

Probability Models for High Dynamic Range Imaging

Chris Pal

Rick Szeliski

Matthew Uyttendaele

Nebojsa Jojic

Microsoft Research, Redmond, WA 98052
pal@psi.utoronto.ca, {szeliski, mattu, jojic}@microsoft.com

Abstract

Methods for expanding the dynamic range of digital photographs by combining images taken at different exposures have recently received a lot of attention. Current techniques assume that the photometric transfer function of a given camera is the same (modulo an overall exposure change) for all the input images. Unfortunately, this is rarely the case with today's camera, which may perform complex non-linear color and intensity transforms on each picture. In this paper, we show how the use of probability models for the imaging system and weak prior models for the response functions enable us to estimate a different function for each image using only pixel intensity values. Our approach also allows us to characterize the uncertainty inherent in each pixel measurement. We can therefore produce statistically optimal estimates for the hidden variables in our model representing scene irradiance. We present results using this method to statistically characterize camera imaging functions and construct high-quality high dynamic range (HDR) images using only image pixel information.

1 Introduction

The ability to map image pixels in different images to some estimate of scene irradiance is of fundamental importance to many computer vision, digital photography, and image processing applications. In this paper, we are particularly concerned with applications involving the imaging of high dynamic range (HDR) scenes using multiple images that are well aligned. Figure 1 illustrates a sequence of images of an HDR scene that were taken through an exterior window of a relatively dark interior room. We are interested in combining such images into a single image of the scene.

Both Charge-Coupled Device (CCD) and photochemical based cameras often possess non-linear relationships between the amount of light hitting the image surface for a given amount of time and the resulting image intensity. In both digital and chemical photography, the aperture size (or f-stop) and shutter speed (or exposure time) have a large impact on the final mapping relating image intensities to

irradiance values. Some intermediate quantities can be defined based on these settings. The term *exposure value* is commonly used to refer to the product of the aperture and shutter speed while the *exposure* refers to the product of the irradiance and the exposure time. For electronic imaging devices, a number of other settings have a bearing on the irradiance-pixel value relationship.

Many electronic cameras have white balance settings that can have a dramatic impact on the irradiance-pixel value relationship. ISO settings may also be present, affecting the sensitivity of the CCD, potentially acting like an electronic gain or bias. Additionally, color saturation settings and general digital signal processing can further complicate the underlying imaging process.

Typically, for the task of constructing an image of an HDR scene from different images, a camera is set to manual mode and the exposure time is altered. However, many cameras do not have a truly, fully manual mode. We are also interested in applications involving the construction of panoramas from video involving HDR scenes. In this paper we do not deal with the obvious alignment issues in such applications. However, in this paper we do deal with the case of estimating imaging functions when the values of camera settings are not available, as is commonly the case with images from video cameras. We achieve this by using Gaussian process priors for our imaging functions.

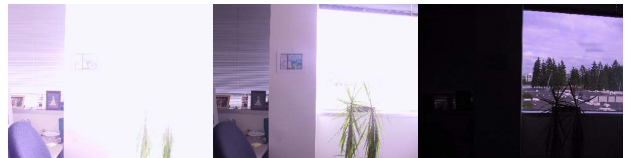


Figure 1: Three images of an HDR scene.

1.1 Previous Work

In the past, a number of models have been proposed taking into account various aspects of the potential relationship between irradiance and pixel values in an image. This is partly due to the fact that imaging functions for film and

CCD imaging devices can have a wide variety of possible functional forms. Indeed, Grossman and Nayar [3] describe a database they have collected consisting of over 200 different response functions.

In [6], Mann presents a table in which he enumerates an extremely wide variety of parametric forms for camera imaging functions. Some of these are variations of simple gamma functions, which have also been used as photographic film models [7], with the form

$$f(r) = \alpha + \beta r^\gamma, \quad (1)$$

where r is the irradiance on the imaging surface and α, β and γ are model parameters. However, Mann lists various other camera imaging function models, including the model

$$f(r) = \left(\frac{e^b r^a}{e^b r^a + 1} \right)^c, \quad (2)$$

where a, b, c and e are the parameters of the model. Equation (1) represents a variation on an $r/(r+1)$ based model which has also been used in [10] for high dynamic range tone reproduction. Of the 13 different functions that Mann enumerates, equations (2) and (1) are cited as being the models most commonly used by the author.

Debevec and Malik's work [1] on recovering high dynamic range images from photographs uses a model that assumes the relationship between irradiance and a pixels value has the form

$$f(r) = f(ar), \quad (3)$$

Importantly, this model assumes a simple, *known* pre-nonlinearity multiplicative gain set equal to the exposure time. Debevec and Malik estimate $h = \ln f^{-1}$, the log of the inverse imaging function

$$h(x_{k,i}) = \ln(r_i) + \ln(a_k). \quad (4)$$

They use a least squares approach to estimate the log inverse function and introduce a smoothness, regularizing term for the function h . Their objective function is written

$$\mathcal{F} = \sum_{i=1}^N \sum_{k=1}^P (h(x_{k,i}) - \ln(r_i) - \ln(a_k))^2 + \lambda \sum_{x=x_{min}+1}^{x_{max}-1} h''(x)^2, \quad (5)$$

where N is the number of spatial locations and P is the number of images.

