

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/221807692>

Digital camera simulation

Article in *Applied Optics* · February 2012

DOI: 10.1364/AO.51.000A80 · Source: PubMed

CITATIONS

65

READS

4,163

3 authors, including:



Joyce Farrell

Stanford University

61 PUBLICATIONS 1,158 CITATIONS

[SEE PROFILE](#)



Brian Wandell

Stanford University

443 PUBLICATIONS 26,272 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Computational observer analysis of visual performance [View project](#)



Human MRI and Modeling [View project](#)

Digital Camera Simulation

Joyce Farrell, Manu Parmar, Peter Catrysse and Brian Wandell

Introduction

Customers judge the image quality of a digital camera by viewing the final rendered output. Achieving a high quality output depends on the multiple system components, including the optical system, imaging sensor, image processor and display device. Consequently, analyzing components singly, without reference to the characteristics of the other components, provides only a limited view of the system performance. In multi-device systems, a controlled simulation environment can provide the engineer with useful guidance that improves the understanding of the system and guides design considerations for individual parts and algorithms.

In this chapter, we describe a system approach to modeling and simulating the image processing pipeline of a digital camera, beginning with a radiometric description of the scene captured by the camera and ending with a radiometric description of the final image as it appears on an LCD display. The chapter is organized into five sections.

The section on the **Scene** describes how to create a radiometric description of the scene. We represent a scene as a multidimensional array describing the spectral radiance (photons/sec/nm/sr/m²) at each pixel in the sampled scene. We use linear models of surfaces and lights to simplify and compress the representation of spectral radiance emitted from a scene. We also describe several methods for acquiring high dynamic multispectral images of natural scenes to generate radiometric image data (surfaces and illuminants).

The section on **Optics** describes how scene radiance data are converted into an irradiance image at the sensor surface. The conversion from radiance to irradiance is determined by the properties of the optics, which gather the diverging rays from a point in the scene and focus them onto the image sensor [1].

The section on the **Sensor** describes how the irradiance image is converted into electrons. This transformation includes a model of the optical and electrical properties of the sensor and pixel. The image sensor model includes a great many design parameters, only some of which will be discussed here. Among the various factors accounted for are the spatial sampling of the optical image by the image sensor, the size and position of pixels and their fill-factor. The wavelength selectivity of color filters, intervening filters (such as an infrared filter), and the photodetector spectral quantum efficiency are also included in the simulation. After converting photons into electrons, we add sources of noise and quantize the data to generate digital sensor values.

The section on the **Processor** describes how digital values generated by the two-dimensional sensor array are converted into a three-dimensional (RGB) image that can be rendered on a specified display. This is accomplished by interpolating missing RGB sensor values

(demosaicking) and transforming sensor RGB values into an internal color space for encoding and display (color-balancing and display rendering). In this chapter, we describe the general principles of auto-exposure, demosaicing and color-balancing. Other chapters describe algorithms for these processes in more detail.

The section on **Displays** describes how to generate a radiometric description of the final image as it appears on an LCD display. The spectral radiance for any image rendered on a typical display can be predicted by three functions – the display gamma, the spatial point spread functions for the red, green and blue pixel components and the spectral power distributions of the display color primaries [2]. Modeling the display makes it possible to use the radiance field as the input to objective image quality metrics. Calculating the spatial-spectral radiance is useful because, unlike the digital image values, the radiance field is the stimulus that actually reaches the eye.

In the final section of this chapter, we use an integrated suite of Matlab® software tools to explore the full imaging pathway, including scene, optics, and sensor of a calibrated digital camera. We illustrate how it is possible to use simulation tools to characterize imaging sensors and explore novel designs.

Many of the ideas we describe in this chapter are an extension of work that we and our colleagues have described in previous publications. For example, Chen et al [3] developed a grayscale digital camera simulator to evaluate the effects of pixel size on dynamic range, signal-to-noise ratio and camera modulation transfer functions. Vora et al [4] and Longere and Brainard [5] developed a color digital camera simulator that combined hyperspectral scene data with a linear model of imaging sensors. They demonstrated the viability of their simulator by comparing simulated and measured sensor performance. And Longere and Brainard [5] used their digital camera simulator to evaluate the effect that sensor spectral sensitivity, noise and color filter array patterns have upon image quality. We recommend these previous publications as an introduction to the ideas developed more completely in this chapter.

Scene

Digital camera simulation requires a physically accurate description of the light incident on the imaging sensor. We represent a scene as a multidimensional array describing the spectral radiance (photons/sec/nm/sr/m²) at each pixel in the sampled scene. The spectral radiance image data are assumed to arise from a single image plane at a specified distance from the optics.

There are several different sources of scene data. For example, there are synthetic scenes, such as the Macbeth ColorChecker, spatial frequency sweep patterns, intensity ramps and uniform fields. When used in combination with image quality metrics, these synthetic target scenes are useful for evaluating specific features of the system, such as color accuracy, spatial resolution, intensity quantization, and noise.

Another important source of scene data is calibrated representations of natural scenes. In the section below, we describe several methods for acquiring high dynamic multispectral images of natural scenes which can be used to generate radiometric image data.

Finally, scene radiance image data can be estimated from RGB data. In the section on Displays, we show that it is possible to use linear models to predict the spectral radiance of a displayed image.

High Dynamic Range Multispectral Scene Data (HDRS)

We describe several methods for creating high dynamic range spectral (HDRS) radiance scene data using calibrated cameras and a variety of different filters and light sources.

We extend the dynamic range of a single image exposure by combining unprocessed or so-called “raw” camera images across multiple exposures [2, 4-7]. If we exclude specular highlights, the dynamic range of the vast majority of natural scenes is on the order of 5000:1 [8]. Hence, a capture system based on a 12-bit ADC that uses a few exposure durations to bracket the range can capture the full dynamic range of most natural scenes [7-10].

We can also extend the spectral information in the image data by combining raw camera images from more than three color channels. Additional color channels can be created by interposing different color filters between successive images. The color filters can either be placed between the light and the scene (creating different colored lights) [11, 12] or between the scene and the camera (adjusting the spectral sensitivity of the camera) [13-15]. In both cases we can use linear models of surfaces and illuminants, briefly described below and more thoroughly in other publications [16-23] to estimate the spectral radiance.

Linear models of surfaces and illuminants

Data gathered over the last fifteen years suggests that for scenes containing only one or two light sources, the dimensionality of the scene spectral radiance functions will be captured accurately (within 5% rmse) using as few as five or six basis functions for a particular illumination condition [23-26]. This number is much smaller than the 61 values necessary to represent spectral radiance measurements from 400 to 700nm, in 5 nm steps, but it is greater than the three values captured with a traditional RGB digital camera.

As noted earlier, there are several ways to increase the number of spectral samples. In the example described here, we use a three-color camera in combination with three filters. One of the captures is made with no color filter, and two are made with color filters placed between the camera and the scene. This procedure yields nine different spectral samples. While these samples are not independent, they do provide enough independent information to obtain a reasonable spectral approximation to many scenes.

The spectral radiance of a surface measures the rate of photons emitted (or scattered) from a surface as a function of area, angular direction, and wavelength. In most cases, such as for a

simple reflective surface, the spectral radiance (in units of photons/sec/nm/m²) can be represented by a vector, s . If we have measurements from 400 to 700nm, in 5 nm steps, then the vector s will have 61 entries. This representation requires a lot of memory for images of any significant spatial size. For example, a 512 x 512 image with 16-bits per wavelength plane is stored in about 30 MB. A 1Kx1K image is stored in about 120 MB of data. It is possible, however, to use low-dimensional linear models to summarize efficiently the scene spectral radiance data. This data compression takes advantage of the fact that spectral radiance functions in most natural images are regular functions of wavelength.

The spectral representation of the light reflected from each point a scene can be approximated by a linear combination of a small set of spectral basis functions, B_i , where w_i are the weights chosen to minimize the error between the spectral radiance and its linear model approximation, and N is the dimensionality of the spectral representation.¹

$$s \approx \sum_{i=1}^N w_i B_i \quad (1.1)$$

The low-dimensional linear models of spectral radiance reduce the amount of data that must be stored when the number of basis functions, N , is less than 61. The accuracy of the low-dimensional spectral representations are limited by 1) how well the spectral radiance data conform to a low dimensional linear model, 2) the dimensionality of the image capture system which may or may not have enough sensors to capture the scene information, and 3) the quality of the algorithms for inferring the spectral radiance functions from the sensor data.

