From Pixels to Physics: Probabilistic Color De-rendering

Ying Xiong
Harvard University
yxiong@seas.harvard.edu

Kate Saenko
Harvard University
saenko@seas.harvard.edu

Trevor Darrell
UC Berkeley
trevor@eecs.berkeley.edu

Todd Zickler
Harvard University
zickler@seas.harvard.edu

Abstract

Consumer digital cameras use tone-mapping to produce compact, narrow-gamut images that are nonetheless visually pleasing, however, in doing so, they discard or distort substantial radiometric signal that could otherwise be used for computer vision. Existing methods attempt to undo these effects through deterministic maps to de-render *the reported narrow-gamut colors back to their original wide-gamut sensor measurements. Unfortunately, deterministic approaches are unreliable, as this reverse narrow-to-wide mapping is one-to-many and has inherent uncertainty. Our solution is to use probabilistic maps, providing uncertainty estimates useful to many applications. We use a non-parametric Bayesian regression technique—local Gaussian process regression—to learn for each pixel's narrow-gamut color a probability distribution over the scene colors that could have created it. Using a variety of consumer cameras we show that these distributions, once learned from training data, are effective in simple probabilistic adaptations of two popular applications: multi-exposure imaging and photometric stereo. Our results on these applications are shown to be qualitively better than using the corresponding deterministic approaches, especially in saturated and out-of-gamut regions.*

1. Introduction

Most digital images produced by consumer cameras and shared online exist in narrow-gamut, low-dynamic range formats.[1] This is efficient for storage, transmission, and display, but it is unfortunate for computer vision systems that seek to interpret this data radiometrically when learning object appearance models for recognition, reconstructing scene models for virtual tourism, or performing other visual tasks with Internet images. Indeed, most computer vision algorithms are based, either implicitly or explicitly, on the assumption that image measurements are proportional to the spectral radiance of the scene (called *scene color* hereafter), and when a consumer camera renders its digital linear color measurements to a narrow-gamut output color space (called *rendered color* hereafter), this proportionality is almost always destroyed. Fig. 1 shows an example.

Existing approaches to color de-rendering attempt to undo the effects of a camera's color processing pipeline through "radiometric calibration" [6, 20, 21], in which rendered colors (i.e., those reported in a camera's JPEG output) are reverse-mapped to corresponding scene colors (i.e., those that would have been reported by the same camera's RAW output) using a learned deterministic function. This approach is unreliable, as it ignores the inherent uncertainty caused by the loss of information. A typical camera renders many distinct sensor measurements to the same small neighborhood of narrow-gamut output colors (see Fig. 2, right), and, once these output colors are quantized, the reverse mapping becomes one-to-many in some regions and cannot be deterministically undone.

How can we know which predictions are unreliable? As supported by Fig. 2, one expects the one-to-many effect to be greatest near the edges of the output gamut (i.e., near zero or 255 in an 8-bit JPEG file), and practitioners try to mitigate it using heuristics such as ignoring all JPEG pixels having values above or below certain thresholds in one or more of their channels. This trick improves the reliability of deterministic radiometric calibration, but it raises the question of how to choose thresholds for a given camera. ("Should I only discard pixels with values 0 or 255, or should I be more conservative?")[2] A more fundamental concern is that this heuristic works by discarding information that would otherwise be useful. Referring to Fig. 1,

[1] Typically sRGB color space with JPEG encoding: IEC 10918-1:1994 and IEC 61966-2-1:1999

[2] Our experiments in Fig. 4 and those of [18] reveal significant variation between models, suggesting the answer is the latter.

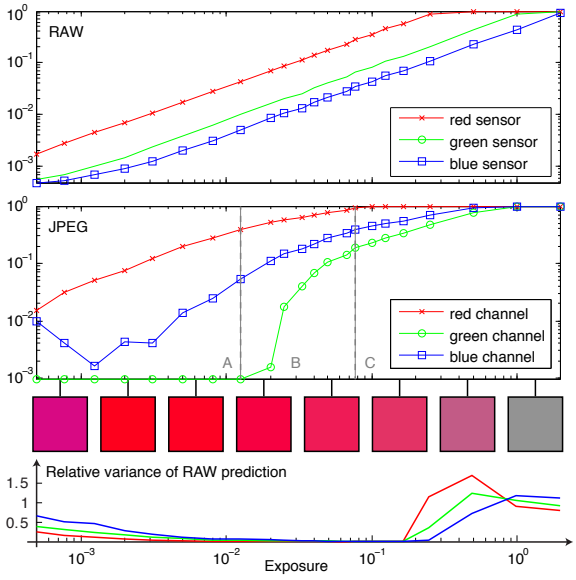

Figure 1. RAW and JPEG values for different exposures of the same spectral scene radiance collected by a consumer digital camera (DMC-LX3, Panasonic Inc.), along with normalized-RGB visualizations of the reported JPEG colors at a subset of exposures. Apart from saturation, RAW values are linear in exposure and proportional to spectral irradiance; but narrow-gamut JPEG values are severely distorted by tone-mapping. Given only JPEG values, what can we say about the unknown RAW values—and thus the scene color—that induced it? How can we use all of the JPEG color information, including when some JPEG channels are saturated (regions A and C)? The proposed method answers these questions by providing a confidence level (bottom plot) of the estimate, which can be incorporated into radiometry-based computer vision systems.

such a heuristic would ignore all JPEG measurements in regions A and C, even though these clearly tell us *something* about the latent scene color.