When irradiance values are known or if one wishes to compute the *ratio* of two imaging functions, Mitsunaga and Nayar [8] have advocated the use of high-order polynomial models for $g = f^{-1}$, the inverse imaging function

$$r = g(x) = \sum_{n=0}^N c_n x^n, \quad (6)$$

where, x is the pixel intensity, N is the order of the polynomial and c_n are the polynomial coefficients. In this work, the calibration procedure must determine the order of the polynomial and the coefficients c_n .

In the work of Tsin, Ramesh and Kanade [11] a model specifically aimed at the CCD imaging process is presented. Here, white balancing is explicitly modeled as a pre-nonlinearity scaling a and offset b independently applied within color channels of the image. The following imaging function is proposed

$$f(r) = f(ar + b), \quad (7)$$

where $a = a_o t$, if one wishes to explicitly model the exposure time, t . They explicitly model pre-nonlinearity *shot noise* and *thermal noise* as additive terms and also include a post-nonlinearity noise term. The inverse function g is estimated using a Taylor series approximation. They also use a smoothness term for g , adding a term to their cost function consisting of the summation of a finite difference approximation to the second derivative of g .

Importantly, the analysis of Grossberg and Nayar's in [2] illustrates the impossibility of simultaneously recovering the response function of the imaging device and exposure ratios (or pre-nonlinearity multiplicative factors) without making assumptions on the form of the response function. As such, we use clear and explicit but weak prior models to simultaneously estimate response functions and irradiances.

In contrast to all these methods, we construct a formal generative model for the high dynamic range imaging problem. This formalism allows us to make our assumptions about the form of imaging functions explicit. This formalism also allows us to construct a well defined objective function. Using a variational formulation of our optimization approach, we can guarantee that we optimize the log probability of our data. We use graphical models to illustrate the structure of our model. In the following sections, we show how generative models and Gaussian processes allow us to explicitly specify well defined priors on the derivative structure of imaging functions, allowing us to estimate models consisting of a different imaging function for each image.

2 A Generative Model for the HDR Imaging Problem

We now outline a generative probability model for our imaging system. We apply the following model and algorithm to the pixel intensity values in each color channel of an image independently. For a given spatial location i in our image, we generate an irradiance value r from the distribution $p(r)$. For each of our k images, $k \in \{1, \dots, K\}$, we

generate a pixel intensity from

$$p(x_k|r) = \mathcal{N}(x_k; f_k(r), v_k(r)), \quad (8)$$

where $f_k(r)$ is our imaging function and $v_k(r)$ is the level-dependent variance of the additive noise in the sensor. (See Figures 5-6 for examples of such functions and their variances, which are shown as the vertical bands around the response function.) We can think of this conditional distribution as a probabilistic lookup table that maps irradiance values to a Gaussian distribution over pixel values in each image. The joint distribution for the irradiance and image pixels for spatial location i is thus given by

$$p(x_1, x_2, \dots, x_P, r) = \prod_{k=1}^K \mathcal{N}(x_k; f_k(r), v_k(r)) p(r). \quad (9)$$

In our model, we assume pixels at different spatial locations $i \in \{1, \dots, N\}$ are independent. We use uniform distributions for $p(r_i)$. The joint distribution of our hidden variables (all the irradiance values), and our observed variables (all the pixels in all the images) is then given by

$$p(x_{1\dots K, 1\dots N}, r_{1\dots K}) = \prod_{i=1}^N \prod_{k=1}^K \mathcal{N}(x_{k,i}; f_k(r_i), v_k(r_i)) \prod_{i=1}^N p(r_i). \quad (10)$$

With the addition of priors for our functions, the joint distribution of our observed pixels, hidden irradiances, functions $f_k = f_k(r)$ and variances $v_k = v_k(r)$ can be written as

$$p(x_{1\dots K, 1\dots N}, r_{1\dots K}, f_{1\dots K}, v_{1\dots K}) = \prod_{i=1}^N \prod_{k=1}^K \mathcal{N}(x_{k,i}; f_k(r_i), v_k(r_i)) \prod_{i=1}^N p(r_i) \prod_{k=1}^K p(f_k) \prod_{k=1}^K p(v_k). \quad (11)$$

We illustrate our complete graphical model using the Bayesian Network in Figure 2. Here, observed pixel values are illustrated as grey nodes in the graph, unobserved irradiances are illustrated as a horizontal row of uncolored nodes across the top of the graph. The functions themselves are treated as random variables and are illustrated as a vertical row of uncolored nodes on the left of the graph. If our functions were known, we would treat the function variables in the graph as being observed. This would eliminate the problem of conditional dependencies between unknown functions and irradiances. We could thus also relatively easily perform probabilistic inference for posteriors for r_i . However, we wish to learn or estimate the functions and the irradiances from our image pixel data.