Acquisition methods and validation

To illustrate high dynamic range multispectral imaging, we used a Nikon D100 digital SLR camera with two carefully chosen filters to record high dynamic range multispectral measurements of natural scenes. Because of its low noise and dark current, the Nikon D100 is close to a true 12-bit camera. We used the Nikon D100's exposure auto-bracketing function to take three images in rapid sequence at exposure settings each separated by a factor of four. The shortest exposure duration was based on photometric measurements of the maximum luminance.

For each scene, we captured three exposure auto-bracketing pictures each with and without the two color filters in the imaging path. This resulted in a total of 9 images (3 exposure settings x 3 color filter conditions) for each scene. To reduce image registration problems, the aperture setting (f-stop) was fixed for all 9 images. The camera was positioned on a tripod for stability

¹The complete characterization of the spectral radiance factor of a surface that has both diffuse and fluorescent components requires the full $N \times N$ matrix, S , where N is the number of wavelength samples represented in the matrix S . If the surface is diffuse and does not fluorescence, S has values between 0 and 1 along the diagonal and zeroes in the off-diagonal positions in the matrix. Hence, a vector, s , corresponding to the diagonal component of S , can represent the spectral radiance of a diffuse surface.

across the capture. The images were spatially registered when necessary, though very little camera motion was present. Stable scenes, without significant image motion, were selected.

The Nikon sensor contains a Bayer mosaic with two green, one red and one blue photodetector type in a 1012x1517 grid. We down-sampled the Nikon sensor images to obtain rgb images with a spatial resolution of 253x380. (We use higher resolution digital cameras and various sampling rates to obtain rgb images with higher resolution.)

To estimate the intensity recorded by each pixel, we used the last sample before saturation (LSBS) algorithm [6]. This sample has the highest signal-to-noise ratio. From this estimate, we created a single high dynamic range image with 9 color channels.

We use the 9-color high dynamic range images captured for each scene to estimate a spectral radiance image for that scene. Spectral power distributions of many natural images can be modeled as a linear combination of a relatively small (~ 5) set of spectral basis functions [24-26].

To improve the estimation of the spectral radiance of the image, we acquired a measurement of the spectral power distribution of the scene illumination. Using this measurement, we could simulate the light scattered from a standard test chart (the Macbeth Color Checker) and, using the singular value decomposition, derive three scene basis functions for that illumination. We added two spectral basis functions to these, a DC and a linear ramp function (see Figures 1, 2 and 3, below). These functions are not orthogonal, but they are independent. Through empirical evaluations, described below, we found that these basis functions provide a high quality representation of the image data. Each HDRS image, then, includes a set of spectral bases, B_i , along with the basis coefficients (weights), w_i . The basis coefficients, w_i , are calculated from the nine color channels recorded by the multicapture imaging system.

We estimate the basis coefficients in the following way. We represent the wavelength information at 61 samples, from 400 to 700 nm in 5 nm steps. Let r be a 9x1 vector representing the output of nine color channels for a single pixel a sampled scene. Let T be a 9 x 61 matrix representing the spectral responsivities of the nine color channels in the image capture device. Let B be a 61 x 5 matrix representing scene spectral basis functions. Finally, let w be a 5x1 vector representing the weights on B . Then,

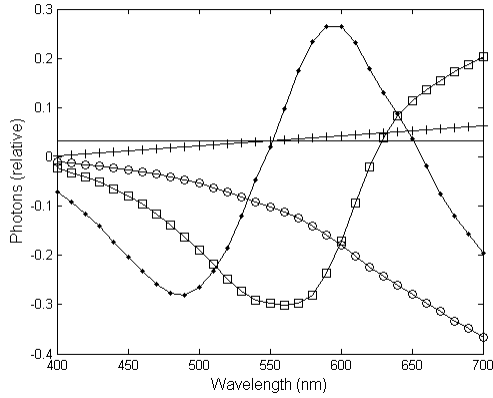
$$r = TBw \quad (1.2)$$

Solve for w using the pseudoinverse of TB ,

$$w = [TB]^{-1} r \quad (1.3)$$

In this example, we used 5 spectral basis functions in the linear model approximations for the spectral radiance measured from the 24 color patches in the Macbeth ColorChecker under tungsten and fluorescent illumination (Figure 1). Two spectral basis functions were a constant (DC) and ramp signal. The remaining 3 spectral basis functions correspond to the 3 most significant eigenvectors derived from the singular value decomposition (SVD) of the spectral radiance measured from the 24 surfaces illuminated by tungsten or fluorescent lights.

A.



B.

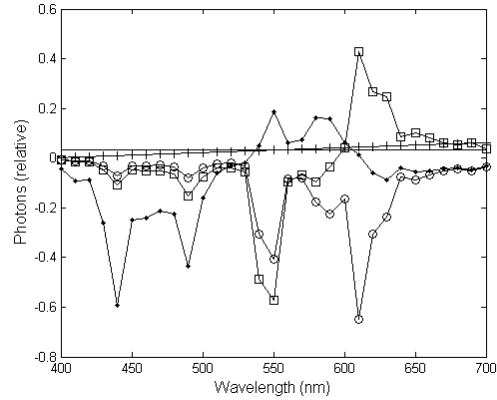


Figure 1. Spectral basis functions for diffusely reflecting surfaces when surfaces are illuminated by tungsten(A) and fluorescent (B) illumination.

Figures 2 and 3 compare the estimated and measured spectral radiance of 24 color patches in the Macbeth Color Checker illuminated by tungsten (Figure 2) and fluorescent (Figure 3). The rmse between the estimated and measured data is less than 3.5 %, though in some parts of the spectrum, particularly above 650 nm, the Nikon D100 has poor wavelength sensitivity and the errors are larger. In subsequent work, we used a modified Nikon D200 digital camera that can capture energy in both the visible and near infrared regions of the spectrum [27].

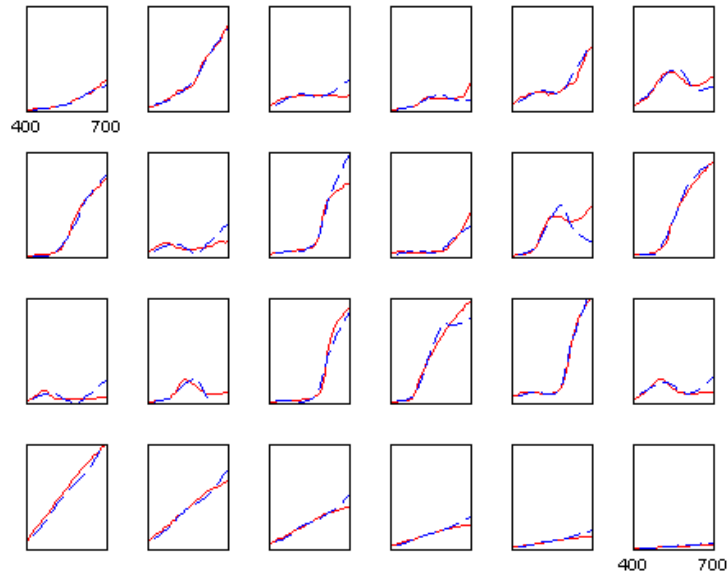


Figure 2. Measured (solid line) and estimated spectral radiance of the twenty-four surfaces of a Macbeth ColorChecker illuminated by a tungsten light. The estimation was made using the five linear model basis functions shown in Figure 1a and data from the Nikon D100, as described in the text.

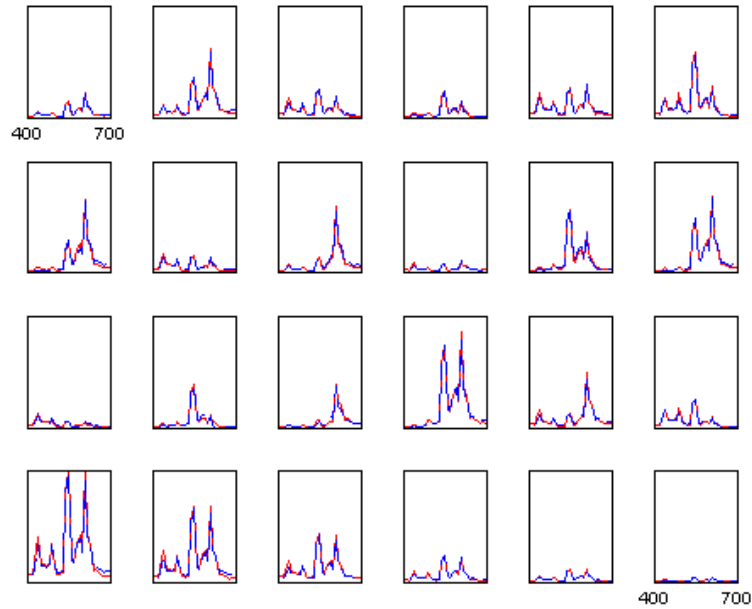


Figure 3. Measured (solid line) and estimated scene radiance of the twenty-four surfaces of a Macbeth ColorChecker illuminated by a fluorescent light. The estimation was made using the five linear model basis functions shown in Figure 1b and data from the Nikon D100, as described in the text.

