

Statistical Calibration of CCD Imaging Process

Yanghai Tsin *
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Visvanathan Ramesh
Imaging Department
Siemens Corporate Research
Princeton, NJ 08540

Takeo Kanade
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Charge-Coupled Device (CCD) cameras are widely used imaging sensors in computer vision systems. Many photometric algorithms, such as shape from shading, color constancy, and photometric stereo, implicitly assume that the image intensity is proportional to scene radiance. The actual image measurements deviate significantly from this assumption since the transformation from scene radiance to image intensity is non-linear and is a function of various factors including: noise sources in the CCD sensor, as well as various transformations occurring in the camera including: white balancing, gamma correction and automatic gain control. This paper illustrates how careful modelling of the error sources and the various processing steps enable us to accurately estimate the “response function”, the inverse mapping from image measurements to scene radiance for a given camera exposure setting. It is shown that the estimation algorithm outperforms the calibration procedures known to us in terms of reduced bias and variance. Further, we demonstrate how the error modelling helps us to obtain uncertainty estimates of the camera irradiance value. The power of this uncertainty modeling is illustrated by a vision task involving High Dynamic Range image generation followed by change detection. Change can be detected reliably even in situation where the two images (the reference scene image and the current image) are taken several hours apart.

1 Introduction

Charge-Coupled Device (CCD) cameras are used widely in computer vision systems. Many photometric algorithms, such as shape from shading [8], color constancy [11], and photometric stereo [15], implicitly assume that the image intensity is proportional to scene radiance. However, the CCD imaging process usually deviates from this ideal linear model. This is due to the fact that there are several independent sources of error in the sensor and that the operation

pipeline in the camera includes additional operators such as color correction/white balancing and gamma correction. The camera non-linearities make the above mentioned vision algorithms to be systematically biased since they expect the real scene irradiance as input. Thus it is desirable to calibrate the process of imaging, including typical operations such as γ -correction and white balancing. Given a static scene with constant camera irradiance, the calibration procedure aims to estimate the camera response function, i.e. the mapping from the cross product space of image intensity and exposure time to camera irradiance, by acquiring images with multiple exposure settings. Since the irradiance is assumed to be constant within the acquisition period, the calibration step essentially estimates the response function by minimizing prediction disagreements across the images.

The calibration can be greatly improved if the noise characteristics and the various processing steps in the conventional CCD sensor are modeled carefully. In this paper we present a calibration technique that is motivated from statistical model of the transformation steps and errors in the CCD camera. It is shown that the model helps in obtaining accurate estimates of the camera response function, the scene radiance, and its uncertainty. We illustrate how the calibration results can be used in a vision operation sequence involving High Dynamic Range (HDR) image generation followed by illumination invariant change detection. The knowledge of uncertainty in the radiance is used to automatically setup the change detection measure threshold so that a given false alarm rate can be achieved. The paper is organized as follows. Section 2 describes the CCD imaging process and the error sources at various steps. Section 3 describes the probabilistic generative model for the image observations. Section 3.1 describes the statistical model and the iterative calibration algorithm. Section 4 describes the experiments performed on synthetic and real data to validate our calibration algorithm. Section 5 discusses the application of the calibration method to perform illumination invariant change detection. We conclude in section 6 with a discussion and outlook.

In this paper we denote true values of scalars as Roman

*The first author acknowledges the research fellowship from Siemens Corporate Research.

letters, such as g . Observations or estimates of a random variable are denoted with a hat, \hat{g} . Matrices and vectors are denoted by using bold face letters.

2 Imaging Process

Before we address the problem of estimating the camera response function, we first provide an overview of the nature of various transformations and error sources in a typical color CCD camera.

A typical CCD camera imaging process is shown in figure 1. The camera irradiance, i.e. the energy incident on the CCD is usually considered to be proportional to the attenuated scene radiance under normal weather conditions [8]. The lens system transmits this camera irradiance unevenly because of factors including: the $\cos^4\theta$ law and *vignetting* [8]. Modern cameras usually correct the “ $\cos^4\theta$ ” distortion while the *vignetting* effect could be accounted for by calibration (see for instance [2]). The light energy transmitted through the lens system is then converted to electrons in the CCD array. In color CCD cameras, a red, green, blue filter array (called the *Bayer pattern*) is used in the sensor. Two rows in a *Bayer pattern* consists of a sequence of alternating red/green filter elements in one row followed by alternating green/blue filter elements in the subsequent row. The photo-electronic transformation is corrupted by various sources of errors (noise). For an excellent review the readers are referred to [7]. Due to manufacturing defects, the CCD well sizes fluctuate with a small variance. Thus, different CCD units output different voltage for the same light input. This phenomenon is called the *fixed pattern noise* and is usually modeled by a constant multiplier associated with each site. In addition, there is the *dark current* due to the thermal energy. The dark current usually adds a constant offset to the observed intensity, plus a small fluctuation which can be modeled as Gaussian noise. In this paper we will model the ideal CCD output, after scaled by the fixed pattern noise and offset by the dark current, as E . The CCD unit response is integrated over an exposure time t . Usually E is corrupted by the *shot noise* N_s and the thermal noise N_{c1} . The variance of the shot noise is proportional to Et . The output of the CCD unit can then be modeled as following.

$$\hat{E}_1 = tE + N_s + N_{c1} \quad (1)$$

To obtain the color image values from the Bayer Image, some form of interpolation (e.g. bilinear interpolation) is carried out to obtain the full resolution color image [1]. This operation is followed by *white balancing* [17], a processing step that attempts to generate output images that are consistent with human perception of color under different illumination conditions. In this paper we will model white balancing as separate transformations on each channel involving the scaling of values and an offset. Now equation

(1) can be rewritten as the following.

$$\hat{E}_2 = aE + N_s + N_{c1} + b \quad (2)$$

Here a is the combined scaling factor that incorporates the scale due to white balancing and the exposure time.