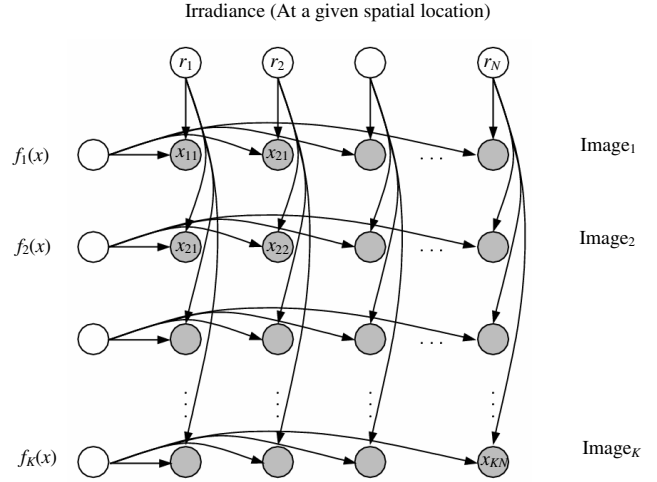


Figure 2: A probability model for image pixel values, irradiances and imaging functions.

2.1 Priors on Functions

To optimize the probability model in Figure 2, we need some additional “hard” identifiability constraints or “soft” priors. We wish to encode a prior on the smoothness of $f_k(r)$ and $v_k(r)$, we also wish to encode priors concerning the scaling and any stretching of the irradiance space. We may also wish to enforce the monotonicity of our functions¹. A close connection exists between the Bayesian estimation of stochastic processes and smoothing by splines [12, 5, 4]. Here, we use Gaussian process priors [9, 12] to encode the correlations between different function values.

We first develop a probabilistic model for the imaging function $f(r)$, and then later show how a slightly modified version of this model can also be used to model the variance $v(r)$. An instantiation of the random variable $f(r)$ can be thought of as a lookup table. For clarity, we write $f(r)$ as \mathbf{f} , a random vector for the discretized version of the function.

We can generate the $(i-1)$ th derivative of a function by integrating the i th derivative of a function using an integrating matrix

$$\mathbf{A}_0 = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}. \quad (12)$$

We can thus construct a probability model for a function in terms of the random variables for derivatives \mathbf{f}'' , \mathbf{f}' and

¹In practice, we have found that using our pixel sampling scheme described later, the additional complexity of the monotonicity constraint is not warranted.

scalars b_1, b_0 as

$$\begin{aligned}
& p(f''(r), f'(r), f(r), b_1, b_0) \\
&= p(f(r)|f'(r), b_0)p(f'(r)|f''(r), b_1) \\
&\quad p(f''(r))p(b_1)p(b_0) \\
&= \mathcal{N}(\mathbf{f}; \mathbf{A}_0 \mathbf{f}' + b_0, \Sigma_f) \mathcal{N}(\mathbf{f}'; \mathbf{A}_0 \mathbf{f}'' + b_1, \Sigma_{f'}) \\
&\quad \mathcal{N}(\mathbf{f}''; \boldsymbol{\mu}_{f''}, \Sigma_{f''}) \mathcal{N}(b_1; \mu_{b_1}, \sigma_{b_1}^2) \mathcal{N}(b_0; \mu_{b_0}, \sigma_{b_0}^2).
\end{aligned} \tag{13}$$

Equation (13) represents a general form for a complete, generative model for functions, \mathbf{f} . The Bayesian Network in the far left of Figure 3 illustrates the model we have constructed so far. We now simplify this model so that we can express our prior on functions as compactly as possible.

In a properly defined generative model for smooth functions, \mathbf{f}' is deterministically related to \mathbf{f}'' and b_1 while \mathbf{f} is deterministically related to \mathbf{f}' and b_0 . As such, we have freedom on how to set $\Sigma_{f''}$, $\sigma_{b_1}^2$ and $\sigma_{b_0}^2$. In practice, we set our covariance model for \mathbf{f}'' to either $\Sigma_{f''} = \sigma_{f''}^2 \mathbf{I}$ (an isotropic Gaussian) or to a general diagonal covariance matrix. We typically use fairly broad priors or large values for our σ_b^2 s. The middle graph in Figure 3 illustrates this model where we have integrated out the random variable for \mathbf{f}' .

To further simplify our generative model for functions, we define the random variable $\mathbf{z} = [\mathbf{f}''^T \quad b_1 \quad b_0]^T$ and define $\boldsymbol{\mu}_z = [\boldsymbol{\mu}_{f''}^T \quad \mu_{b_1} \quad \mu_{b_0}]^T$. If we then define $\mathbf{A} = \mathbf{A}_1 \mathbf{A}_2$, where $\mathbf{A}_1 = [\mathbf{A}_0 \quad \boldsymbol{\nu}]$, $\boldsymbol{\nu} = [1, 1, \dots, 1]^T$, $\mathbf{A}_2 = \begin{bmatrix} \mathbf{A}_1 & 0 \\ 0 & 1 \end{bmatrix}$, we can write our generative model for functions as

$$p(f(r), \mathbf{z}) = \mathcal{N}(\mathbf{f}; \mathbf{A}\mathbf{z}, \Sigma_f) \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_z, \boldsymbol{\Psi}), \tag{14}$$

where

$$\boldsymbol{\Psi} = \begin{bmatrix} \Sigma_{f''} & 0 & 0 \\ 0 & \sigma_{b_1}^2 & 0 \\ 0 & 0 & \sigma_{b_0}^2 \end{bmatrix}.$$