To overcome these limitations, this paper introduces a probabilistic approach for de-rendering. It presents a method for producing from each rendered (JPEG) color a probability distribution over the (wide gamut, high dynamic range) scene colors that could have induced it. The method relies on an offline calibration procedure involving registered RAW and JPEG image pairs, and from these it infers a statistical relationship between rendered colors and scene colors using local Gaussian process regression. This probabilistic approach provides a measure of confidence (e.g. the variance of the output distribution) for every de-rendered scene color, thereby eliminating the need for heuristic thresholds and making better use of the scene radiance information that is embedded in an Internet image. The offline calibration procedure is required only once for each different imaging mode of each camera, thus many per-camera de-rendering models could be stored in an online database and accessed on demand using camera model and mode information embedded in the metadata of an Internet image.[3]

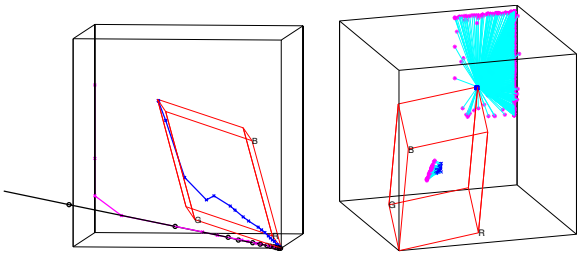

Figure 2. 3D visualization of color rendering in a consumer camera. *Left*: Same data as in Fig. 1. The black circles show the scene color $\mathbf{x}$. The corresponding RAW values $\tilde{\mathbf{x}}$ (magenta) are clipped due to saturation, and they are tone-mapped to create rendered colors $\mathbf{y}$ (blue) within the output sRGB gamut. *Right*: Rendered colors (blue) in small neighborhoods of $[127, 127, 127]$ and $[253, 253, 253]$ in an JPEG image are connected (through cyan lines) to the corresponding RAW measurements (magenta).

We evaluate our approach in a few different ways. First, we assess our ability to recover wide-gamut scene colors from JPEG observations in four different consumer cameras. Next, we employ our probabilistic de-rendering model in relatively straight-forward probabilistic adaptations of two established applications: high-dynamic range imaging with an exposure-stack of images (e.g. [31]) and three-dimensional reconstruction via Lambertian photometric stereo (e.g. [20]). In all cases, a probabilistic approach significantly improves our ability to infer radiometric scene structure from tone-mapped images.

1.1. Related work

There is a history of radiometric calibration for computer vision, the goal of which is to invert non-linear transformations of scene lightness and color that occur during imaging. The most common approach is to assume that the non-linearity can be described by a collection of three "radiometric response functions", which are monotonic deterministic functions that separately affect the measurements in each output color channel [20, 21, 6, 10]. The benefit of this approach is that it enables "self-calibration" through analysis of edge profiles [19] and image statistics [7, 16] or, assuming white balance is fixed or happens per-channel in the output color space [12], by making use of multiple illuminations or exposures [20, 21, 6, 9, 26, 28].

A significant limitation of the monotonic per-channel model is that it cannot recover out-of-gamut chromaticities. This can be explained using Fig. 2(left), which is a three-dimensional visualization of Fig. 1. In Fig. 2(left), the black axes depict the set of possible RAW measure-

[3] As has been done for lens distortion, e.g., PTlens (accessed Nov 21, 2011): http://www.epaperpress.com/ptlens/

ments, and the output sRGB gamut, shown as a red parallelogram, is the subset of this space to which all measurements must be tone-mapped.[4] When an out-of-gamut scene color $\mathbf{x} = (x^R, x^G, x^B)$ is rendered to a within-gamut output color $\mathbf{y} = (y^R, y^G, y^B)$, the traditional per-channel approach attempts to undo it by computing the estimate $\widehat{\mathbf{x}} = \{f^c(y^c)\}_{c=R,G,B}$ using positive-valued, monotonic functions $f^c(\cdot)$. This estimate cannot always be accurate because it is restricted to lie within the cone defined by the radial extension of the output sRGB gamut.

Chakrabarti et al. [3] show that more accurate deterministic models can be fit using an offline calibration procedure involving registered RAW and JPEG images. They consider multivariate polynomial models for the forward map from scene color $\mathbf{x}$ to output color $\mathbf{y}$, and while they find reasonable fits for most cameras, the residual errors remain quite high at 4-6 times most camera noise levels. Lin et al. [18] perform a thorough, larger-scale study and obtain significantly improved fits using radial basis functions, which are more flexible. Both approaches avoid the restrictions of per-channel response functions and can theoretically recover out-of-gamut chromaticities; but they remain deterministic, reporting a single color value instead of a distribution and not allowing for uncertainty prediction.

Pal et al. [23] proposed a probability model for traditional per-channel response functions, imposing weak priors such as smoothness and monotonicity. The model is learned from a sequence of aligned images with multiple exposures, and used to reconstruct an HDR image for that particular sequence. Unlike our approach, this model is not able to recover out-of-gamut chromaticities; furthermore, it does not produce a probability distribution over scene color values given a single rendered color vector and thus cannot make predictions on novel test images.