To account for the non-linear transfer function relating the input voltage in the display monitor to the display brightness, γ correction is usually applied to the sensed image value. Considering other noise effects such as: quantization error, amplifier noise, D/A and A/D noise, as an additional noise term N_{c2} with a constant variance, we have the expression for the observed image measurements:

$$\hat{z} = f(aE + N_s + N_{c1} + b) + N_{c2} \quad (3)$$

3 Calibration algorithm

We are interested in the estimation of the inverse function $g = f^{-1}$. We call this function the *response function* of the imaging process. Using equation (3) and using a Taylor series approximation:

$$g(\hat{z}) \approx aE + N_s + N_{c1} + b + g'(\hat{z})N_{c2} \quad (4)$$

Without noise, the true values have the following relation.

$$g(z) = aE + b \quad (5)$$

We write the zero mean noise term as

$$N(a, \hat{z}) = N_s + N_{c1} + g'(\hat{z})N_{c2} \quad (6)$$

The variance of the noise term is given by

$$\sigma^2(a, \hat{z}) = aE\sigma_s^2 + \sigma_{c1}^2 + g'(\hat{z})^2\sigma_{c2}^2 \quad (7)$$

While the response function can be modeled by a continuous/smooth parametric function, we simplify the problem by following [5] to compute values of g at discrete values only. In this paper we will assume 256 gray levels are used.

Besides the error sources mentioned above, other factors contribute to deviations in the image measurements. These factors include, but are not limited to, the following: blooming, color aberration, and spectral sensitivity. The readers are referred to [16] for further discussion. In this paper we deal with blooming by removing the infected pixels, i.e. the pixels surrounding a saturated pixel within a certain radius. The radius is usually 3 pixels for our cameras and is chosen empirically. To perform the calibration of the imaging process, we essentially use the same idea as the approaches in [12], [5], and [14]. We assume a constant, but spatially varying, camera irradiance, E , (within the given acquisition time) at a given pixel. Under this assumption, it is clear that spatially varying errors such as fixed pattern noise, dark current noise and vignetting are not relevant since these spatial

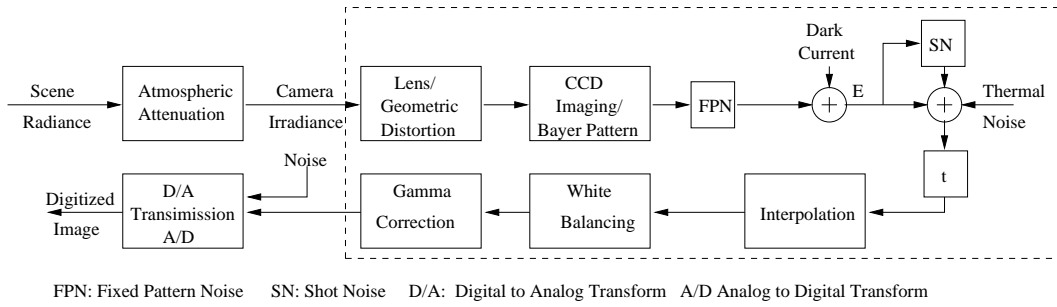


Figure 1: CCD camera imaging pipeline

variations are decoupled from the temporal random noise. The prediction of E can be done by using our statistical model from a collection of image measurements at a given pixel (with various exposure settings). When calibration of g , a and b is finished, other error sources can be calibrated using [7] and [2]. In this paper our focus is only on the estimation of the response function g , white balancing coefficients a and b .

We build upon the approaches of [5][14], with some major improvements. First, instead of considering a small group of representative pixels for calibration purposes, we will study all the valid pixels. Statistics on pools of pixels helps us to properly form and weight calibration functions. Second, we model white balancing explicitly to account for color correction that is carried out during the process of taking images. Third, the noise is modeled and the variance is computed. Consequently, when we construct our HDR image, the blending step involves a meaningful weight function and the output HDR map has an predictable variance map. Fourth, we realize the problem of *u-ambiguity* is worse than the authors of [14] have expected. Moderate noise level and small error tolerance bring their solution to a trivial one. We put constraints on the variations of exposure times to avoid the trivial solutions. The next subsection details the statistical estimation process.