This model is illustrated in the graph in the far right of Figure 3. The marginal distribution of \mathbf{f} can then be written

$$p(f(r)) = \mathcal{N}(\mathbf{f}; \mathbf{A}\boldsymbol{\mu}_z, \boldsymbol{\Phi}), \tag{15}$$

where $\boldsymbol{\Phi} = \mathbf{A}\boldsymbol{\Psi}\mathbf{A}^T$. If we set $\boldsymbol{\mu}_{f''} = [0, 0, \dots, 0]^T$, $\mu_{b_0} = 0$, $\mu_{b_1} = 0$, then after integrating over our hidden variables we arrive at an even simpler form for our prior for $f(r)$,

$$p(f(r)) = \mathcal{N}(\mathbf{f}; 0, \boldsymbol{\Phi}). \tag{16}$$

There is a close relationship between equation (16) and smoothness regularization methods. (See Appendix A.1 for further details.) In modeling our response functions $f_k(r)$ and variances $v_k(r)$, for reasons of computational efficiency, one can use the simpler form (16). However for the highest gain image, $f_1(r)$, we use the mode of equation (15).

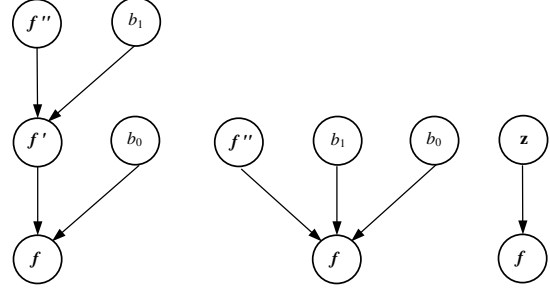


Figure 3: Illustrating a generative model for functions.

3 Optimizing the Model

We have now defined a formal probability model for our problem. To optimize our model, we wish to maximize the log of the marginal probability of all our observed data under our model. For clarity we use the more compact notation for denoting sets of continuous random variables $\{x_{k,i}\} = \{x_{1\dots K, 1\dots N}\}$. Thus, we wish to maximize

$$\begin{aligned}
& \log P(\{x_{k,i}\}) = \\
& \log \int_{\{r_i\}} \int_{\{f_k\}} \int_{\{v_k\}} p(\{x_{k,i}\}, \{r_i\}, \{f_k\}, \{v_k\}).
\end{aligned} \tag{17}$$

To deal with the intractability of marginalizing over the hidden variables, we construct a procedure equivalent to iteratively updating *maximum a posteriori* (MAP) estimates for our functions and then for our irradiance estimates. Viewed as a variational bound, this procedure is equivalent to minimizing

$$\begin{aligned}
& -E_Q[\log p(\{x_{k,i}\}, \{r_i\}, \{f_k\}, \{v_k\})] \\
&= -\left[\sum_{k=1}^K \sum_{i=1}^N \log \mathcal{N}(x_{k,i}; f_k(r_i), v_k(r_i)) + \right. \\
&\quad \left. \sum_{i=1}^N \log p(r_i) + \sum_{k=1}^K \log p(f_k) + \sum_{k=1}^K \log p(v_k) \right],
\end{aligned} \tag{18}$$

corresponding to the expectation under our approximating distribution, Q of the log probability of our data and hidden variables. In equation (18), Q consists of Dirac delta functions and the $\{r_i\}, \{f_k\}, \{v_k\}$ terms in equation (18) are estimates of the locations of these delta functions or equivalently MAP estimates of the corresponding random variables. This procedure guarantees that we optimize a bound on the log of the marginal probability of the data. (See Appendix A.2 for details.)

To simplify our presentation, we write our data $x_{k,i}$ as vectors \mathbf{x}_k . We write our estimates for our function as the vector \mathbf{f}_k or $\mathbf{f}_k(\boldsymbol{\tau})$, where $\boldsymbol{\tau} = [0, 1, \dots, \max(\mathbf{r})]$, $\boldsymbol{\tau} \in \mathbb{I}^d$. Similarly, we define \mathbf{v}_k as the estimate of our variance as a function of $\boldsymbol{\tau}$. We define r_i as the elements of vector \mathbf{r} and

the result of each function $f_k(r_i)$ applied to each r_i of \mathbf{r} as a vector $\mathbf{f}_k(\mathbf{r})$. We can then write $\mathbf{f}_k(\mathbf{r})$ as $\mathbf{\Lambda}_k \mathbf{f}_k$, where $\mathbf{\Lambda}_k$ is a sparse matrix with a single one located in each row with positions defined by r_i . If we then define $\mathbf{\Sigma}_k$ as a diagonal matrix consisting of $v_k(r_i)$, we can write the first term of equation (18) as

$$\sum_{k=1}^K \sum_{i=1}^N \log \mathcal{N}(x_{k,i}; f_k(r_i), v_k(r_i)) = \sum_{k=1}^K \left(\frac{1}{2} \log |\mathbf{\Sigma}_k| + (\mathbf{x}_k - \mathbf{\Lambda}_k \mathbf{f}_k)^T \mathbf{\Sigma}_k^{-1} (\mathbf{x}_k - \mathbf{\Lambda}_k \mathbf{f}_k) \right). \quad (19)$$

