the actual measured CRF for the camera and the estimations based on the MAP approach. RMS errors between the MCMC CRF estimation and the aim curves were 0.028, 0.021, and 0.021 for the red, green, and blue channels, respectively, showing significant improvement over the MAP approach,[15] which produced RMS errors of 0.077, 0.047, and 0.044 for the same data. **Figure 4** shows the MCMC approximation of the PCA coefficients CRF model target functions that represent the 5D posterior probability distribution function of g.

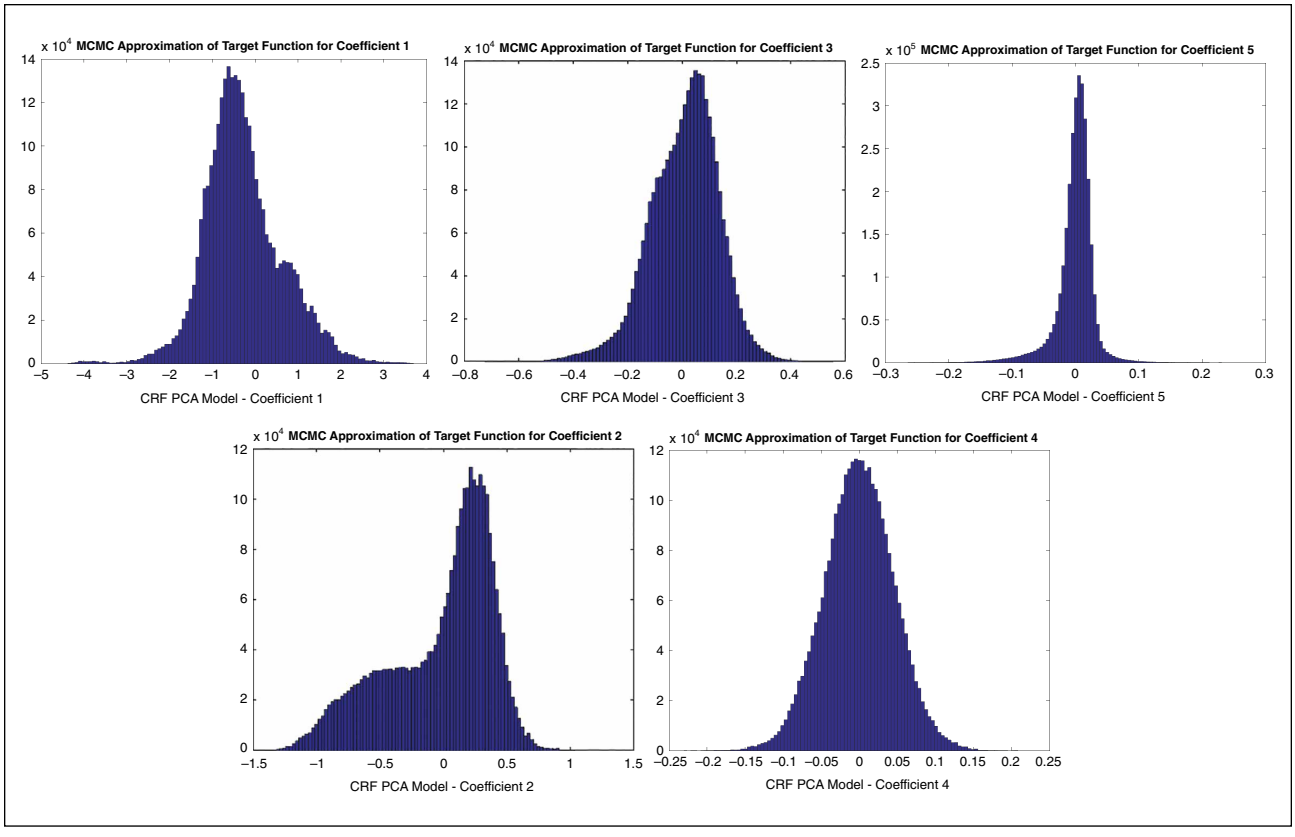

FIGURE 4. MCMC posterior distribution approximation of CRF coefficients 1–5.