We employ a Bayesian non-parametric regression scheme to allow the data to determine the form of the mapping while providing an inherent representation of uncertainty. We adapt the method reported in [30], which learns a local Gaussian process for the neighborhood around a test point, in the spirit of locally-weighted regression [4].

2. Probabilistic de-rendering model

We begin with a model for the forward color processing pipeline of a typical consumer digital camera; then describe our representation for the reverse mapping. Both models ignore de-mosaicking, flare removal, noise removal, and sharpening since these have significantly less impact on the output than non-linear tone-mapping. More details on these secondary issues can be found elsewhere [3, 2, 24, 14].

[4]The boundary of the output gamut is determined automatically from image data in two steps. The edge directions of the parallelepiped are extracted from RAW metadata using dcraw[5], and then its scale is computed as a robust fit to RAW-JPEG correspondences.

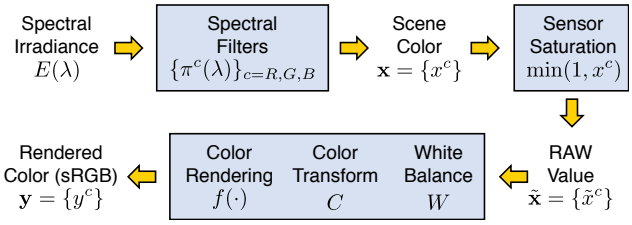

Figure 3. The forward color processing model used in this paper, along with our notation for it. Lesser effects, such as flare removal, de-mosaicking, and vignetting are ignored and treated as noise.

2.1. Forward (rendering) model

Referring to Fig. 3, the forward model begins with a collection of three idealized spectral sensors with sensitivity profiles $\{\pi^c(\lambda)\}_{c=R,G,B}$ that sample the spectral irradiance incident on the sensor plane. These sensors are idealized in the sense that they do not saturate and have infinite dynamic range, and we refer to their output $\mathbf{x} = \{x^c\}_{c=R,G,B}$ as the *scene color*. Real sensors have limited dynamic range, so scene colors are clipped as they are recorded. In some consumer cameras these recorded sensor measurements $\tilde{\mathbf{x}} = \{\tilde{x}^c\}_{c=R,G,B}$ are made available through a RAW output format, and in others they only exist internally. Empirical studies suggest that the RAW values (in the absence of clipping) are proportional to incident irradiance and related by a linear transform to measurements that would be obtained by the CIE standard observer [3, 2, 15] (also see Fig. 1). For this reason, they provide a "relative scene-referred image" [12] and can be used directly by computer vision systems to reason about spectral irradiance.

Two linear transforms are applied to the sensor measurements. The first (W) is scene-dependent and induces white balance, and the second (C) is a fixed transformation to an internal working color space. Then, most importantly, the linearly transformed RAW values $CW\tilde{\mathbf{x}}$ are rendered to colors $\mathbf{y} = \{y^c\}_{c=R,G,B}$ in the narrow-gamut output sRGB color space through a non-linear map $f\colon \mathbb{R}^3 \to \mathbb{R}^3$. This map has evolved to produce visually-pleasing results at the expense of physical accuracy, and since the quality of a camera's color rendering process plays a significant role in determining its commercial value, there is a dis-incentive for manufacturers to share its details. In our model, the map f includes the per-channel non-linearity (approximately a gamma of 2.2) that is part of the sRGB standard (IEC 61966-2-1:1999).

Fig. 2 shows signal values at various stages of this forward model for a consumer camera (DMC-LX3, Panasonic Inc.). In these graphs, the black box represents the range of possible RAW values $\tilde{\mathbf{x}}$, and the red parallelepiped marks the boundary of the output sRGB gamut to which all RAW values must be mapped into. The left plot shows the signals acquired for a single scene reflectance point over multiple

exposures. The scene colors $\mathbf{x}$ (black) lie a line that extends well beyond the cube. In this example, the scene colors are outside the sRGB gamut, and while the RAW values $\tilde{\mathbf{x}}$ (magenta) are very close to these scene colors for low exposures, they are clipped when the intensity grows large. Finally, the rendered colors $\mathbf{y} = f(CW\tilde{\mathbf{x}})$ (blue) lie within the output gamut, and are significantly affected by the combined effects of sensor saturation, white balance, and the color space transform. Interestingly, these rendered colors are relatively far inside the boundary of the sRGB gamut, so the conventional wisdom in radiometric calibration that one should discard pixels with very small or very large JPEG values as being "clipped" is unlikely to detect and properly treat them.

2.2. Inverse (de-rendering) model

Our goal is to infer, for each possible rendered color $\mathbf{y}$, the original scene color $\mathbf{x}$ that created it. As information is lost in the forward rendering process, exact recovery is not possible and thus any deterministic function that predicts a single point estimate is bound to be wrong much of the time. For that reason, we propose to estimate a *distribution* over the space of possible scene colors, which can be effectively adapted to applications that combine information on several images (e.g. high dynamic range imaging, photometric stereo, etc.). Specifically, we seek a representation of $p(\mathbf{x}|\mathbf{y})$ from which we can either obtain a MAP estimate of $\mathbf{x}$ or directly employ Bayesian inference as desired for a given application (see Sec. 3.1 and Sec.3.2).