3.1 Parameter estimation

The estimation problem will be formally defined in this section. Let $\mathcal{I}(x) = \hat{z}(x)$ denote the image measurement at a given spatial location x . A set of images $\mathcal{I}_1, \mathcal{I}_2, \dots, \mathcal{I}_N$ of a static scene is taken under different exposure settings t_1, t_2, \dots, t_N . Denote the observed intensity value in the k^{th} image at pixel location x as $\hat{z}_k(x)$. At each pixel x , we observe a pixel trace over the N images $\hat{z}_1(x), \hat{z}_2(x), \dots, \hat{z}_N(x)$. They all correspond to the same unknown $E(x)$. Denote by $\mathcal{P}(k, \hat{z}, \epsilon) = \{x : \|\mathcal{I}_k(x) - \hat{z}\| < \epsilon\}$, the *pool* of pixels in the k^{th} image that centered at

\hat{z} with a radius ϵ . Then from equations (4) and (6):

$$g(\hat{z}_k(x)) = E(x)a_k + b_k + N(a_k, \hat{z}_k). \quad (8)$$

The objective is to obtain the maximum-likelihood estimate of the response function $g(\cdot)$, a_k , b_k and $E(x)$. This is done in an iterative fashion. Let $\hat{g}^{(j)}(\cdot)$, $\hat{a}_k^{(j)}$, $\hat{b}_k^{(j)}$ denote the estimates at the start of the j^{th} iteration. Given these values it is possible to predict $\hat{E}^{(j)}(x, k)$, $k = 1, \dots, N$ and the weighted average of these predictions across the pixel trace provides the iteration estimate (of the true unknown scene radiance $E(x)$), $\hat{E}^{(j)}(x)$ as seen in equation (9). The weights are set to a constant value in our calibration process.

$$\hat{E}^{(j)}(x, k) = \frac{\hat{g}^{(j)}(\hat{z}_k(x)) - \hat{b}_k^{(j)}}{\hat{a}_k^{(j)}}, k = 1, 2, \dots, N$$

$$\hat{E}^{(j)}(x) = \sum_{i=1}^N \hat{w}_i^{(j)} \frac{\hat{g}^{(j)}(\hat{z}_i(x)) - \hat{b}_i^{(j)}}{\hat{a}_i^{(j)}} \quad (9)$$

Given the new estimates for the scene radiance, we can recover the residuals $\hat{e}_k^{(j)}(x, \hat{z}_k)$, and the mean error across pools $\hat{e}_k^{(j)}(\hat{z}_k)$. In addition, we can recover the standard deviation of the errors in scene radiance estimate $\hat{\sigma}_k^{(j)}(\hat{z}_k)$ by using the residuals in each pixel pool. To deal with outliers we use robust versions of the above estimates. For example, we use the 66 percent trimmed mean in our estimation step involving $\hat{e}_k^{(j)}(\hat{z}_k)$ (see equation (11)). Note that $\phi(\cdot)$ in this equation corresponds to the trimmed mean estimator. In addition, in equation (12) $\hat{\sigma}_k^{(j)}(\hat{z}_k)$ uses the median of the residuals to estimate the standard deviation.

$$\hat{e}_k^{(j)}(x, \hat{z}_k) = \hat{a}_k^{(j)} \hat{E}^{(j)}(x) + \hat{b}_k^{(j)} - \hat{g}^{(j)}(\hat{z}_k) \quad (10)$$

$$\hat{e}_k^{(j)}(\hat{z}_k) = \phi(\hat{e}_k^{(j)}(x, \hat{z}_k), x \in \mathcal{P}(k, \hat{z}, \epsilon)) \quad (11)$$

$$\frac{1}{\hat{w}_k^{(j)}} = \hat{\sigma}_k^{(j)}(\hat{z}_k)$$

$$= c \cdot \text{median}_{x \in \mathcal{P}(k, \hat{z}, \epsilon)} \|\hat{e}_k(x, \hat{z}_k) - \hat{e}_k^{(j)}(\hat{z}_k)\| \quad (12)$$

Denote by $e_k(\hat{z}_k)$ the theoretical errors at each iteration as a function of $\Delta g(\cdot)$, Δa , and Δb , then it can be shown that:

$$\begin{aligned} e_k(\hat{z}_k) &= \mu(e_k(x, \hat{z}_k)) \\ &\approx \sum_{i \neq k} w_i \frac{a_k}{a_i} [\Delta g(\hat{z}_i(x)) - \Delta b_i - E\Delta a] \\ &\quad + (w_k - 1) [\Delta g(\hat{z}_k) - \Delta b_k - E\Delta a_k] \\ &\stackrel{\text{def}}{=} \mathcal{F}(\Delta \mathbf{g}, \Delta \mathbf{a}, \Delta \mathbf{b}, \hat{z}_k, k) \end{aligned} \quad (13)$$

The constant c in (12) is affected by two issues. First, a correction factor has to be applied since a robust estimate of σ is obtained by taking the median of the absolute deviations. This factor is 1.483 for the Gaussian distribution. A second factor has to be multiplied to correct for the correlations induced by the Bayer pattern in the color image. For color images, neighboring pixels are usually correlated due to interpolation of the Bayer pattern. While independence can be achieved by pre-selection of pixels, we model the impact of the correlations on our variance prediction and correct for it. This correction factor can be shown to be 1.265 for green bands and 1.333 for red and blue bands.