3.1 Updating the Variational Parameters

Our algorithm for updating the variational parameters for our model consists of three simple equations. The procedure amounts to computing a MAP estimate by iteratively updating our estimates for $\{r_i\}$, $\{f_k\}$, and $\{v_k\}$. We compute our updates for each function \mathbf{f}_k using our prior from equation (16) except for one of our functions (see Section 3.3) where we use the prior from equation (15). Then, setting $\partial F / \partial \mathbf{f}_k = 0$, our updates can then be computed from

$$\mathbf{f}_k^{new} = \left[\mathbf{\Lambda}_k^T \mathbf{\Sigma}_k^{-1} \mathbf{\Lambda}_k + \mathbf{\Phi}^{-1} \right]^{-1} \left[\mathbf{\Lambda}_k^T \mathbf{\Sigma}_k^{-1} \mathbf{x}_k + \mathbf{\Phi}^{-1} \mathbf{A} \boldsymbol{\mu}_z \right]. \quad (20)$$

We compute updates for \mathbf{v}_k using our prior from equation (16). Here we set $\mathbf{\Sigma}_k = \mathbf{I}, \forall k$ and specify $\mathbf{\Lambda}_k$ as before. We first compute $y_{k,i} = (x_{k,i} - f_k(r_i))^2$. Then, for each function k we compute

$$\mathbf{v}_k^{new} = \left[\mathbf{\Lambda}_k^T \mathbf{\Sigma}_k^{-1} \mathbf{\Lambda}_k + \mathbf{\Phi}^{-1} \right]^{-1} \left[\mathbf{\Lambda}_k^T \mathbf{\Sigma}_k^{-1} \mathbf{y}_k \right]. \quad (21)$$

Updates of the irradiance estimates, r_i are given by the maximum a posteriori (MAP) values, computed

$$r_i^{new} = \arg \min_{\tau} \left(- \sum_{k=1}^K \left(\log \mathcal{N}(x_{k,i}; f_k(\tau), v_k(\tau)) \right) - \log(p(\tau)) \right), \quad (22)$$

which we compute using lookup tables over the range of x and τ computed for our functions. We use a uniform prior for $p(\tau)$ over the range zero to the maximum possible value of τ we wish to have.

3.2 Updating the Model Parameters

We wish to update parameters $\theta_k = \{\sigma_{f''}^2, \sigma_{b_0}^2, \sigma_{b_1}^2, \boldsymbol{\mu}_{f''}, \mu_{b_1}, \mu_{b_0}\}_k$ (or some subset of

these parameters). We thus optimize the third term in equation (18), which can be written as

$$\frac{\partial \log p(\mathbf{f}_k)}{\partial \theta_k} = \frac{\partial}{\partial \theta_k} \log \left(\int_{\mathbf{z}_k} p(\mathbf{f}_k, \mathbf{z}_k) \right) d\mathbf{z}_k = E \left[\frac{\partial}{\partial \theta_k} \log p(\mathbf{f}_k, \mathbf{z}_k) \Big| \mathbf{f}_k \right], \quad (23)$$

where this expectation is based on the posterior distribution,

$$p(\mathbf{z}_k | \mathbf{f}_k) = \mathcal{N}(\mathbf{z}_k; \boldsymbol{\mu}_{z_k} + \beta(\mathbf{f}_k - \mathbf{A} \boldsymbol{\mu}_{z_k}), \mathbf{\Psi} - \beta(\mathbf{A} \mathbf{\Psi})), \quad (24)$$

where

$$\beta = (\mathbf{A} \mathbf{\Psi})^T (\mathbf{A} \mathbf{\Psi} \mathbf{A}^T)^{-1}. \quad (25)$$

We thus perform an E-step for our update where we compute the expectations

$$E[\mathbf{z}_k | \mathbf{f}_k] = \boldsymbol{\mu}_{z_k} + \beta(\mathbf{f}_k - \mathbf{A} \boldsymbol{\mu}_{z_k}), \quad (26)$$

$$E[\mathbf{z}_k \mathbf{z}_k^T | \mathbf{f}_k] = \mathbf{\Psi} - \beta \mathbf{A} \mathbf{\Psi} + E[\mathbf{z}_k | \mathbf{f}_k] E[\mathbf{z}_k | \mathbf{f}_k]^T. \quad (27)$$