We model the underlying de-rendering function, denoted z, using Gaussian Process (GP) regression [25]. Given a training set $\{\mathcal{D} = (\mathbf{y}_i, \mathbf{x}_i),\ i = 1, \cdots, N\}$, composed of inputs $\mathbf{y}_i$ and noisy outputs $\mathbf{x}_i$, we model the outputs x_i^c in each channel $c = 1, 2, 3$ separately as coming from a latent function z^c that has a prior distribution described by a GP, and corrupted by additive noise ϵ_i:

$$x_i^c = z^c(\mathbf{y}_i) + \epsilon_i, \quad \epsilon_i \propto \mathcal{N}(0, \sigma_n^2) \tag{1}$$

We can consider z to be the inverse of the mapping containing the color rendering function, color transform, and white balance operations depicted in Fig. 3; practically, we learn it on images for which the white balance has been fixed to remove scene-dependence. Our model assumes a spatially invariant de-rendering function.

The classic GP regression paradigm uses a single set of parameters defining the smoothness of the inferred function. However, our analysis of the camera data has revealed that such globally stationary smoothness is inadequate for our problem, as shown in Fig. 2. The variance of z must vary over local neighborhoods in the input space to model this phenomenon. We therefore employ a *local* GP regression model, which exploits the observation that, for compact radial covariance functions, only the points close to a test point will have significant influence on the results [30]. Given a training dataset and a test point, the method identifies the set of nearest neighbors to the test point, and learns a Gaussian Process on the fly using those nearest neighbors as training data, varying the covariance parameters locally.[5] Given a JPEG pixel observation $\mathbf{y}$, we infer a test distribution of RAW values conditioned on $\mathbf{y}$ as follows: we find the nearest neighbors to $\mathbf{y}$ in $\mathcal{D}$, $\mathcal{D}_{N(\mathbf{y})}$, and then obtain an estimate of the corresponding raw value using

$$p_x(\mathbf{x}|\mathbf{y}) = \prod_c p_{GP}(x^c|\mathcal{D}_{N(\mathbf{y})}, \mathbf{y}) \tag{2}$$

where $p_{GP}(x|D, \mathbf{y})$ is the conditional GP likelihood of x using training data D for $\mathbf{y}$.

3. Working with photometric uncertainty

Linear measurement of the scene irradiance is a crucial requirement for many computer vision algorithms (e.g. shape from shading, photometric stereo, image-based rendering, deblurring, color constancy, etc.), and the output of our de-rendering model can be readily used in such tasks. In this section, we describe two new methods enabled by the proposed probabilistic derendering model, showing how photometric uncertainty modeling is critical to obtain robust results.

3.1. Probabilistic wide gamut imaging

Many applications that use Internet images operate by inferring radiometric scene properties from multiple observations of the same scene point. For example, multiple observations under different illuminations can be exploited for inferring diffuse object color [22] or more general BRDFs [11]. To explore the benefits of uncertainty modeling in such cases, we consider an example scenario motivated by traditional HDR imaging with exposure stacks [20, 6]. Given as input multiple exposures of the same stationary scene, we seek to combine them into one floating-point, HDR, and wide-gamut image.

Assume we are given a sequence of JPEG sRGB vectors captured at shutter speeds of $\{\alpha_1, \alpha_2, \ldots, \alpha_N\}$ seconds. Represent these by $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$. We would like to predict the RAW image that would have been obtained with a shutter speed of α_0 seconds. Let's call this $\mathbf{x}_0$. Note that α_0 need not be one of the shutter speeds used to capture the JPEG input.

Given a training set $\mathcal{D}$ as described below, for each sRGB value $\mathbf{y}$ we estimate the conditional distributions $p_{x_i}(\mathbf{x}_i|\mathbf{y}_i)$

[5] To handle multimodality in the mapping, [30] shows how clustering may be performed in both input and output spaces for the training data, and a set of local regressors returned. However we believe that the inverse map does not have multimodal structure, and we found that a single local regressor provided adequate results as described in Sec. 4.

for the RAW value $\mathbf{x}_i$ that would have been obtained with shutter speed α_i.

To obtain a prediction for $\mathbf{x}_0$ we combine these as follows:

$$p_{x_0}(\mathbf{x}_0|\mathbf{y}_1,\ldots,\mathbf{y}_N) = \prod_i p_{x_0}(\mathbf{x}_0|\mathbf{y}_i) = \prod_i p_{x_i}\left(\frac{\alpha_i}{\alpha_0}\mathbf{x}_0|\mathbf{y}_i\right) \quad (3)$$

Since each channel $p_{x_i}(\mathbf{x}_i|\mathbf{y}_i)$ is modeled by a Gaussian process, then $p_{x_i}\left(\frac{\alpha_i}{\alpha_0}\mathbf{x}_0|\mathbf{y}_i\right)$ will have a Gaussian distribution, and so their product, the conditional distribution $p_{x_0}(\mathbf{x}_0|\mathbf{y}_1,\ldots,\mathbf{y}_N) = \prod_i p_{x_0}(\mathbf{x}_0|\mathbf{y}_i)$ will be Gaussian as well. Therefore, our output for $\mathbf{x}_0$ also provides both a mean and a variance.

From this application, we can see the power of our probablistic model: it gives an distribution estimate rather than a point estimate, which can be made use of when combining different estimate results, put more weights on the accurate estimates with small variance and less weights on the ones with large variance.