Other methods for acquiring multispectral image data

In the example described above, we created additional color channels by placing color filters directly on the camera, effectively adjusting the spectral sensitivity of the camera. It is also possible to place the color filters directly in front of a light, effectively changing the spectral composition of the scene illuminant.

Many researchers have used methods based on optical filters to acquire multispectral images of artwork [9-12], scenes with IR energy [27] and natural scenes [28]. Since a number of acquisitions are required with different optical filters, this method is best suited for imaging static scenes. Long exposure times preclude the use of this method for acquiring scenes that include people and other dynamic objects. The time required to change filters may be reduced by the use of filters mounted on a color wheel, but this renders the device physically cumbersome and expensive.

An alternate approach is to use multiple acquisitions of a scene illuminated by lights with different spectral power distributions. Since lighting systems may be designed to switch at high rates, this approach can be used to acquire multispectral scenes with people and other animate objects. We have created a system that uses multiple LEDs for capturing multispectral scenes[29]. The spectral data may be within the visible or extend into the infrared[27].

Spectral Radiance Image Representation

The spectral radiance image can be stored in a compact wavelength representation as a data file that contains a representation of the spectral basis functions and coefficients. When the scene illumination is known, it can be stored in the file as well. Hence, a relatively small (four, five or six) chromatic samples – along with the modest overhead of including the basis functions - represent the full spectral radiance information. It is further possible to compress the spatial images of the coefficients, of course.

Figure 4 shows an image derived from a multispectral representation of a female face, and the spectral reflectance data for a selected region of her face. In order to illustrate the multispectral image, we summed the energy in long-, middle- and short-visible wavebands and assigned these to the R, G and B primaries, respectively. The graph in Figure 4 compares the measured and estimated spectral reflectance of a region selected from the subject's forehead.

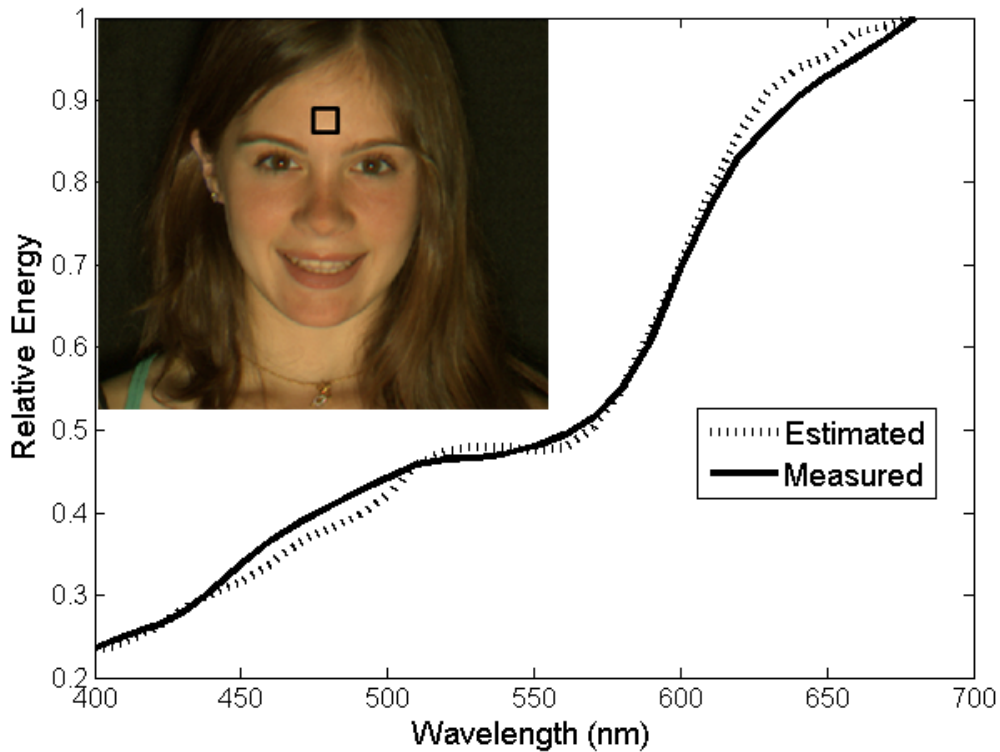


Figure 4. The spectral image of a female face is rendered by summing the energy in long-, middle- and short-visible wavebands and assigning these to the R, G and B primaries, respectively. The graph compares the measured and estimated spectral reflectance of a region selected from her forehead.

In the current implementation, image data are modeled as arising from a plane. Extending the image representation to include depth information would allow the simulation to account for several additional factors, such as the optical depth of field. Also, the image radiance is assumed to radiate equally in all directions, while in a true natural scenes the scattering depends on the geometry of the light surface, surface normals, and material specularities. Consequently, while the current spectral radiance data represent one particular view, they are not easily generalized to other camera viewpoints. To lift these limitations, it is possible to capture multispectral images of objects in a scene from multiple vantage points. For example, one can use multiple cameras in different locations [30] or rotate the objects in the scene [31] into different positions. It would then be possible to render different instances of the scene, captured from different viewpoints.

Optics and sensor irradiance

In the scene radiance, photons are emitted in multiple directions by light sources or scattered in multiple directions. The lens gathers up these divergent rays so that they converge to an irradiance image at the sensor surface. We call the irradiance image at the sensor surface, just prior to capture, the optical irradiance. To compute the optical irradiance image, we must account for a number of factors. First, we account for the lens f-number and magnification.

Second, we account for lens shading (relative illumination), the fall-off in intensity with lens field height. Third, we blur the optical irradiance image. The blurring can be performed with one of three models: a wavelength-dependent shift-invariant diffraction-limited model, a wavelength-dependent general shift-invariant model (arbitrary point spread), and a general ray trace calculation which further incorporates geometric distortions and a wavelength-dependent blur that is not shift-invariant. The reader can find a thorough description of imaging optics in Chapter 2 of this book [1].

Converting radiance to irradiance

The camera equation [32] defines a simple model for converting the scene radiance function, L_{scene} , to the optical irradiance field at the sensor, I . The camera equation is

$$I_{image}(x, y, \lambda) \cong \frac{\pi T(\lambda)}{4(f/\#)^2} L_{scene}\left(\frac{x}{m}, \frac{y}{m}, \lambda\right)$$

The term $f/\#$ is the effective f-number of the lens (focal length divided by effective aperture); m is the lens magnification; and $T(\lambda)$ is the transmissivity of the lens. The camera equation holds with good precision for the center of the image (i.e., on the optical axis). For all the other locations in the optical image, we need to apply an off-axis correction.

Relative illumination

The fall-off in illumination from the principal axis is called the relative illumination or lens-shading, $R(x, y, \lambda)$. There is a simple formula to describe the shading in the case of a thin lens without vignetting or (geometric) distortion. In that case, the fall-off is proportional to $\cos^4 \theta$, where θ is the off-axis angle

$$R(x, y, \lambda) = \cos^4 \theta \approx \left(\frac{d}{S}\right)^4$$

The term S is the image field height (distance from on-axis) and d is the distance from the lens to the image plane. The (x, y) coordinates specify the position with respect to the center of the image axis. This formula is often called the Cosine-fourth Law. In real lenses, or lens-collections, the actual off-axis correction may differ from this function. It is often used, however, as a good guess for the irradiance decline as we measure off-axis.

Irradiance image blurring

The irradiance image, $I_{image}(x, y, \lambda)$ cannot be a perfect replica of the scene radiance, $L_{scene}(x, y, \lambda)$. Imperfections in the lens material or shape, as well as fundamental physical limitations (diffraction), limit the precision of the reproduction. The imperfections caused by these factors can be modeled by several types of blurring calculations of increasing complexity.

Diffraction-limited optics

When a wave encounters a small obstacle, the wave's path is changed. Similarly, as waves pass through a small aperture their direction of travel is influenced by the aperture. This phenomenon is known as diffraction. The effect of the aperture on the passage of the light waves imposes a fundamental limit on the possibility of forming an accurate image of the scene using conventional optics.