Define the updates $\Delta \mathbf{g}$, $\Delta \mathbf{a}$ and $\Delta \mathbf{b}$:

$$\begin{aligned} \Delta \mathbf{g} &= \{\Delta g(z)\}, z = 1, 2, \dots, 256, \\ \Delta \mathbf{a} &= a_i, i = 1, 2, \dots, N, \text{ and}, \\ \Delta \mathbf{b} &= b_i, i = 1, 2, \dots, N \end{aligned} \quad (14)$$

The update step at each iteration involves the computation of the updates by determining:

$$\underset{\Delta \mathbf{g}, \Delta \mathbf{a}, \Delta \mathbf{b}}{\operatorname{argmin}} \sum_{k, \hat{z}_k} \frac{(\hat{e}_k^2(\hat{z}_k) - \mathcal{F}(\Delta \mathbf{g}, \Delta \mathbf{a}, \Delta \mathbf{b}, \hat{z}_k, k))^2}{\hat{\sigma}_k^2(\hat{z}_k)} \quad (15)$$

and updating the current estimates:

$$\begin{aligned} \hat{\mathbf{g}}^{(j+1)} &= \hat{\mathbf{g}}^{(j)} - \eta \Delta \mathbf{g} \\ \hat{\mathbf{a}}^{(j+1)} &= \hat{\mathbf{a}}^{(j)} - \eta \Delta \mathbf{a} \\ \hat{\mathbf{b}}^{(j+1)} &= \hat{\mathbf{b}}^{(j)} - \eta \Delta \mathbf{b} \end{aligned} \quad (16)$$

At the end of the iterations, we use the relationship given by (7) to obtain a system of linear equations relating the parameters σ_{c1} , σ_{c2} , and σ_s to $\sigma^2(a, z)$. The estimates $\hat{\sigma}_k(\hat{z}_k)$, $\hat{g}(\cdot)$, $\hat{E}(\cdot)$, and \hat{a}_k are used as measurements and a least squares fitting step is used to estimate the unknown CCD sensor parameters. The system of equations relating the ideal parameters is given below:

$$\sigma_k^2(\hat{z}_k) = \sum_{i \neq k} w_i^2 \sigma^2(a_i, z_i) + (w_k - 1)^2 \sigma^2(a_k, \hat{z}_k) \quad (17)$$

After the calibration has finished, and the variances of the noise estimated, the HDR image can be blended using (9). The weighting function is given by

$$w_k = \frac{a_k / \sigma(a_k, \hat{z}_k)}{\sum_i a_i / \sigma(a_i, \hat{z}_i)} \quad (18)$$

The standard deviation of scene radiance estimation (9) is given by

$$\sigma_E(x) = \frac{\sqrt{N}}{\sum_i a_i / \sigma(a_i, \hat{z}_i)} \quad (19)$$

3.2 Iterative Calibration Algorithm

While the above estimation algorithm is in theory reasonable, we have to adapt it to address practical considerations. First, there are several types of known outliers in a typical digitized image, including boundary pixels and blooming contaminated pixels. They can be easily identified and the corresponding pixels can be taken care of by preprocessing the input image. Equations of form (13) must be properly constrained to avoid trivial solutions. Notice that what we are going to recover is the affine transformed version of the original scene radiance. The true scene radiance remains unknown without first calibrating for the spatial variations and the dark current. Therefore, a shifted and scaled version of $g(z)$ gives the same consistency over different images. To fix the ambiguity, it is necessary to fix the values of two points of the response function. Let the two points be x_1 and x_2 . $\Delta g(x_1)$ and $\Delta g(x_2)$ are set to zero once $g(x_1)$ and $g(x_2)$ have been initialized. A further ambiguity arises when different b_k 's are allowed for different images. For instance, when we add a value $c \cdot a_k$ to b_k , different images agree the same. To remove this ambiguity, we fix one of the b_k 's and let $\Delta b_k = 0$. These constraints can be used directly by setting appropriate values to zeroes in the equations of form (13).

In [14] the authors mentioned the so called *u-ambiguity* arose when both \mathbf{g} and \mathbf{a} are allowed to change. Consider the case when $b_k = 0$ for all k . The ideal imaging process becomes $g(z) = Ea$. It is easy to see that the imaging process can also be interpreted as scene E^u scaled by a^u , with corresponding response curve $g^u(z)$. In one extreme case, $u = 0$, the response curve is flat, and all the a_k 's are the same. This will make arbitrary images "agree" on each other. Mitsunaga and Nayar ([14]) claimed that their polynomial formulation alleviates this ambiguity if the true response curve is indeed a polynomial. However, the true response curve can seldom be represented exactly by a low degree polynomial. The model approximation error translates to a larger variance in the estimated response function. This problem is fundamental and cannot be solved without ground truth of the scene radiance. In our algorithm, we bound the changes in the scaling factors by placing *a priori* knowledge on them. For example, we punish large deviations of a_i from the nominal exposure times. For electronically controlled shutters and small white balancing, this is usually a reasonable assumption.

Finally, we put smoothness and monotone constraints on the response function. To enforce smoothness we add an ex-

tra term $\lambda \sum_{k=2}^{254} (\Delta g(k-1) - 2\Delta g(k) + \Delta g(k+1))$ to the cost function (15). Usually the response curve is monotonically increasing. Algorithms to solve order constrained optimization problem are within the field of isotonic regression [3]. To simplify the problem we post process the estimated response function by *pool-adjacent-violators (PAV)* algorithm. More rigorous algorithms, such as quadratic-programming should be used if the order constraint is to be imposed along with the optimization.