Conclusion

Initial results estimating the posterior probability distribution function were very encouraging. The estimation accuracy improvement over MAP is evidence of the effectiveness of implementing a complete Bayesian framework. Immediate future work includes optimization of the MCMC sampling algorithm for more efficient estimation of the CRF. By optimization, we include domain-specific knowledge into the MCMC sampling algorithm with the goal of limiting the sampling space while still accurately estimating the CRF. This is important to be able to estimate independent CRFs for the red, green, and blue channels in an efficient manner. To this end, we will look at updating the 2003 DoRF with current camera systems CRFs and customizing the DoRF for specific motion picture workflow applications and then evaluate the performance of the MCMC sampling. It is our belief that, after introducing specific knowledge about the camera systems used today, the sampling space for g can be optimized while maintaining the performance needed for the accurate encoding of images in various motion picture workflows.

References

1. T. Mitsunaga and S. K. Nayar, "Radiometric Self Calibration," *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, p. 2, 1999.
2. P. E. Debevec and J. Malik, "Recovering High Dynamic Range Radiance Maps from Photographs," *Proc. 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 97*, New York, NY, pp. 369–378, 1997.
3. M. D. Grossberg and S. K. Nayar, "Determining the Camera Response From Images: What Is Knowable?" *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(11):1455–1467, Nov. 2003.
4. C. Manders, C. Aimone, and S. Mann, "Camera Response Function Recovery From Different Illuminations of Identical Subject Matter," *Proc. International Conference on Image Processing*, vol. 5, pp. 2965–2968, 2004.
5. S. J. Kim, J.-M. Frahm, and M. Pollefeys. "Joint Feature Tracking and Radiometric Calibration From Auto-Exposure Video," *Proc. IEEE 11th International Conference on Computer Vision*, pp. 1–8, Oct. 2007.
6. H. Farid. "Blind Inverse Gamma Correction," *IEEE Trans. Image Process.*, 10(10):1428–1433, Oct. 2001.
7. S. Lin, J. Gu, S. Yamazaki, and H.-Y. Shum. "Radiometric Calibration from a Single Image," *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. II–938–II–945, Jul. 2004.
8. M. Diaz and P. Sturm, "Radiometric Calibration Using Photo Collections," *Proc. IEEE International Conference on Computational Photography*, pp. 1–8, Apr. 2011.
9. K. Shafique and M. Shah, "Estimation of the Radiometric Response Function of a Color Camera from Differently Illuminated Images," *Proc. International Conference on Image Processing*, vol. 4, pp. 2339–2342 v. 4, Oct. 2004.
10. S. Kuthirummal, A. Agarwala, D. B Goldman, and S. K Nayar, "Priors for Large Photo Collections and What They Reveal About Cameras," *Proc. European Conference on Computer Vision Computer Vision*, pp. 74–87, 2008.
11. M. D. Grossberg and S. K. Nayar, "Modeling the Space of Camera Response Functions," *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(10):1272–1282, Oct. 2004.
12. R. Li, P. Shi, J. Pelz, C. O. Alm, and A. R. Haake, "Modeling Eye Movement Patterns to Characterize Perceptual Skill in Image-Based Diagnostic Reasoning Processes," *Comput. Vis. Image Understand.*, 151:138–152, 2016.
13. Andrieu, N. D. Freitas, A. Doucet, and M. I. Jordan, "An Introduction to MCMC for Machine Learning," *Mach. Learn.*, 50(1–2):5–43, 2003.
14. C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag: Berlin, Heidelberg, 2006.
15. R. R. Figueroa and J. Gu, "Camera Radiometric Calibration From Motion Images," presented at the SMPTE 2013 Annual Technical Conference and Exhibition, Hollywood, CA, Oct. 2013.

About the Authors

Ricardo R. Figueroa is currently an associate professor and program director for the Motion Picture Science Program at the Rochester Institute of Technology (RIT), Rochester, NY. Figueroa joined RIT after working at Eastman Kodak, Rochester, NY, for ten years, where he held positions as the film and digital lab manager, digital/hybrid technologies regional director, digital imaging research engineer, and Kodak Operating System Manager. Figueroa holds BSEE and MSEE degrees from the University of Puerto Rico at Mayagüez, Mayagüez, Puerto Rico, and is currently pursuing a PhD degree in computing and information sciences at RIT. His research interests are in the areas of machine learning and deep learning for image processing applications. Outside of work, Figueroa is an avid triathlete, soccer player, and golfer and is also active in various professional and civic organizations.

Rui Li is an assistant professor at the Golisano College of Computing and Information Sciences, Rochester Institute of Technology (RIT), Rochester, NY. He received a BS degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2003. He earned a PhD degree in computing and information sciences from RIT in 2013. His research focus is on machine learning, artificial intelligence, and their applications.

Presented at the SMPTE 2018 Annual Technical Conference & Exhibition, Los Angeles, CA, 22–25 October 2018.