A diffraction-limited system can be modeled as having a wavelength-dependent, shift-invariant point spread function [33, 34]. Diffraction-limited modeling uses a wave-optics approach to compute the blurring caused by a perfect lens with a finite aperture. The point spread function of a diffraction-limited lens is quite simple, depending only on the $f/\#$ of the lens and the wavelength of the light. It is particularly simple to express the formula in terms of the Fourier Transform of the point spread function, which is also called the optical transfer function (OTF).

The formula for the diffraction-limited OTF is

$$OTF = \begin{cases} \frac{2}{\pi} a \cos(\rho) - (\rho \sqrt{1 + \rho^2}), & \rho < 1 \\ 0, & \rho \geq 1 \end{cases} \quad (1.4)$$

Where $\rho = f(A/(\lambda d))$ (normalized frequency)

f = frequency in cycles/meter

A = aperture diameter in meters

λ = wavelength

d = distance between the lens aperture and detector

Shift-invariant modeling

The diffraction-limited point spread function is a specific instance of a shift-invariant linear model. It is natural to generalize this model by allowing the point spread function to vary. For example, rather than having a point spread defined by physical bounds the user may define a wavelength-dependent point spread function. There are various reasons to model such blur in a typical camera path; for example, many cameras include a blurring glass on the sensor surface that includes protection from infrared rays and reduces aliasing from detector sampling.

In optics, the term isoplanatic is used to define conditions when a shift-invariant model is appropriate. Specifically, an isoplanatic patch in an optical system is a region in which the aberrations are constant; experimentally, a patch is isoplanatic if translation of a point in the object plane causes no change in the irradiance distribution of the PSF except its location in the image plane. This is precisely the idea behind a shift-invariant linear system.

The lens transformation from a shift-invariant system can be computed much more efficiently than a shift-variant (anisoplanatic) system. The computational efficiency arises because the computation can take advantage of the Fast Fourier Transform to calculate the spatial distribution of the irradiance image.

Specifically, we can convert the image formation and point spread function into the spatial frequency domain. The shift-invariant convolution in the space domain is a point-wise product in the spatial frequency domain. Hence, we have

$$FT\{I_{image}(x, y, \lambda)\} = FT\{PSF(x, y, \lambda)\} \cdot FT\{I_{ideal}(x, y, \lambda)\} \quad (1.5)$$

$$FT\{I_{image}(x, y, \lambda)\} = OTF(f_x, f_y, \lambda) \cdot FT\{I_{ideal}(x, y, \lambda)\} \quad (1.6)$$

$$I_{image}(x, y, \lambda) = FT^{-1}\{OTF \cdot FT\{I_{ideal}(x, y, \lambda)\}\} \quad (1.7)$$

where $FT\{\}$ is the Fourier transform operator and the optical transfer function, OTF, is the Fourier transform of the PSF. Because no photons are lost or added, the area under the PSF is one, or equivalently, $OTF(0,0,\lambda) = 1$. In this shift-invariant model, we assume that the point spread is shift-invariant for each wavelength, but the point spread function may differ across wavelengths. Such differences are common because of factors such as longitudinal chromatic aberrations.

Ray Trace Modeling

The ray-trace method model replaces the single, shift-invariant, point spread function with a series of wavelength-dependent point spread functions that vary as a function of field height and angle. These functions can be specified by the user, or more likely they can be calculated using lens design software. Ray trace programs also specify the geometric distortion from the lens. This takes the form of a displacement field that varies as a function of input image position ($d(x,y)$). The displacement field is applied first, and then the result is blurred by the local point spread function, wavelength-by-wavelength.

Sensor

Image sensors transform the optical irradiance image into a two-dimensional array of voltage samples, one sample from each pixel. Each sample is associated with a position in the image space. Most commonly, the pixel positions are arranged to form a regular, two-dimensional sampling array. This array matches the spatial sampling grids of common output devices, including displays and printers.

In most digital image sensors the transduction of photons to electrons is linear: specifically, the photodetector response (either CCD or CMOS) increases linearly with the number incident photons. Depending on the material properties of the silicon substrate, such as its thickness, the

photodetector wavelength sensitivity will vary. But even so, the response is linear in that the detector sums the responses across wavelengths. Hence, ignoring device imperfections and noise, the mean response of the photodetector to an irradiance image ($I(\lambda, x)$, photons/sec/nm/m²) is determined by the sensor spectral quantum efficiency ($S(\lambda)$, e⁻/photon), aperture function across space $A_i(x)$, and exposure time (T, sec). For the i^{th} photodetector, the number of electrons will be summed across the aperture and wavelength range

$$e_i = T \iint_{\lambda, x} S_i(\lambda) A_i(x) I(\lambda, x) d\lambda dx \quad (1.8)$$

A complete sensor simulation must account for the device imperfections and noise sources. Hence, the full simulation is more complex than the linear expression in Equation 1.7. Here, we outline the factors and computational steps that must be incorporated in a simulation.

Computing the signal current density image. One approach to simulation is to first convert the sampled irradiance image, whose values represent the irradiance in units of quanta per sec per square meter, to a signal current density image (amps/m²). This image represents what we would measure if the surface of the sensor were one continuous photodetector.

The irradiance image already includes the effects of the imaging optics. To compute the signal current density we must further specify the effect of several additional optical factors within the sensor and pixel. For example, most consumer cameras include an infrared filter that covers the entire sensor. This filter is present because the human eye is not sensitive to infrared wavelengths, while the detector is. For consumer-imaging the sensor is designed to capture the spectral components of the image that the eye sees – and to exclude those parts that the eye fails to see. The infrared filter helps to accomplish this goal, and thus it covers all of the pixels.

We must also account for the color filters placed in front of the individual pixels. While the photodetectors in the array are typically the same, each is provided with a color filter that permits certain wavelengths of light to pass and rejects others.

The geometric structure of a pixel, which is something like a tunnel, also has a significant impact on the signal current density image. The position and width of the opening to the tunnel determine the pixel aperture. Ordinarily the photodetector is at the bottom of the tunnel in the silicon substrate. In modern CMOS imagers, usually built using multiple metal layers, the pixel depth can be as large as the pixel aperture. Imagine a photon that must enter through the aperture and arrive safely at the photodetector at the bottom. If the pixel is at the edge of the sensor array, the photon's direction as it travels from the center of the imaging lens must be significantly altered. This re-direction is accomplished by a microlens, positioned near the aperture. The position of each microlens with respect to the pixel center varies across the array because the optimal placement of the microlens depends on the pixel position with respect to the imaging lens.

As the photon travels from the aperture to the photodetector, the photon must pass through a series of materials. Each of these has its own refractive index and thus can scatter the light or change its direction. The optical efficiency of each pixel depends on these materials.

Space-time integration. After accounting for the photodetector spectral quantum efficiency, the various filters, the microlens array, and pixel vignetting, we can compute the expected current per unit area at the sensor. This signal current density image is represented at the same spatial sampling density as the optical irradiance image.

The next logical step is to account for the size, position and exposure duration of each of the photodetectors by integrating the current across space and time. In this stage, we must coordinate the spatial representation of the optical image sample points and the pixel positions. Once these two images are represented in the same spatial coordinate frame, we can integrate the signal across the spatial dimensions of each pixel. We also integrate across the exposure duration to calculate the electrons accumulated at each pixel.

Incorporating sensor noise. At this stage of the process, we have a spatial array of pixel electrons. The values are noise-free. In the third step we account for various sources of noise, including the photon shot noise, electrical noise at the pixel, and inhomogeneities across the sensor.

Photon shot noise refers to the random (Poisson) fluctuation in the number of electrons captured within the pixel even in response to a nominally identical light stimulus. This noise is an inescapable property of all imaging systems. Poisson noise is characterized by a single rate parameter that is equal both the mean level and the variance of the distribution.

There are a variety of electrical imperfections in the pixels and the sensor. Dark voltage refers to the accumulation of charge (electrons) even in the absence of light. Dark voltage is often referred to as thermally generated noise because the noise increases with ambient temperature. The process of reading the electrons accumulated within the pixel is noisy, and this is called read noise. Resetting the pixel by emptying its electrons is an imperfect process, and this noise is called reset noise. Finally, the detector only captures a fraction of the incident photons in part because of the material properties and in part because the photodetector only occupies a portion of the surface at the bottom of the pixel. The spectral quantum efficiency is a wavelength-dependent function that describes the fraction of the photons that produce an electron. The fill-factor is the percentage of the pixel that is occupied by the photodetector.