The algorithm can now be summarized as follows:

1. Preprocess the image. This involves:
 - Taking out all the outlier pixels;
 - Building a look up table to all the pooled pixel sets $\mathcal{P}(k, \hat{z}, \epsilon)$;
 - Building pixel traces, and estimating the most likely intensity values that correspond to a pool under a given exposure. This is done by taking the median of the intensity values of the pooled pixels for a given exposure.
2. Initialize the response curve to be a straight line passing through the fixed point x_1 and x_2 . Set the initial scale factors a_i to be the nominal exposure times, and the initial offsets to be all zeros. For color images, each channel has its own g, a and b . Each channel is calibrated separately.
3. Build HDR using (9). In the place of g, a and b use current estimate.
4. Calculate the prediction error for each pixel using (10).
5. Estimate $\hat{e}_k(\hat{z}_k)$ and $\hat{\sigma}_k^2(\hat{z}_k)$ from equations (11) and (12). Form equations of (13) and (17) for each pool of pixels.
6. Enforce smoothness constraints. The constraints $\Delta g(x_i) = 0, i = 1, 2$ can be approximated by linear equations $K\Delta g(x_i) = 0$, where K is a large number.
7. Solve for $\Delta g, \Delta a$ and $\Delta b, \sigma_s, \sigma_{c1}$, and σ_{c2} by using equations (15) and (17).
8. Update \hat{g}, \hat{a} and \hat{b} using (16);
9. Impose monotonic constraints by pool-adjacent-violator (PAV) algorithm.
10. Iterate 3 until the 1-norm of $\Delta \mathbf{g}$ is smaller than error tolerance.

4 Experiments

4.1 Simulated Images

In each sequence eight images were generated. Consecutive images had exposure ratio of $\sqrt{2}$. The response function was $g(z) = (z/255)^{1.2}, z = 0, 1, 2, \dots, 255$. The radiance E was designed to have uniform distribution in $[0, 1]$. Pixel pools on average had about 200 elements. Pools with at least 100 pixels were considered. The noise levels were set to be $\sigma_s = 0.015, \sigma_{c1} = 0.01$ and $\sigma_{c2} = 1$. These noise terms were chosen to simulate real camera noises.

To compare our algorithm with that of [5] and [14], 50 sequences were produced without white balancing. We applied all three algorithms on the same data set.

For each algorithm, response functions estimated from the 50 sequences were compared with the true one. Average and standard deviation of the response function error were computed. Though the algorithms of [5] and [14] had already achieved high accuracy, less than 0.5% at high intensity end, our algorithm showed further improvement in terms of reduced bias 2(a) and variance 2(b).

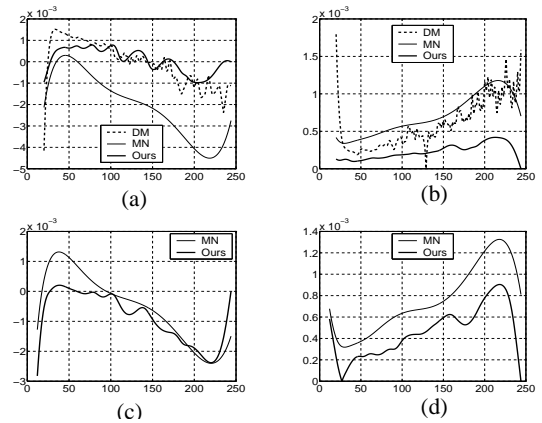


Figure 2:

Comparison on the response function estimation error: First row, estimation errors without white balancing; second row, with white balancing. (a)(c) average estimation errors for the 50 sequences; (b)(d) standard deviation of the estimation errors.

In addition to the response function, our algorithm output estimation of the variance term. The average estimated σ_s, σ_{c1} and σ_{c2} were 0.0150, 0.0101, and 0.8617, with standard deviation 0.0004, 0.0003 and 0.3839. We didn't use any constrained optimization on the variance terms. 4 out of 50 sequences output negative value on σ_{c2} and were forced to be 0 after optimization. If the 4 sequences were taken out, average of the variance term σ_{c2} for the remaining 46 sequences were 0.9366 with standard deviation 0.2978.

Readers should realize the limitations of the noise model

(7). When calibrated, $aE \approx g(z)$ for a given pixel. If $g(z)$ has similar function form as $g'(z)^2$, it is not easy to discriminate the individual contributions of σ_s^2 and σ_{c2}^2 . For example, when $g(z) = z^2 = g'(z)^2$, it is impossible to quantify shot noise individually. In such case, however, the total noise variance can still be estimated by combining σ_s^2 and σ_{c2}^2 .

Next, 50 sequences with white balancing introduced were studied. Our algorithm was able to recover both the scale factor and offset reliably. The nominal scale factor a was multiplied by a Gaussian random variable with mean 1 and standard deviation 0.05 and was provided to the program. Our algorithm was able to recover the scale factor to the accuracy of 0.5% in the above mentioned noise settings. Our algorithm also recovered the offset factor reliably. The extent to which the offset can be recovered especially depends on the noise term N_{c1} . In all the cases, the larger the pixel pool size, the better the accuracy on estimation. Performance of [14] were comparable (see figure 2(c)(d)). Our algorithm has smaller deviations.