We then perform an M-step where we maximize the expected log likelihood from equation (23). The resulting updates are

$$\boldsymbol{\mu}_{z_k}^{new} = E[\mathbf{z}_k | \mathbf{f}_k] \quad (28)$$

and for each function k we update our variances using $\sigma_{b_1}^2 = \mathbf{\Psi}_{n-1, n-1}, \sigma_{b_0}^2 = \mathbf{\Psi}_{n, n}$. Our covariance model is updated using either $\sigma_{f''}^2 = \frac{1}{n-2} \sum_{i=1}^{n-2} \mathbf{\Psi}_{i, i}$, for a $\sigma_{f''}^2 \mathbf{I}$ model or $\mathbf{\Sigma}_{f''}^{i, i} = \mathbf{\Psi}_{i, i}, i = 1 \dots (n-2)$, for a diagonal covariance model. In either case,

$$\mathbf{\Psi} = \text{diag} \{ E[\mathbf{z}_k \mathbf{z}_k^T | \mathbf{f}_k]^T - 2 \boldsymbol{\mu}_{z_k} E[\mathbf{z}_k | \mathbf{f}_k]^T + \boldsymbol{\mu}_{z_k} \boldsymbol{\mu}_{z_k}^T \}. \quad (29)$$

3.3 Initialization and Update Steps

Even small misalignments of the underlying images can cause problems for any procedure estimating imaging functions. In fact, without a tripod and a remote shutter switch, we have found that significant noise can be introduced into certain pixels. As such, we only consider the subset of pixels that are least likely to be affected by such misalignment of the images, i.e., pixels that are not near intensity edges in a given color channel. Thus, for each image, we sort the image pixels in each color channel based on their edge magnitude. We then label the pixels with the lowest edge magnitude at each intensity level from each image. We record their spatial locations and gather pixels from the other images in the same spatial locations. We then use this subset of pixels in our estimation procedure for a given color channel.

We initialize $v_k(r) = c, \forall k, \forall r$, where c is some constant (typically $c \in [1, 10]$). We sort the images based on their mean values so $k = 1$ is the highest gain image and $k = K$ is the lowest gain image. For $k = 1$, we initialize a

subset of our irradiance estimates so that $\{R\} = \{r_i = x_{k,i}\}, \forall i: x_{k,i} \neq x_{max}$. As such, we implicitly initialize $f_1(r) = x, r < x_{max}$. Therefore, we use the prior from equation (15), i.e., the prior with a non-zero b_1 .

For $k \neq 1, \forall r \in \{R\}$ we use equation (20) to compute $f_{k+1}(r)$ with the prior of equation (16). In practice we initialize a $\Sigma_{f''} = \sigma_{f''}^2 \mathbf{I}$ model, where $\sigma_{f''}^2 = \epsilon$ with $\epsilon \ll 1$, typically $\epsilon = 10^{-6}$ and $\sigma_{b_0}^2 \approx 10^2$ for all functions. However, when we update our model parameters, we typically allow a diagonal $\Sigma_{f''}$. We set $\sigma_{b_1}^2 \approx \sigma_{f''}^2$ for the high gain function but $\sigma_{b_1}^2 \approx \sigma_{b_0}^2$ for the other functions. Interestingly, as $\sigma_{f''}^2 \rightarrow 0$ and $\sigma_{b_1}^2, \sigma_{b_0}^2 \rightarrow \infty$, this initialization procedure becomes equivalent to finding a linear $f(r) = ar + b$ model to map irradiance values from each image to the irradiance values in the highest gain image.

After our initialization procedure, we cycle through updating the variational parameters as described in section 3.1. This is then followed by an update of the model parameters. We then repeat our process until our procedure converges.

4 Results

Figure 4 shows three images of a high dynamic range scene followed by the composite HDR image obtained using our algorithm to compute irradiance estimates. To display the HDR image, we used the simple global version of the luminance re-mapping function in [10]. Figure 5 illustrates the evolution of our imaging function estimates over successive iterations of the algorithm. Each plot illustrates the set of three function, one for each image, along with the pixel intensities and irradiance estimates produced using our algorithm for the green color channels of the images. For this problem, the algorithm took 6 iterations to converge to the final solution.

To investigate the convergence of our algorithm with a larger number of images, we estimated curves for five images. Three of the images we used are shown in Figure 1 in our introduction. Figure 6 shows the results of this estimation procedure for the green color channel. For this data set, our algorithm converged after 25 iterations. The original pixel intensities lie in the range $[0, 255]$. We used a uniform prior in the range $[0, 1000]$ and our algorithm converged to an estimated range of $[0, 675]$. Figure 7 illustrates the resulting HDR image after luminance adjustment as described before. To further illustrate the utility of our algorithm, in Figure 8 we have re-mapped the pixels in the lowest gain image through the learned imaging function to an irradiance estimate and then generated a pixel intensity using the highest gain images mapping to pixel space. We compare this procedure to the best we could have done if



Figure 4: Three images of a high dynamic range scene. (Upper Left) Highest gain image. (Upper Right) Middle Gain image. (Lower Left) Low Gain image. (Lower Right) HDR image with luminances remapped into the viewable range.

we had estimated a purely multiplicative transformation or a linear model mapping pixels in the low gain image to the high gain image. There is a surprising amount of information contained in the low gain image (albeit noisy) with respect to dark pixel intensities.