Finally, there is the inevitable variance in the electrical linear response function of the pixels. One variation is in the slope of the response to increasing light intensity; this differs across the array and is called photodetector response nonuniformity (PRNU). Second, the offset of the linear function differs across the array, and this variance in the offset is called dark signal nonuniformity (DSNU). PRNU and DSNU are a type of fixed pattern noise – another source of FPN is due to variation in column amplifiers

Over the years, circuit design has improved greatly and very low noise levels can be achieved. Also, various simple acquisition algorithms can reduce sensor noise. An important example is correlated double sampling. In this method the read process includes two measurements – a reference measurement and a data measurement. The reference measurement includes certain types of noise (reset noise, FPN). By subtracting the two measurements, one can eliminate or

reduce these noises. Correlated double sampling does not remove and may even increase, other types of noise (e.g., shot noise or PRNU variations) [35, 36]

Analog to digital conversion. In the fourth step, we convert the current to a voltage at each pixel. In this process we use the conversion gain and we also account for upper limit imposed by the voltage swing. The maximum deviation from the baseline voltage is called the voltage swing. The maximum number of electrons that can be stored in a pixel is called the well-capacity. The relationship between the number electrons and the voltage is called conversion gain (volts/e⁻).

In many cases, the output voltage is further scaled by an analog gain factor; this too can be specified in the simulation. Finally, the voltages are quantized into digital values.

Processor

Next, we review the general principles of auto-exposure, demosaicking and color-balancing. Other chapters cover these and other topics (e.g. noise removal and sharpening) in more detail.

Auto-Exposure

The most important aspect of the exposure duration is to guarantee that the acquired image falls in a good region of the sensor's sensitivity range. In many devices, the selected exposure value is the main processing step for adjusting the overall image intensity that the consumer will see. In these types of applications, the exposure duration must not only guarantee that the image falls within the sensor range, but it is also part of the image rendering pipeline that should guarantee a pleasing image. This is particularly true for imaging products in which the acquired data are rendered almost immediately, say on a mobile phone. Hence, auto-exposure algorithms are a key element of the image-rendering pipeline that influence image quality.

There is a very small academic literature on auto-exposure algorithms[37-39], although there is a large patent literature [40-42]. Many of the first digital cameras used a separate metering system to set exposure duration, rather than using data acquired from the sensor chip. More recently, the exposure-metering function has been integrated onto the main imaging sensor (usually called through-the-lens, or TTL, metering).

Methods for auto-exposure in digital cameras are based on methods first developed for film cameras. Experts and photographers in film photography use a measure called *exposure value* (EV) to specify the relationship between the f-number, F , and exposure duration, T :

$$EV = \log_2 \left(\frac{F^2}{T} \right) = 2 \log_2(F) - \log_2(T) \quad (1.9)$$

The exposure value becomes smaller as the exposure duration increases, and it becomes larger as the f-number grows.

Most auto exposure algorithms work in the following way. First, take a picture with a pre-determined exposure value (EV_{pre}). Then, calculate a single statistic from the brightness image,

B_{pre} , such as the center-weighted mean in the average of the red and green channels. Assume that we know what the ideal brightness statistic, B_{opt} , should be. This value is typically selected empirically. The optimum exposure value, EV_{opt} , is then defined as

$$EV_{opt} = EV_{pre} + \log_2(B_{pre}) - \log_2(B_{opt}) \quad (1.10)$$

Auto-exposure algorithms differ in how they derive the single number B_{pre} from the picture. For example, B_{pre} might correspond to the mean brightness across the whole image or a center-weighted mean that puts more weight on the center part than the surrounding area. And, of course, there are many variants of these methods. One could average multiple areas in an image, calculate the median instead of the mean brightness, and analyze the green channel only. Notice that the methods we described assume that only exposure duration varies. This is a valid assumption for digital cameras that have a fixed aperture.

Demosaicking

Typically, each pixel in the sensor image is red, green OR blue. To display an image, each pixel must have a red, green AND blue value. We create the display image from the sensor pixel mosaic by interpolating the missing values. This interpolation process is called “demosaicking”.

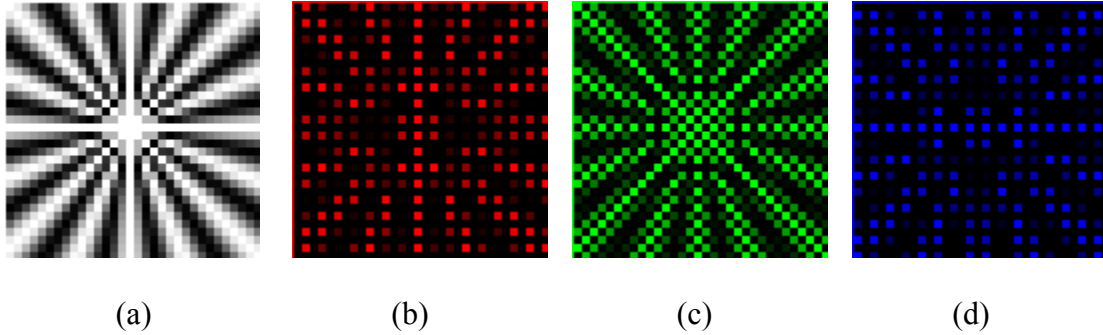


Figure 5. An illustration of color filter array (CFA) sampling. Each pixel captures information about only one color band. (a) A cropped image from a Mackay ray chart, (b-d) The red, green, and blue CFA samples, respectively, from a Bayer CFA.

Demosaicking has attracted interest from a wide range of people in signal processing and applied mathematics. The reader can find an accessible survey in a recent article [43]. Demosaicking algorithms draw on a diverse array of signal processing techniques, e.g., inverse problems [44], neural networks [45], Wavelets [46], Bayesian statistics [47, 48], convex optimization [49] etc.

Here we present a brief overview of several approaches to demosaicking. We classify these algorithms into a few categories that loosely follow the progression of algorithm development in the field.

(a) *Within channel interpolation*

The first group of algorithms interpolates the color planes separately. That is, only the data from, say, the red pixels are used to interpolate the red values at the missing locations. The only prior information that can be used in this approach is the smoothness of each color channel. This

is a major disadvantage since some very useful information about inter-channel correlation, which is critical to good demosaicing performance, is neglected. Methods such as bilinear interpolation, channel-wise spline interpolation, etc. fall in this category.

Figure 6 shows results of interpolating each channel separately using two within channel interpolation methods. Figure 6b shows a nearest neighbor interpolation and Figure 6c shows a bilinear interpolation. In this case the reconstruction produces visible color fringing that is not present in the original image. These methods are simple but as demosaicing algorithms developed the limitations of within channel interpolation quickly became apparent.

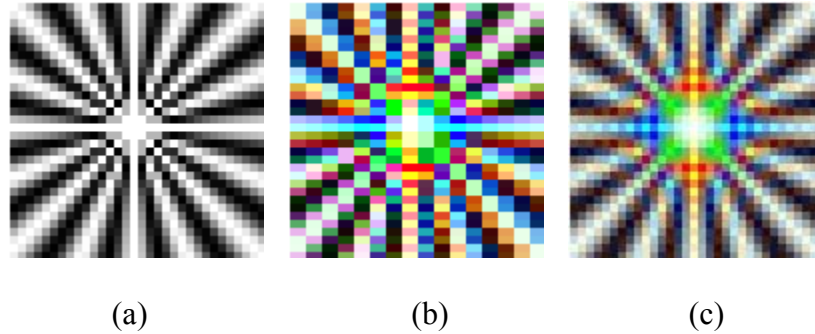


Figure 6. Comparison of within channel interpolation methods. (a) Original image, and channel-wise demosaicked images using (b) the nearest neighbor method, and (c) bilinear interpolation.

(b) *Combined channel interpolation*

A second generation of algorithms uses data from all color planes to interpolate each missing value. There is a wide range of principles that have been offered to guide this interpolation.

One key observation, common to these algorithms, is that the response at a pixel is correlated with other nearby pixels even when they are in different color planes. The reason for this correlation can be understood from first principles. The value of a pixel response is the sum over wavelength of the local irradiance weighted by the pixel's spectral quantum efficiency (Section 'Sensor', Eq. 1.7). The irradiance signal is typically a smoothly varying function of wavelength. Further, in most cameras, the spectral bandwidths of color filters overlap. Consequently, the neighboring pixel values, even those in different color channels, are themselves correlated.

Even more correlated than the mean pixel levels are the local differences in pixel levels. This spatio-chromatic correlation is easily revealed by simple experiments with pictures. We separated an image into its R, G, and B bands. Within each band, we calculated the second difference in the horizontal and vertical directions. These local differences measure the horizontal and vertical high frequency components. The scatter plots in Figure 7bc show that the local differences in the R-channel are highly correlated with the local differences in the G-channel. This correlation is present for both the horizontal and vertical directions. In our experience, the correlation coefficients are generally higher than 0.95 [46].