3.2. Probabilistic Lambertian photometric stereo

When illumination varies, another way to combine multiple observations of the same scene is to recover lighting information and/or scene geometry. This may be useful when using Internet images for weather recovery [27], geometric camera calibration [17], or 3D reconstruction [1]. To quantitatively assess the utility of uncertainty modeling in these types of applications we consider the toy problem of recovering from JPEG images three-dimensional scene shape using Lambertian photometric stereo.

Lambertian photometric stereo is a technique for estimating the surface normals of a Lambertian object by observing that object under different lighting conditions and a fixed viewpoint [31]. Suppose there are N different directional lighting conditions, and $\mathbf{l}_i$ is the direction of ith directional light source. Consider a single color channel of single pixel in the image plane, denote by I_i the *linear* intensity recorded under the ith light direction, and let $\mathbf{n}$ and ρ be the normal direction and the albedo of the surface patch at the back-projection of this pixel. Under the Lambertian model, we can write $\rho\langle\mathbf{l}_i,\mathbf{n}\rangle = I_i$, and the goal of photometric stereo is to infer ρ and $\mathbf{n}$ given the set $\{\mathbf{l}_i, I_i\}$.

Defining $\mathbf{b} = \rho\mathbf{n}$, (since $\mathbf{n}$ is a unit vector, $\mathbf{b}$ uniquely determines ρ and $\mathbf{n}$, and vice versa), the relation between intensity and light direction can be written as

$$\mathbf{l}_i^T\mathbf{b} = I_i \quad (4)$$

Given three or more $\{\mathbf{l}_i, I_i\}$-pairs, the traditional approach to Lambertian photometric stereo estimates $\mathbf{b}$ in a least square sense:

$$\mathbf{b} = (\boldsymbol{L}^T\boldsymbol{L})^{-1}\boldsymbol{L}^T\mathbf{I}, \quad (5)$$

where $\boldsymbol{L}$ and $\mathbf{I}$ are the matrix and vector formed by stacking the light directions $\mathbf{l}_i$ and measurements I_i, respectively.

The linear relation between $\mathbf{I}$ and scene irradiance is crucial in photometric stereo, and therefore a RAW measurement is required. However, if we only have access to the JPEG images for this task (which is a reasonable assumption for many internet-based vision problems), we can still recover the linear measurement using our GP model. In this case, the linear measurement of each pixel is described as a Gaussian random variable $I_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, and Equ. (4) can be written as

$$\mathbf{l}_i^T\mathbf{b} = \mu_i + \sigma_i\epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0,1) \quad (6)$$

As noted in [13], when each measurement has different uncertainty, the maximum likelihood estimator for Equ. (6) is a weighted least square, using the reciprocal of variance as weight. In this case, the solution is given by

$$\mathbf{b} = (\boldsymbol{L}^T\boldsymbol{W}\boldsymbol{L})^{-1}\boldsymbol{L}^T\boldsymbol{W}\boldsymbol{\mu} \quad (7)$$

where $\boldsymbol{W} = \text{diag}\{\sigma_i^{-2}\}_{i=1}^N$.

This application shows again how we can incorporate the uncertainty measurement produced by the model and get more robust results. The performance of our algorithm is shown in Sec. 4.3.

4. Evaluation

For training, we collect for each camera model densely sampled corresponding measurements of scene color and rendered color. We obtain these by capturing a set of registered RAW and JPEG images of a standard color chart (140-patch Digital ColorChecker SG, X-Rite Inc.). To get a complete coverage of the RAW space, we use various camera exposure settings (from all-black to all-white) and various illumination spectra (Lowel Pro tungsten lamp sequentially filtered by 16 distinct gels). This provides a much denser set of RAW/JPEG matches and contains more information on saturated and overexposed colors than is available in any existing database, such as [3] and [18]. We average the RAW and JPEG pixel values within each of the 140 color patches in each image to suppress the effects of demosaicking, noise, and compression, and all in all, we obtain between 30,000 and 50,000 RAW/JPEG color pairs $\{\tilde{\mathbf{x}}_i, \mathbf{y}_i\}$ for each camera. Scene colors $\mathbf{x}$ are obtained from RAW values $\tilde{\mathbf{x}}$ using dcraw [5] for demosaicking without white balance or a color space transform, which produces 16-bit uncompressed color images in the color space defined by the camera's spectral filters. RAW values corresponding to saturated sensor measurements are discarded and replaced

	rmse		rrmse	
	all	out	all	out
CANON 40D-baseline	.05	.09	.31	.36
CANON 40D-ours	.02	.03	.07	.09
CANON S90-baseline	.08	.14	.32	.49
CANON S90-ours	.03	.04	.13	.14
PANASONIC-baseline	.14	.09	.64	.56
PANASONIC-ours	.04	.03	.13	.16
FUJI-baseline	.24	n/a	1.46	n/a
FUJI-ours	.13	n/a	.39	n/a

Table 1. Derendering results. The "rmse" in the table head stands for root-mean-square error, and "rrmse" for *relative* root-mean-square error. The "all" means the error measurement is taken for all data point, as oppose to "out", where error is measured only with *out-of-gamut* colors.

by estimates of scene color $\mathbf{x}$ extrapolated from RAW measurements by the same camera under the same illuminant but with lower camera exposure settings.