4.2 Real Images

We took 21 sequences of outdoor images using a Sony EVI 330 camera. One representative image is shown in figure 3(a). We changed shutter speed ranging from 1/60 to 1/10000 in 14 steps to capture the high dynamic scene. Figure 3(b) is the pixel pool size histogram of the image in (a).

Figure 3(c) shows the histogram of the prediction error $e_k(x, \hat{z}_k)$ in the pool centered at 199 before calibration. Initially the bias is large, about 30% relative error. After a couple of iterations our algorithm is able to bring the whole pool to very small bias 3(d).

Estimated response curves for the sequence in figure 3(a) are shown in 3(e). Curves of the three channels are super-imposed and it is interesting to observe that the three curves are very close to each other despite that they were estimated independently. Figure 3(f) shows the results of camera variance estimation step using the model equations (17). Over the 21 sequences, the average estimated $\sigma_s, \sigma_{c1}, \sigma_{c2}$ are 0.0138, 0.0037, 0.9965 for the green channel, with standard deviation 0.0025, 0.0007, and 0.5473 respectively.

We further tested our algorithm on the Cathedral sequence downloaded from the web site of Debevec [5]. As a comparison, we ran the code of both [5] and [14] on the set of images. The estimated response curves and the blended HDR images were then used to test the prediction agreement (10)(11) of different images. Representative prediction errors are shown in figure (4). Our algorithm gives much better accuracy than those of [5] and [14].

We don't have ground truth white balancing information for our data set. However, we frequently observed much improved agreement across images when scale and offset

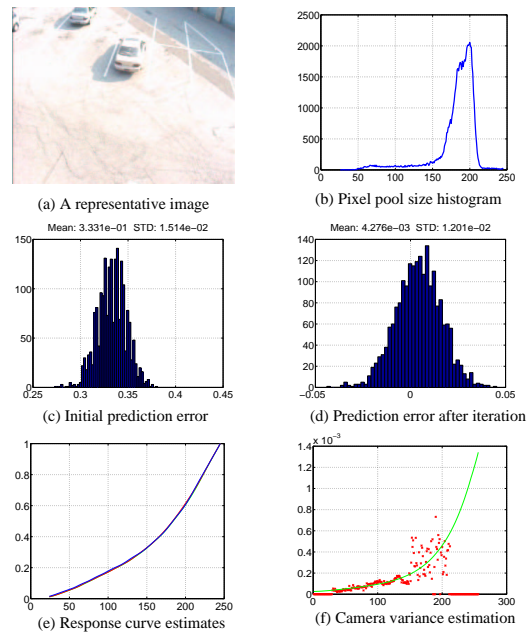


Figure 3:

Calibration of a Sony camera. See text for explanations..

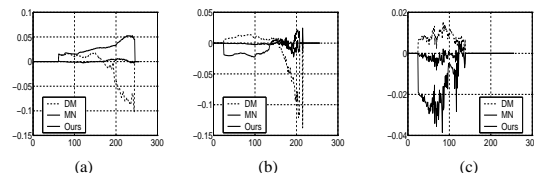


Figure 4:

Real image results: (a)-(c) prediction errors for images 1,6 and 12 in the Cathedral sequence.

factors are allowed to be calibrated. Intuitively scale factors change the slope of curves shown in figure 4, and offset factors shifts the curves in vertical direction. Our better performance in the Cathedral sequence is largely due to these extra degrees of freedom.

5 Application: Illumination Invariant Change Detection

Since our calibration provides an uncertainty estimate of $E(x)$, we believe it is possible to make use of this information to demonstrate significant improvements of various vision algorithms such as change detection, color constancy, etc. In this paper, we address the problem of change detection under varying illumination.

For simplicity we only consider the case of static Lambertian scene, static camera, and planar background. In our

example we took a sequence of images of a empty parking lot. Hours later we took a second set of images. In both cases images were calibrated, and HDR images with uncertainty model were obtained. Two such images are shown in figure (5a,c). Our goal is to detect the parked cars under significantly changed lighting while having the ability to discriminate shadow from the changes due to reflectance.

Past works have shown that the illumination change can be modeled as linear transformations in the RGB space (see for instance [20]). The argument is supported by early research that the surface reflectance can be represented by low dimensional linear systems [10]. Representing the measured color vector at a pixel x as a row vector $E(x) \in \mathcal{R}^{1 \times 3}$, and the radiance of the same pixel under a different illumination as $E'(x)$, the authors of [20] claimed $E'(x) = E(x)\mathbf{M}$, $\mathbf{M} \in \mathcal{R}^{3 \times 3}$. If a population of K pixels are coplanar in the 3D world, and they are illuminated by the same light source in each case, they all obey the same transformation \mathbf{M} . The transformation can be estimated using the observed radiance at these pixels. Thus we can stack all the color vectors into two $K \times 3$ matrices $\hat{\mathbf{E}}$ and $\hat{\mathbf{E}}'$, and we have the system of equations:

$$\hat{\mathbf{E}}\mathbf{M} = \hat{\mathbf{E}}' \quad (20)$$

The estimate $\hat{\mathbf{M}}$ that minimizes the error in a least squares sense can be computed.