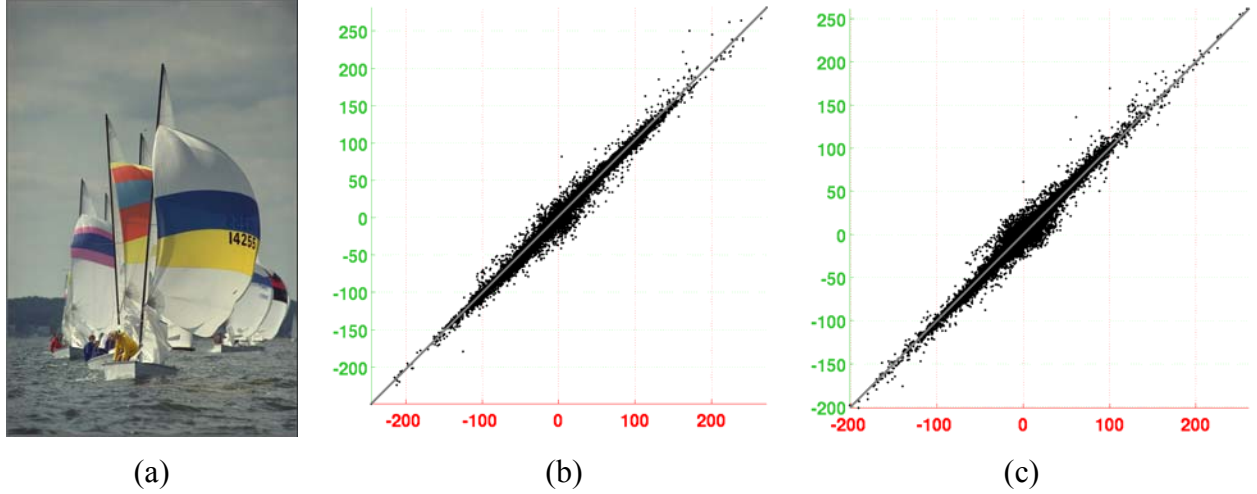


Figure 7. Scatter diagrams of color channels for image 9 from the dataset in Kodak’s PhotoCD PCD0092. Axes colors correspond to channel colors. (a) RGB color image (b) horizontal high frequency component of the red channel vs. the similar component of the green channel, and (c) vertical high frequency component of the red channel vs. the similar component of the green channel.

The correlation of the mean and of the local differences has been used to motivate many demosaicing algorithms [50-54]. Practical implementations of demosaicing that exploit high inter-channel correlations rely on the higher rate of sampling in the green CFA channel in the ubiquitous Bayer array. In general terms, most algorithms derive the best estimate of high frequency image information using the green channel. They then use the inter-channel correlation to improve the interpolation in the red and blue channels.

A second key observation common to demosaicing algorithms concerns identifying locations in an image where the correlations break down. These failures typically happen at object edges. Edges can be approximately identified by finding locations where there are larger differences between adjacent pixel levels. Many algorithms proceed by identifying these locations and interpolating along the edge, but preventing interpolation across the edge. Such algorithms are often called adaptive because the interpolation method adapts to the local image properties. We illustrate three adaptive algorithms in Figure 8.

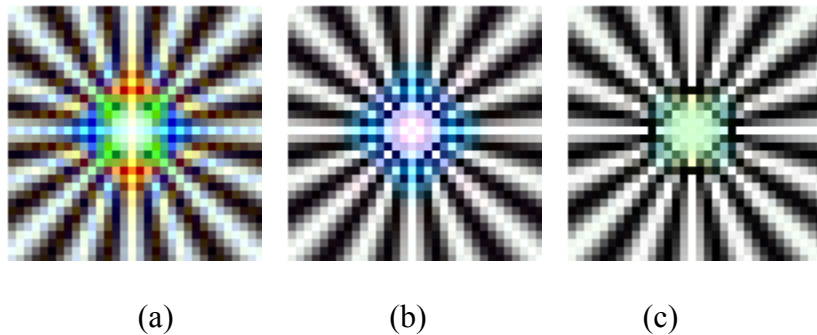


Figure 8. The Mackay image in Fig 1a demosaicked using (a) adaptive Laplacian (b) projections on convex sets [49] , and (c) adaptive homogeneity [50]

(c) Combined color and space interpolation

The field of demosaicking continues to produce new and innovative ideas. An interesting idea suggested by Alleysson and colleagues [55] interprets the CFA-based acquisition in the spatial frequency domain.

Consider a frequency representation of the CFA without regard to the color of individual pixels. In the transform domain, different spatial frequency terms capture information about different color combinations. Consider the most trivial case of a one Bayer super-pixel, (G, R; B, G). The difference between the left column (G + R) and the right column (B + G) is referred to as the 1 cycle vertical term. This difference is B – R because the two G terms cancel. Hence, the amplitude of the CFA transform coefficient at 1 cycle per super-pixel represents a measurement difference between the B and R samples.

There are additional complexities when we consider the full array. Alleysson and colleagues explain how to extract color information from the Fourier transform of the full color filter array of a Bayer pattern. Dubois [56] proposed an adaptive extension of this framework. Hirakawa [57] further extended the framework to design novel CFA patterns optimized for this computational approach. The results of demosaicking the Mackay ray image in Fig. 1a using Dubois's method [56] based on these ideas is shown in Fig. 9.

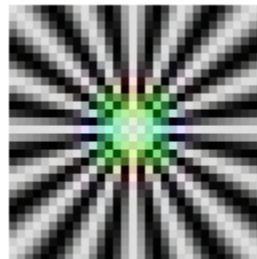


Figure 9. The Mackay image in Fig 1a demosaicked using Dubois's method [56].

The choice of demosaicking algorithm can have a significant impact on the final image quality. It is not possible, however, to assess demosaicking in the absence of knowledge about other imaging components. For example, in a very low noise system simple adaptive algorithms may perform well. In a high noise system, it may be better to use non-adaptive interpolation methods. Further, it is necessary to consider the interaction between demosaicking and other image processing algorithms, such as denoising and color balancing. Demosaicking changes the image statistics by introducing structure into the noise. This can adversely affect the performance of later denoising and sharpening algorithms. Considering the capabilities of the sensor, and the color imaging pipeline in its entirety is important when evaluating demosaicking methods.

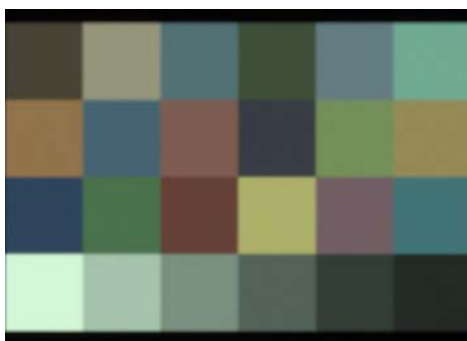
Color balancing

If one simply copies the sensor pixel values into the display values, the resulting image will not generally be a good color representation of the original scene. A color transform, also called a color correction, is used to convert the sensor data to display values that create a *perceptual*

match between the original and displayed images. Figure 10 compares corrected and uncorrected color images.

There are several factors that must be taken into account in designing the color transform. First, imaging sensors do not generally have the same spectral response as human viewers. Thus, the device and human eye weight the spectral image differently. Second, the properties of the display medium must be taken into account; for example, the transformation must be adjusted for different display primaries. Third, the camera often acquires the scene under ambient lighting conditions that differ from those in which humans view the display. The human visual system adjusts its sensitivities (adapts) in response to ambient lighting conditions. The differences in adaptation can be very large, and adaptation often causes the same physical stimulus to appear different under the two conditions. As a first-order approximation, adaptation between ambient viewing conditions generally preserves surface colors. Hence, the consequence of visual adaptation is commonly referred to as “color constancy”[58]. The color transform must account both for the physical properties of the sensor and display, as well as for the physiological adaptation between different ambient viewing conditions.

The complexity of the problem has spawned a large, complex and interesting literature that intermixes device and physiological modeling. The scope of that literature is too large for this review. Here we describe a simple but common approach for designing a color transform. To explain this approach, we first describe how to set the display primary intensities to create a radiance whose appearance matches the light reflected from a surface illuminated by a known illuminant, say 6500 Kelvin daylight. We refer to this process as **display rendering**. Then we describe how to calculate the expected sensor rgb values to this surface under a known light. We refer to this process as **image capture**. Finally, we describe a computational approach for converting the captured image data to display values. We refer to this process as **color balancing**.