Three of the cameras that we evaluate—two point and shoot models (Canon Powershot S90; Panasonic DMC-LX3) and a digital SLR (Canon EOS 40D)—provide simultaneous RAW and JPEG output, allowing training from each of these camera's data on its own. We also evaluate a fourth camera (Fuji FinePix J10) that provides only JPEG output, and for this we use one of the RAW-capable cameras (the Panasonic) as a proxy to collect the registered RAW images.

For GP regression, we use the GPML toolkit.[6] We tested linear and squared exponential (SE) kernels and found the latter to provide superior performance, perhaps because of the nonlinear nature of the rendering operation. The parameters of the SE kernel, as well as the parameters of the additive noise covariance on the outputs, were estimated via maximum likelihood for each local GP. In all experiments, we use k-means clustering with $k = 10$ and output the nearest cluster as the $\mathcal{D}_{N(\mathbf{y})}$ neighborhood for a test point.

4.1. De-rendering

To begin, we evaluate our ability to hallucinate scene colors from low-gamut images. We use as a baseline the deterministic representation proposed by Chakrabarti et al. [3], which suggested a forward rendering model consisting of a linear map C followed by a per-channel polynomial. Here we aim to recover the reverse mapping, so for our deterministic baseline, we invert their model numerically. We only consider the best (MAP) point estimate and ignore the uncertainty output of our model, therefore the derendering results of the proposed GP in this experiment is equivalent to using an RBF model in Lin et al. [18]. (Our advantage over [18] of providing confidence levels will be shown in the next two experiments.)

For each camera, we split the data points into training and testing sets at random, training on 5000 pairs $\{\mathbf{x}_i, \mathbf{y}_i\}$ and testing on the rest. This means that in this experiment, for each particular patch number and illumination, we may be training on some of the exposures, and testing on the rest. This experiment is designed to provide insight into the predictive power of our model, as compared to the baseline. We report both root mean squared error (RMSE) and relative RMSE between the ground truth scene color and each model's prediction. Because our dataset is dominated by lower-valued RGB colors, relative RMSE gives a better picture of the error as it accounts for the total brightness of the RGB vectors. Finally, we report separately the errors corresponding to datapoints that are outside of the sRGB gamut (29% of RAW colors captured by CANON are outside the sRGB gamut) because, as suggested by Fig. 2, these are more affected by color rendering.

The results are shown in Table 1, from which we can say the following: 1) our model achieves significantly lower mean errors than the deterministic baseline on all four cameras; 2) overall the errors are higher for the Fuji camera, which is not surprising since the original RAW values were not available and the Panasonic RAW was used; 3) for Canon and Panasonic, our model performs equally well for scene colors that are inside and outside of the sRGB gamut (we cannot identify them for the Fuji).

4.2. Wide gamut imaging

Here we follow a different experimental paradigm: we hold out all 22 multiple-exposure images taken under a single illumination as our test sequence, and train on a randomly sampled subset of 5K points from the rest. We repeat this for all 16 illuminants and average the results. Comparisons are made with a traditional HDR algorithm implemented in [6] with per-channel deterministic polynomial [3] and cross-channel RBF [18]. Results are shown in Fig. 4.

The results show that our GP model consistently outperform both HDR baselines, especially in the out-of-gamut region. Fig. 4 also provides a strong evidence that traditional deterministic algorithms are seriously affected by saturation and out-of-gamut colors. When we shrink the testing interval (taking only the middle trunk of the JPEG values), their performance is nontrivially improved. This is the reason that most previous work only considers the "non-saturated region" of the colors, which unfortunately differs from camera to camera. On the other hand, the prediction error of the proposed probablistic model is small and constant in all test intervals, since it takes account of confidence level (variance) of the estimate. The advantage of the GP model is more clear when only considering the out-of-gamut colors. As the traditional HDR algorithm limits its operation inside the sRGB gamut, it is unable to accurately infer those colors that are outside and therefore the prediction performance is poor. The GP model has the advantage of calibrating the