5 Conclusions and Future work

We have presented a complete probabilistic model for the HDR imaging problem in which we are able to produce results where we have estimated imaging functions using only pixel intensity measurements. To increase the speed of our estimation procedure, one could also gather pixel measurements into intensity classes. However, in our present method, when computing our final HDR image we are presently able to detect “outlier” pixels arising from scene motion by looking for values probabilistically “incompatible” with the other images.

It is important to note that without any absolute ground truth information incorporated into the model, there may be an additional, unknown smooth warping of the estimated irradiance space. However, if ground truth information for

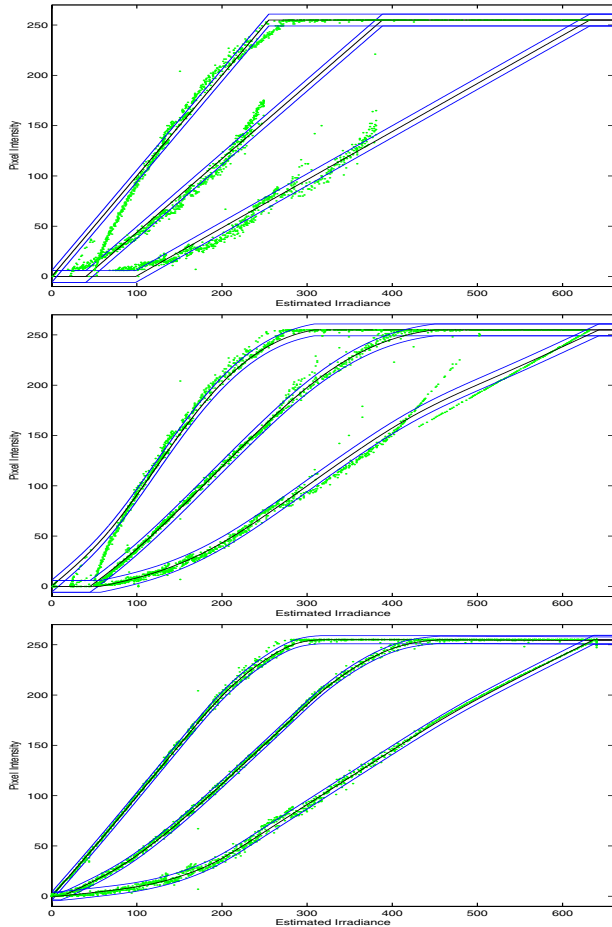


Figure 5: Iteration 0,1, and 6 (the final iteration) of the algorithm.

irradiance is available, it is easy to adapt our algorithm by assigning certain irradiance values to their known quantities. Thus, in some situations it may be desirable to perform color channel adjustments on the final HDR image. As future work, it is possible to incorporate chromaticity priors that could couple the estimation of curves across the color channels. Such priors could improve our estimation procedure by making the chromaticity of the final HDR irradiance estimates as close as possible to the chromaticity of unsaturated or low uncertainty pixels across the different images. It would also be useful to investigate how information concerning camera setting information would be formulated in the context of our prior structures so as to benefit the estimated imaging functions.

References

[1] P. E. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. *Computer Graphics*,

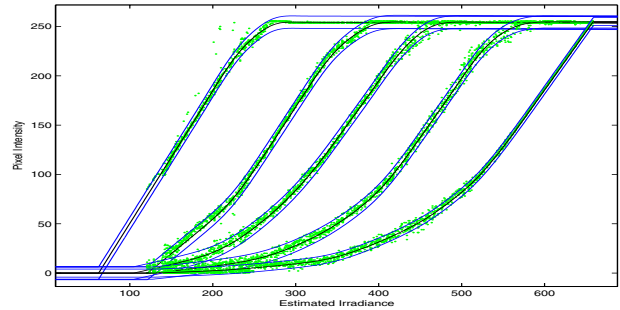


Figure 6: Five imaging functions with pixel intensity values and irradiance estimates.



Figure 7: Left to right. The middle gain image, the 4th lowest gain image and the HDR composite image

31(Annual Conference Series, SIGGRAPH 1997):369–378, 1997.

- [2] M. D. Grossberg and S.K. Nayar. What can be known about the radiometric response from images? *Proceedings of the European Conference on Computer Vision (ECCV)*, 4:189–205, 2002.
- [3] M. D. Grossberg and S.K. Nayar. What is the space of camera response functions? *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2:602–609, 2003.
- [4] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer-Verlag, New York, 2001.
- [5] G.S. Kimeldorf and G. Wahba. A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41(2):495–502, 1970.
- [6] S. Mann. Comparametric equations with practical applications in quantigraphic image processing. *IEEE Trans. Image Proc.*, 9(8):1389–1406, August 2000.
- [7] S. Mann and R. Picard. Being ‘undigital’ with digital cameras: Extending dynamic range by combining differently exposed pictures. In *Proceedings of IST 46th annual conference*, pages 422–428, May 1995.
- [8] T. Mitsunaga and S. K. Nayar. Radiometric self calibration. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (ICCV)*, June 1999.
- [9] R. Neal. *Bayesian Learning for Neural Networks*. Springer-Verlag, New York, 1996.