(a)



(b)

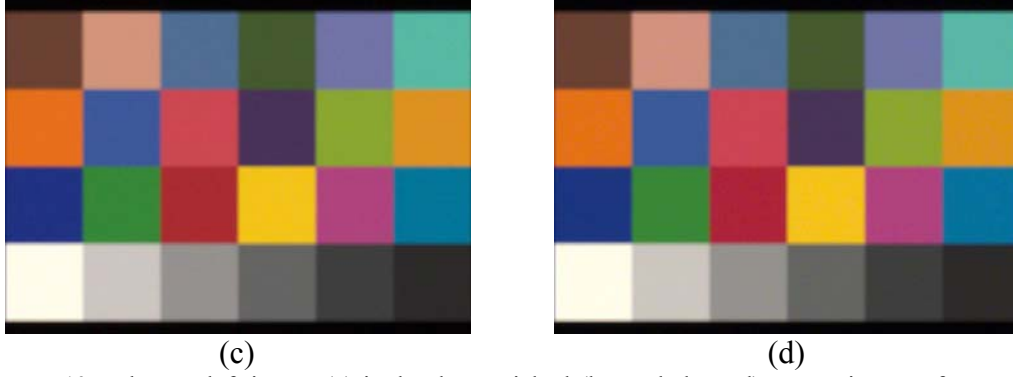


Figure 10. The top left image (a) is the demosaicked (but unbalanced) sensor image of a Macbeth ColorChecker illuminated with daylight (D65). The top right image (b) is based on a sensor image of the Macbeth ColorChecker illuminated with a tungsten source. The bottom images have been corrected using two different 3x3 matrix transforms, one optimized for daylight (c) and the other for tungsten (d).

Display rendering

Given a surface reflectance, $S(\lambda)$, and an illuminant with spectral energy, $E(\lambda)$, the light scattered toward the viewer will be $S(\lambda)E(\lambda)$. The impact of this light on the human eye is described by three numbers, the tri-stimulus coordinates, (X,Y,Z) . The X value is computed using the simple formula:

$$X = \int_v \bar{x}(\lambda) S(\lambda) E(\lambda) d\lambda \quad (1.11)$$

The integral is taken over the visible wavelengths, usually over the range from 370 to 770 nm. To compute the Y and Z terms, one simply replaces $\bar{x}(\lambda)$ with similar standardized functions $\bar{y}()$, $\bar{z}()$. These functions are an international standard found in many references [59].

To produce the same visual effect on a human subject as the surface and illuminant combination, we adjust the display rgb values to emit a light with the same XYZ values. The relationship is easiest to see in matrix form. Suppose that we place sampled values (assume N samples) of the CIE tristimulus functions, $\bar{x}(\lambda)$, $\bar{y}(\lambda)$, $\bar{z}(\lambda)$ in the columns of an Nx3 matrix, V. We place the color signal $S(\lambda)E(\lambda)$ in a vector, c . Then XYZ tristimulus values in the column vector, t , are computed from the matrix product $t = V'c$. Suppose there is a display with three primary lights with peak spectral power distributions $p_i(\lambda)$. We place these three sampled wavelength functions into the column of a matrix P. We combine the three relative intensities of the primary lights, w_i into a column vector w . To match the surface on the display, we require the color signal and the display to produce the same tristimulus coordinates

$$t = V'c = V'Pw \quad (1.12)$$

When an observer views the display and the original color signal in the same context, they will produce the same tristimulus values and thus match one another. We can solve for the intensities of the three primaries using the equation

$$w = (V^t P)^{-1} V^t c \quad (1.13)$$

There are two practical notes. First, the primary intensities (weights) for most displays are related to the digital values by a nonlinear function called the display gamma. This must be accounted for during a calibration process [60, 61]. Second, in some instances the solution for the weights will be outside of the range (0,1). In that case, the color signal is beyond the gamut for that display.

Image capture

The vector of camera rgb values, u , produced by exposure to c can be calculated using a similar set of equations. Suppose that the spectral responsivity of the red channel is $\bar{r}(\lambda)$. Then the red pixel level will be proportional to

$$R = \int_v \bar{r}(\lambda) S(\lambda) E(\lambda) d\lambda \quad (1.14)$$

Shifting to matrix notation, if we place the sampled wavelength functions of the three pixel spectral responsivities into the columns of the Nx3 detector matrix, D , the camera rgb values in the vector will be

$$u = Dc \quad (1.15)$$

Defining the color balancing transform

The goal of the color transform is to specify a means for converting the camera rgb values u into display primary intensities w . The simplest case is when the camera and display viewing contexts are the same. In that case, a color signal c produces a camera response u and is matched by display primary intensities w . One solution is to create a pair of matrices, U and W , whose columns are corresponding pairs of camera responses and display primaries. We then find a color transform matrix M that minimizes the least squares error of the equation $W = MU$, or minimizes the error in a perceptual space (e.g., CIELAB). Nonlinear functions or various more complex transforms based on look-up tables can also be used.

When the ambient lights in the display context and the scene differ, the situation is more complex. In that case, a tristimulus match between the original scene and the display will not be a satisfactory perceptual match. Consequently, we must find a new way to identify the display rgb values that represent the appearance of the original. It is during this calculation that the principles of human color constancy play a role.

As an example, assume that we know the spectral reflectance of several surfaces in the original scene. In that case, we can render the scene on the display as if it were illuminated by a standard

light for display viewing, such as a daylight spectral power distribution (e.g., D65). This provides the target display values, W . The rationale for this choice is that color constancy implies that perceived color is predicted by surface reflectance, not color signal. A simple way to implement this strategy is this: 1) Render a specific color target on the display under the standard illuminant, 2) acquire sensor images of that target under a series of different ambient illuminants, and 3) find a transformation between each of these sensor images and the display.

The transform will vary as we change the ambient illumination for the camera image. For example, when the surfaces are illuminated by tungsten, we calculate a transform, M_T , that maps the linear camera rgb values into the linear display rgb values that generate the perception that the surfaces were illuminated by daylight. When these surfaces are illuminated by a fluorescent light, we calculate a different transform M_F . In general, the transform depends on the spectral power of the scene illuminant, the standard display viewing illuminant, the camera, the surfaces used in the target, and the criterion that we use for the minimization – it might be a least-squares minimization or a minimization with respect to another error metric (e.g., CIELAB).

At this stage, we have described how to specify a transformation from the sensor data to the display data for a range of illuminants. There is one final step required to complete the color balancing algorithm. Given a camera image, we must estimate the illuminant so we know which transform to apply. Figure 11 shows what happens when a camera image is corrected using a color transform that is inappropriate for the scene illumination. Many **illuminant estimation** algorithms are described in the literature [62-70]. Some of these algorithms are simple and based on image statistics such as the mean rgb value or the ratio of the red and blue sensors. Others involve more elaborate Bayesian computations [65] or Retinex-style algorithms [71, 72].



Figure 11. Color images that have been corrected using a 3x3 transform that is optimized for daylight (D65). The left image (a) is based on a camera image of the Macbeth ColorChecker under the D65 daylight illumination and the right image (b) is based on a camera image of the Macbeth ColorChecker illuminated by tungsten.

Display

People judge the quality of a digital imaging system by viewing the final output rendered on a display or on a printer. Hence, we cannot evaluate the quality of a digital imaging system without simulating (or measuring) the intended display. In this section, we describe how to calculate the spatial-spectral radiance of a displayed image. This is the visual stimulus that actually reaches the eye.

A display can be satisfactorily characterized if emissions from the three color channels are independent [60]. Specifically, the emission from one channel should not be influenced by the levels in the other channels. When this condition is met, the spatial-spectral radiance for any image rendered on the display can be predicted by three types of measurements – the display gamma, the spatial point spread functions for the red, green and blue pixel components and the spectral power distributions of the display color primaries [2]. The display gamma converts digital values into a measure of the linear intensity. The point spread functions describe the spatial spread of light for each component within a color pixel, which we call the sub-pixels. The spectral power distributions describe the spectral radiance emitted from each sub- pixel. .