[6] Available online at http://www.gaussianprocess.org/gpml/

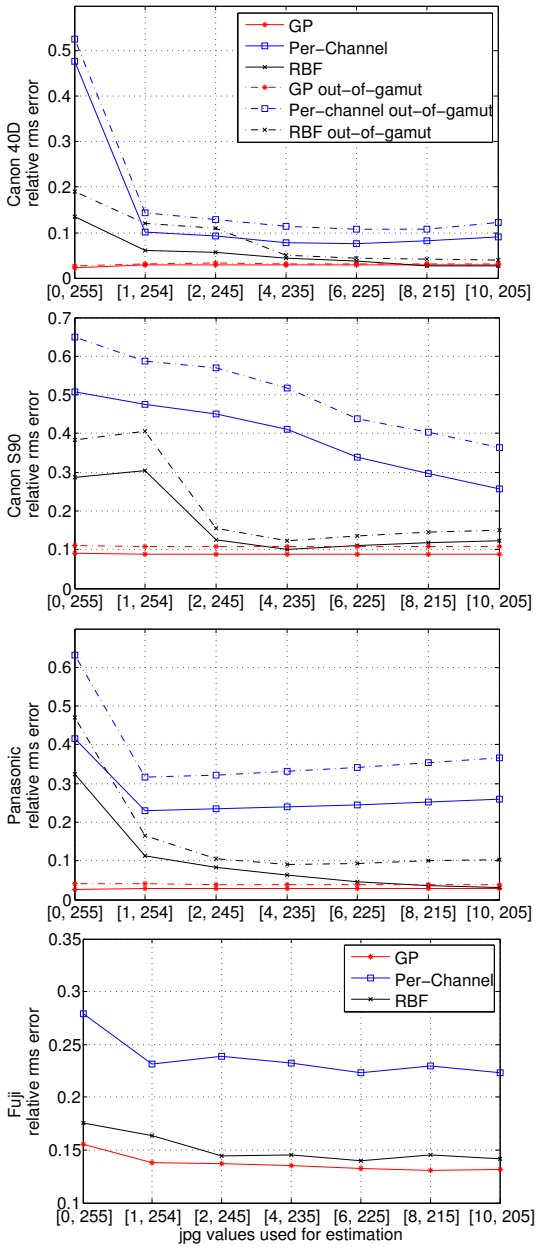

Figure 4. *Wide-gamut imaging:* Results of estimating wide-gamut linear scene colors from an exposure-sequence of sRGB JPEG images captured with 22 different exposures. Plots show relative RMSE in the predicted scene colors averaged over 16 runs with exposure sequences of the same scene under distinct illuminations. Along the x-axis shows the performance change of each model when looking at a smaller (non-saturated) subset of JPEG colors. Legends: GP (red) refers to the proposed algorithm; Per-channel (blue) refers to the algorithm in [3]; RBF (black) refers to the algorithm in [18].

camera on the whole wide-gamut space. From the results, we see that our prediction on out-of-gamut colors are almost as good as those in-gamut. Overall, our algorithm acheives an error that is on average (over all cameras) 0.482, or about half, of the RBF baseline's [18] error in the [0,255] interval, and a third (0.336) for out-of-gamut data in that interval.

4.3. Photometric Stereo

Finally, we evaluate our model for probabilistic Lambertian photometric stereo. For this we use the Canon EOS 40D to collect JPEG images of a wooden sphere from a fixed (approximately orthographic) viewpoint under directional illumination from twenty different known lighting directions. We apply the algorithms from Sec 3.2 to estimate the surface normal for each pixel that back-projects to the surface of the ball. Since the shape of the surface is known (i.e., it is defined by its occluding contour in the orthographic image plane) we can compare our results directly to ground truth.

The angular error (degrees) in the estimated surface normal at each pixel is displayed in the left of Fig. 5. The maximum likelihood estimates obtained with the proposed GP model are more accurate than those estimated by the baseline, in which JPEG values are deterministically de-rendered via [3] prior to least-squares estimation of the surface normals. The baseline method yields inaccurate estimates of the surface normals when the JPEG images contain near-saturated values. The third column shows the error that results from a second baseline using gamma-inversed JPEG values (a gamma parameter of 2.2 is assumed), and such errors are much larger, as expected. Quantitively, the average angular error of the proposed GP model is 3.41, for baseline model the error is 4.54, and for gamma-inversed JPEG the error is 8.92 (all the errors are measured in unit of degree).

As an additional comparison, we integrate each of the three normal vector fields to obtain a height field using [8], and show a one-dimensional cross section of each height field, corresponding to the horizontal scanline through the middle of each sphere. These are drawn in the right of Fig. 5 along with the ground truth shape, and we see that the proposed approach provides a more accurate result.

5. Conclusion

Most images captured and shared online are not in linear (RAW) formats, but are instead in narrow-gamut (sRGB) formats with colors that are severely distorted by cameras' color rendering processes. In order for computer vision systems to maximally exploit the color information in these images, they must first undo the color distortions as much as possible. This paper advocates a probabilistic approach to color de-rendering, one that embraces the multivalued nature of the de-rendering map by providing for each rendered

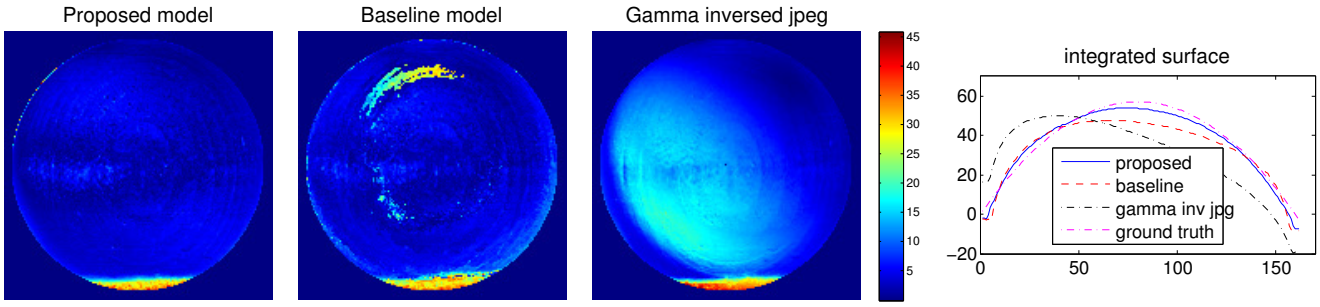

Figure 5. *Photometric stereo:* The left three figures show the angular errors in the per-pixel surface normals obtained using the proposed method, the deterministic baseline, and the gamma-inversed JPEG values (unit: degree). The right figure shows one-dimensional cross sections through surfaces obtained by integrating each set of surface normals and compares them to the ground truth shape.

sRGB color a distribution over the latent linear scene colors that could have induced it. An advantage of this approach is that it does not require discarding any image data using ad-hoc thresholds. Instead, it allows making use of all rendered color information by providing for each de-rendered color a measure of its uncertainty.