Let $\hat{\mathbf{E}} = \mathbf{E} + \mathbf{N}_{\mathbf{E}}$ and $\hat{\mathbf{E}}' = \mathbf{E}' + \mathbf{N}'_{\mathbf{E}}$. Here \mathbf{E} is composed of the true radiance and $\mathbf{N}_{\mathbf{E}}$ is composed of zero mean independent random variables whose variances are estimated by (19). Using standard perturbation analysis techniques [18] (i.e. first order error propagation) the covariance matrix of the prediction error at a given pixel x , $e(x) = \hat{\mathbf{E}}(x)\mathbf{M} - \hat{\mathbf{E}}'(x)$, can be calculated as a function of $\Sigma_{\mathbf{E}}$ and $\Sigma_{\mathbf{E}'}$ and the E and E' . Here $\hat{\mathbf{E}}(x)$ and $\hat{\mathbf{E}}'(x)$ corresponds to 1 by 3 data vectors.

Given the prediction error $e(x)$ and the covariance $\Sigma(e(x))$, the test statistic: $d^2 = e(x)'\Sigma^{-1}(e(x))$ has a χ^2 distribution with 3 degrees of freedom. However, in our situation, we have to use $\hat{\Sigma}$, the estimate of the true unknown covariance matrix evaluated at $\hat{\mathbf{E}}(x)$ and $\hat{\mathbf{E}}'(x)$. Thus, the test statistic calculated, \hat{d}^2 , is only approximately χ^2 . A more closer look points out that \hat{d}^2 is distributed as the sum of two random variables one having a χ^2 distribution with 3 degrees of freedom and another having the distribution of the quadratic form $e(x)'\Delta\Sigma e(x)$. Here $\Delta\Sigma$ corresponds to the error in the covariance matrix estimate. This is verified in our real experiments (Please see figure 6). Thus, it is necessary to have an additional empirical estimation step to estimate the mapping between probability of false alarm and the threshold.

In the above discussion, we mainly considered all pixels in a given surface patch with same illumination effect. In reality, we don't know clustering of the pixels according

to change of reflectance or illumination distribution before hand. This is exactly the problem to be solved. Estimating the transformation matrix in (20) requires such clustering information. To deal with this dilemma we adopted robust estimation algorithms. The basic idea is to iteratively partition the data into clusters through the use of robust techniques that can detect and handle outliers. We use the least trimmed sum of squares(LTS) [13] [19] to give an initial guess of the largest region with same illumination effect. The M-estimator [9] is then used to refine the estimation. When the estimation of M converges, all the pixels obeying this transformation are considered to be within the same cluster. For example, in figure 5(d), all pixels in region A is identified as belonging to the same cluster because they were previously lit by morning skylight (figure 5)(a) and now by sunlight (figure 5(b)). Color changes of the remaining pixels are due to another lighting, the skylight, or parked car. To detect shadows we re-apply the estimation technique to identify the next cluster of pixels among the non-labelled pixels. Shadows of different shade are detected in the second and third run. At each step the test statistic \hat{d}^2 is measured for the cluster pixels and the iterations stop when this test statistic is greater than the chi-square threshold discussed above. In the example, the pixels that are labelled black are considered "change".

The detected changes and shadows are shown in figure (5d). Despite dramatic change in light spectrum and intensity, shadows and parked cars are successfully detected. We applied the algorithm on 21 other HDR images that were taken as much as 10 hours apart from the time the reference images was taken. We get good results for all the cases.

From theory, the mean of the \hat{d}^2 test statistic in the "unchanged" regions should be approximately 3 (assuming zero mean error for $\Delta\Sigma$). We computed this value for the dominant background cluster for the 21 images. The population mean is 3.3792 with the standard deviation of this estimate being 0.5064. For each of the 21 sequences we plotted the distribution of the χ^2 value against the theoretical value in figure 6. The estimated probability density function matches well with the theoretical predictions ¹.

6 Conclusion

In this paper we proposed a statistical method to calibrate the imaging process. We use a statistical model of the measurement errors in a CCD camera that accounts for various transformations in the sensor. We proposed an iterative non-linear estimation algorithm to estimate the camera response function, the white balancing coefficients, and the scene irradiance predictions. We verified the correctness of the al-

¹Theory predicts that there will be some discrepancy between the chi-squared distribution and the observed distribution

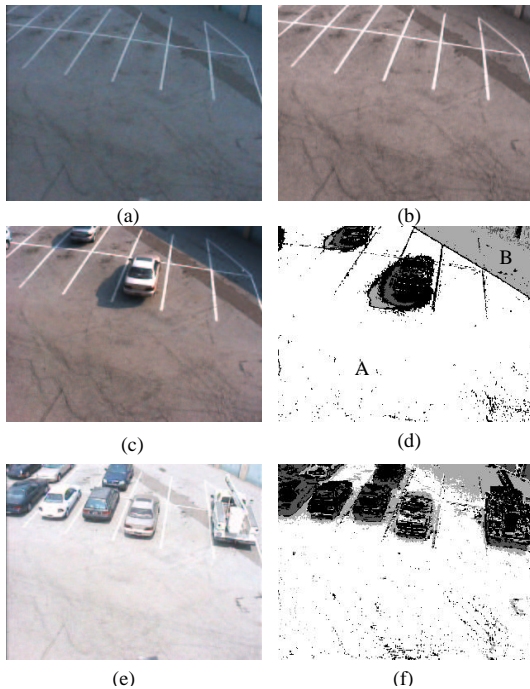


Figure 5: Change detection under varying illumination. (a) Reference image, taken at 7:00. (b) The predicted image: what the background would look like if it is lit by sun at 9:20 of that day. (c) Target image, taken at 9:20 of the same day. (d) Detected changes: light and dark gray: different shades of shadows. Black: change. (e) Another target image, taken at 14:30 of the same day. (f) detected change at 14:30.