Figure 8: Mapping the lowest gain image to the highest gain image. (Upper Left) The highest gain image. (Upper Right) Estimating a model with only a multiplicative gain (Lower Left) A multiplicative factor and a bias. (Lower Right) Our Algorithm

- [10] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda. Photographic tone reproduction for digital images. *Proceedings of ACM SIGGRAPH*, pages 267–276, 2002.
- [11] Y. Tsin, V. Ramesh, and T. Kanade. Statistical calibration of CCD imaging process. *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, July 2001.
- [12] G. Wahba. *Spline Models for Observational Data*. volume 59 of CBMS-NSF regional conference series in applied mathematics. SIAM, 1990.

A Appendix

A.1 From Gaussian Processes to Smoothness Regularization

If we then specify broad priors for b_1 and b_0 by setting $\sigma_{b_1}^2, \sigma_{b_0}^2 \gg \sigma_{f''}^2$, Φ from equation (16) has the property that

$$\begin{aligned} \Phi^{-1} &= (\mathbf{A}\Psi\mathbf{A}^T)^{-1} \\ &\approx \frac{1}{\sigma_{f''}^2} \begin{bmatrix} 1 & -2 & 1 & 0 & 0 \\ -2 & 5 & -4 & 1 & 0 \\ & & \cdots & & \\ 1 & -4 & \mathbf{6} & -4 & 1 \\ & & \cdots & & \\ 0 & 1 & -4 & 5 & -2 \\ 0 & 0 & 1 & -2 & 1 \end{bmatrix} = \frac{1}{\sigma_{f''}^2} \mathbf{D}_2^T \mathbf{D}_2, \end{aligned} \quad (30)$$

where the matrix \mathbf{D}_2 is the finite difference operator used in least squares style regularized estimation, e.g.

$$\mathbf{D}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (31)$$

Since the corresponding penalty term we wish to minimize in smoothness regularized least squares approaches can be written as: $(\mathbf{D}_2 \mathbf{x})^T (\mathbf{D}_2 \mathbf{x}) = \mathbf{x}^T \mathbf{D}_2^T \mathbf{D}_2 \mathbf{x}$, we see that, $\mathbf{D}_2^T \mathbf{D}_2$ is an analogous quantity to Φ_f^{-1} in equation (30).

A.2 Bounding the Probability of the Data

We can bound the log marginal probability of our data by the free energy F by observing that the marginal probability of the data can be written

$$\begin{aligned} \log p(\{x_{k,i}\}) &= \log \int_{\{r_i, f_k, v_k\}} p(\{x_{k,i}, \{r_i\}, \{f_k\}, \{v_k\}) = \\ &\log \int_{\{r_i, f_k, v_k\}} Q(\{r_i\}, \{f_k\}, \{v_k\}) \frac{p(\{x_{k,i}, \{r_i\}, \{f_k\}, \{v_k\})}{Q(\{r_i\}, \{f_k\}, \{v_k\})} \\ &\geq \int_{\{r_i, f_k, v_k\}} Q(\{r_i\}, \{f_k\}, \{v_k\}) \log \left(\frac{p(\{x_{k,i}, \{r_i\}, \{f_k\}, \{v_k\})}{Q(\{r_i\}, \{f_k\}, \{v_k\})} \right), \end{aligned} \quad (32)$$

this bound is known as a free energy, F and can be expressed as

$$F = -H(Q) - E_Q[\log p(\{x_{k,i}, \{r_i\}, \{f_k\}, \{v_k\})]. \quad (33)$$

We can thus increase our bound on the log probability of the data by minimizing our free energy F . To do this tractably, we define our $Q(\{r_i\}, \{f_k\}, \{v_k\}) = q(\{r_i\})q(\{f_k\})q(\{v_k\})$ where our q functions consist of products of Dirac delta's. With this mechanism in place, we can now derive equation (18) as

$$\begin{aligned} &- E_Q \{ \log P(\{x_{k,i}, \{r_i\}, \{f_k\}, \{v_k\}) \} = \\ &- \int_{\{r_i, f_k, v_k\}} \prod_{i=1}^N \delta(r_i - \tilde{r}_i) \prod_{k=1}^K \delta(f_k - \tilde{f}_k) \prod_{k=1}^K \delta(v_k - \tilde{v}_k) \\ &\quad \left[\sum_{k=1}^K \sum_{i=1}^N \log \mathcal{N}(x_{k,i}; f_k(r_i), v_k(r_i)) + \sum_{i=1}^N \log p(r_i) \right. \\ &\quad \left. + \sum_{k=1}^K \log p(f_k) + \sum_{k=1}^K \log p(v_k) \right] \\ &= - \left[\sum_{k=1}^K \sum_{i=1}^N \log \mathcal{N}(x_{k,i}; \tilde{f}_k(\tilde{r}_i), \tilde{v}_k(\tilde{r}_i)) \right. \\ &\quad \left. + \sum_{i=1}^N \log p(\tilde{r}_i) + \sum_{k=1}^K \log p(\tilde{f}_k) + \sum_{k=1}^K \log p(\tilde{v}_k) \right] \end{aligned} \quad (34)$$