Our experimental results suggest that a probabilistic representation can be useful when combining per-image estimates of linear scene color, and when recovering the shape of Lambertian surfaces via photometry. The output of our approach—a mean and variance over scene colors for each sRGB image color—can have a practical impact for various other computer vision tasks (image-based modeling, object recognition, etc.). One direction worth exploring is the use of spatial structure in the input sRGB image(s), such as edges and textures, to further constrain the de-rendered scene colors. This is in the spirit of [29], and it leads one to wonder about the accuracy with which a full-gamut scene color image can be recovered from a single sRGB one.

References

[1] J. Ackermann, M. Ritz, A. Stork, and M. Goesele. Removing the example from photometric stereo by example. In *Proc. Workshop on Reconstruction and Modeling of Large-Scale 3D Virtual Environments*, 2010.

[2] M. Brady and G. Legge. Camera calibration for natural image studies and vision research. *JOSA A*, 26(1):30–42, 2009.

[3] A. Chakrabarti, D. Scharstein, and T. Zickler. An empirical camera model for internet color vision. In *BMVC*, 2009.

[4] W. Cleveland, S. Devlin, and E. Grosse. Regression by local fitting. In *Journal of Econometrics*, volume 37, pages 87–114, 1988.

[5] Decoding raw digital photos in linux. http://www.cybercom.net/ dcoffin/dcraw/, Last accessed: January 10, 2011.

[6] P. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH*, pages 369–378, 1997.

[7] H. Farid. Blind inverse gamma correction. *Image Processing, IEEE Transactions on*, 10(10):1428–1433, 2002.

[8] R. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *PAMI*, 10(4):439–451, 1988.

[9] M. Grossberg and S. Nayar. Determining the camera response from images: what is knowable? *PAMI*, pages 1455–1467, 2003.

[10] M. Grossberg and S. Nayar. Modeling the space of camera response functions. *PAMI*, 26(10):1272–1282, 2004.

[11] T. Haber, C. Fuchs, P. Bekaer, H. Seidel, M. Goesele, and H. Lensch. Relighting objects from image collections. In *CVPR*, 2009.

[12] D. Hasler and S. Süsstrunk. Mapping colour in image stitching applications. *Journal of Visual Communication and Image Representation*, 15(1):65–90, 2004.

[13] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.

[14] J. Holm, I. Tastl, L. Hanlon, and P. Hubel. Color processing for digital photography. In P. Green and L. MacDonald, editors, *Colour Engineering: Achieving Device Independent Colour*, pages 179–220. Wiley, 2002.

[15] M. Kim and J. Kautz. Characterization for high dynamic range imaging. *Computer Graphics Forum (Proc. EGSR)*, 27(2):691–697, 2008.

[16] S. Kuthirummal, A. Agarwala, D. Goldman, and S. Nayar. Priors for large photo collections and what they reveal about cameras. In *ECCV*, 2008.

[17] J. Lalonde, S. Narasimhan, and A. Efros. What do the sun and the sky tell us about the camera? *IJCV*, 88(1):24–51, 2010.

[18] H. Lin, S. J. Kim, S. Susstrunk, and M. Brown. Revisiting radiometric calibration for color computer vision. In *ICCV*, 2011.

[19] S. Lin, J. Gu, S. Yamazaki, and H.-Y. Shum. Radiometric calibration from a single image. In *CVPR*, 2004.

[20] S. Mann and R. Picard. Being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. In *Proc. IS&T Annual Conf.*, pages 422–428, 1995.

[21] T. Mitsunaga and S. Nayar. Radiometric self calibration. In *CVPR*, 1999.

[22] T. Owens, K. Saenko, A. Chakrabarti, Y. Xiong, T. Zickler, and T. Darrell. Learning object color models from multi-view constraints. In *CVPR*, pages 169–176, 2011.

[23] C. Pal, R. Szeliski, M. Uyttendaele, and N. Jojic. Probability models for high dynamic range imaging. In *CVPR*, 2004.

[24] R. Ramanath, W. Snyder, Y. Yoo, and M. Drew. Color image processing pipeline. *IEEE Signal Processing Magazine*, 22(1):34–43, 2005.

[25] C. R. Rasmussen and C. K. Williams. *Gaussian Process for Machine Learning*. MIT Press, 2006.

[26] E. Reinhard, G. Ward, S. Pattanaik, and P. Debevec. *High dynamic range imaging*. Elsevier, 2006.

[27] L. Shen and P. Tan. Photometric stereo and weather estimation using internet images. In *CVPR*, pages 1850–1857. IEEE, 2009.

[28] B. Shi, Y. Matsushita, Y. Wei, C. Xu, and P. Tan. Self-calibrating photometric stereo. In *CVPR*, 2010.

[29] K. E. Spaulding, A. C. Gallagher, E. B. Gindele, and R. W. Ptucha. Constructing extended color gamut images from limited color gamut digital images. U.S. Patent No. 7,308,135, 2007.

[30] R. Urtasun and T. Darrell. Sparse probabilistic regression for activity-independent human pose inference. In *CVPR*, 2008.

[31] R. Woodham. Photometric method for determining surface orientation from multiple images. *Optical engineering*, 19(1):139–144, 1980.