gorithm through extensive simulations and by experiments on real data. A major aspect of the proposed method is that estimates of the uncertainties of irradiance estimates are also obtained. These uncertainties can be used to improve the robustness of computer vision methods. We used illumination invariant long-term change detection in outdoor environments as an example to illustrate this aspect of the work. The change detection algorithm here considered only the most simple case: flat background, fixed camera and static scene. However, the method can be extended to treat more general cases. For example, we can combine our algorithm with SSD method [6] to register two images of the same scene under different illumination. Under varying lighting, the registration algorithm can be used for object recognition, memory based robot localization, color constancy.

7 Acknowledgement

We thank Dr. Sathyakama Sandilya for valuable comments that helped improve the paper.

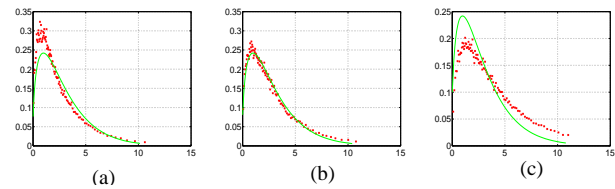


Figure 6: χ^2 distribution: estimation versus theory. (a)-(c) clearly showed that \hat{d}^2 is only approximately distributed as a chi-square distribution with 3 degrees of freedom.

References

- [1] J. E. Adams, "Design of Practical Color Filter Array Interpolation Algorithms for Digital Cameras", *Proceeding of SPIE*, 3028, pp 117-125
- [2] N. Asada, A. Amano and M. Baba, "Photometric Calibration of Zoom Lens Systems", *Proc. of IEEE International Conference on Pattern Recognition*, 1996.
- [3] R.E. Barlow, D. J. Bartholomew, J.M.Bremner and H. D. Brunk, *Statistical Inference under Order Restrictions*. John Wiley & Sons, 1972.
- [4] V. Brajovic, R. Miyagawa, and T. Kanade, "Temporal Photoreception for Adaptive Dynamic Range Image Sensing and Encoding," *Neural Networks*, Vol. 11, No. 7-8, October, 1998, pp. 1149-1158.
- [5] P. E. Debevec, J. Malik, "Recovering High Dynamic Range Radiance Maps from Photographs", *In SIGGRAPH '97*.
- [6] G.D. Hager and P.N. Belhumeur, "Efficient Region Tracking With Parametric Models," *IEEE Trans. on Patt. Analysis and Machine Intelligence*, 20(10):1025-1039, 1998.
- [7] G. E. Healey, R. Kondepudy, "Radiometric CCD Camera Calibration and Noise Estimation", *IEEE Transaction on Pattern Recognition and Machine Intelligence*, Vol. 16, No. 3, March 1994. ;
- [8] B.K.P.Horn, *Robot Vision*. MIT Press, Cambridge. 1986.
- [9] P. J. Huber, *Robust Statistics*. New York, Wiley, 1980.
- [10] L. T. Maloney, "Evaluation of linear models of surface spectral reflectance with small numbers of parameters", *J. Opt. Soc. Am. A*/Vol. 3, No. 10, October 1986.
- [11] L. T. Maloney and B. A. Wandell, "Color Constancy: a method for recovering surface spectral reflectance," *Journal of the Optical Society of America*, A, 3(1):29-33, 1986.
- [12] S. Mann and R. W. Picard, "Being 'Undigital' with digital cameras: Extending Dynamic Range by Combining Differently Exposed Pictures", *IS&T 48th Annual Conference*.
- [13] P. Meer, "Robust Techniques for Computer Vision.", Tutorial at the 1997 *Computer Vision & Pattern Recognition Conference, San Juan, Puerto Rico*.
- [14] T. Mitsunaga, S. Nayar, "Radiometric Self Calibration", *CVPR '99*.
- [15] S. Nayar, K. Ikeuchi, and T. Kanade, "Shape from Interreflections," *Int'l Journal of Computer Vision*, vol. 6, no. 3, 1991, pp. 73-195.
- [16] C. L. Novak, S. A. Shafer, and R. G. Wilson, "Obtaining accurate color images for machine vision research," in *Proc. SPIE Conf. Perceiving, Measuring and Using Color*, Santa Clara, CA, Feb. 1990.
- [17] D. Qian, J. Toker and S. Bencuya, "An Automatic Light Spectrum Compensation Method for CCD White Balance Measurement," *IEEE Trans. Consumer Electronics*, Vol. 43, No. 2 May 1997, pp. 216-220.
- [18] G. S. Rogers, *Matrix Derivatives*, Marcel Dekker, inc, New York, 1980.
- [19] P. J. Rousseeuw, *Robust regression and outlier detection*, Annick M. Leroy, 1987.
- [20] D. Slater and G. Healey, "The Illumination Invariant Matching of Deterministic Local Structure in Color Images," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 10, October 97.