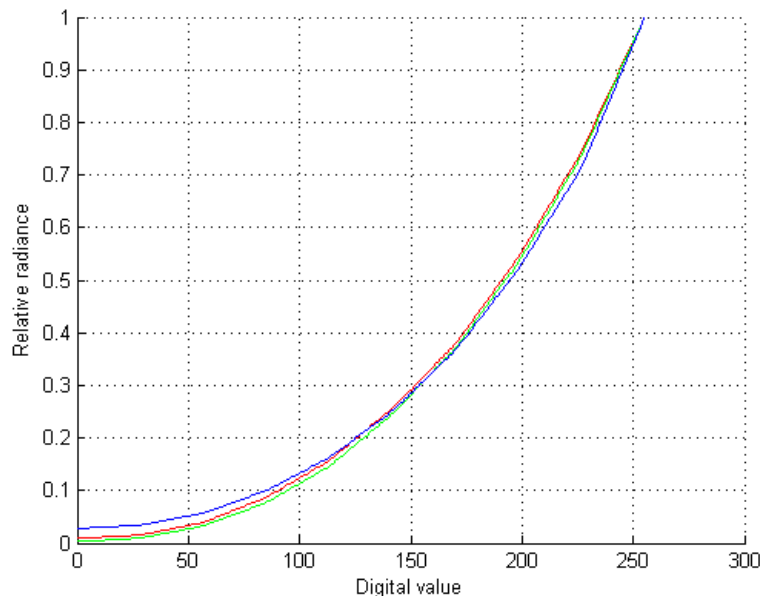


Figure 12. Display gamma for a calibrated LCD monitor. The gamma is comparable for other LCD, CRT and Plasma displays measured in our laboratory.

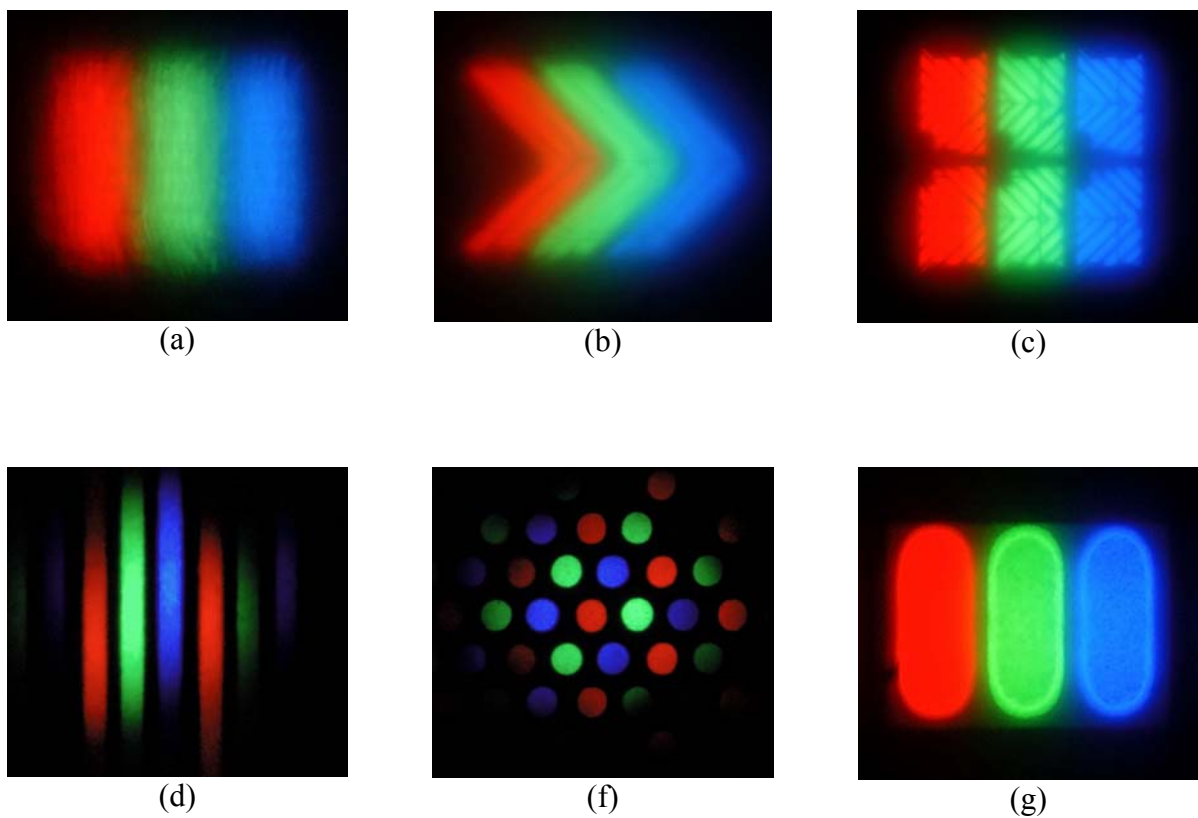
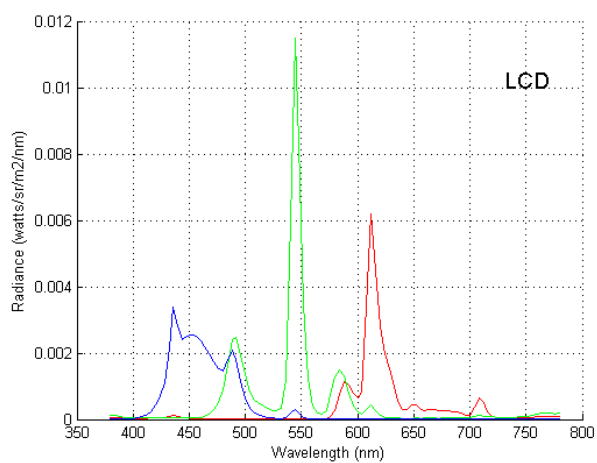


Figure 13. Magnified images of a white pixel in 6 different displays. Top images (a, b and c) are pixels in different LCD displays. The bottom images show pixels in d) a CRT with an aperture grille, e) a CRT with an hexagonal pixel array, and f) a plasma display.



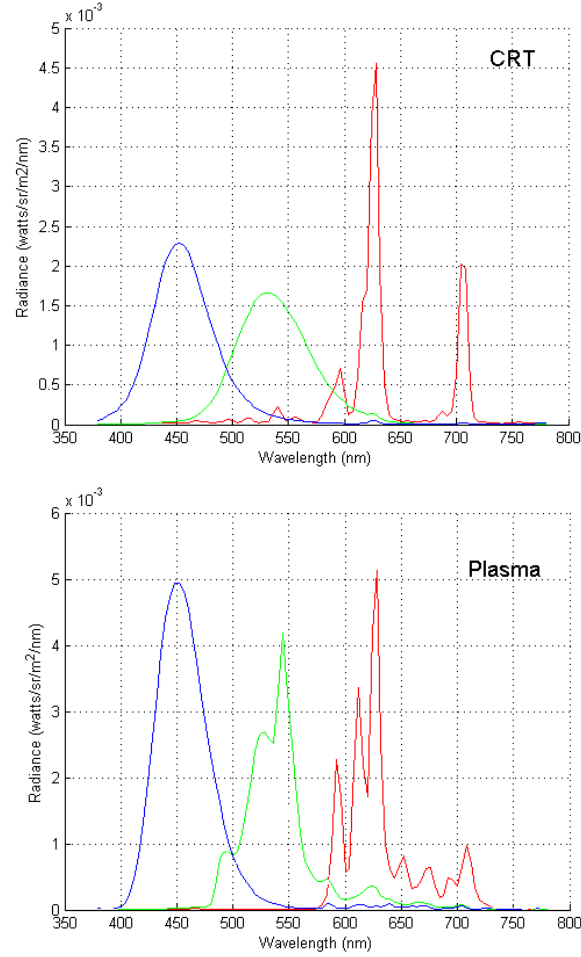


Figure 14 Spectral power distributions of red, green and blue color primaries in an LCD (top), CRT (middle) and Plasma display (bottom).

We model the sub-pixels as (a) space-wavelength separable and (b) independent. Separable means that the spectral power distribution emitted by a component is the same across the entire pixel. Formally, the spatial-spectral distribution of light emitted from the i^{th} sub-pixel can be defined by the product of two functions:

$$p_i(x, y, \lambda) = s_i(x, y)w_i(\lambda) \quad (1.16)$$

The function $w_i(\lambda)$ describes the spectral power distribution of the i^{th} sub-pixel. The function $s_i(x, y)$ is the spatial spread of the light from that sub-pixel. This additivity assumption means that the light emitted from the i^{th} sub-pixel does not depend on the intensity of the other sub-pixels. In this case, the radiance emitted by a pixel is the sum of the radiance emitted by the sub-pixels

$$p(x, y, \lambda) = \sum_i p_i(x, y, \lambda) \quad (1.17)$$

We introduce notation to describe the static nonlinearity between the digital value, $\mathbf{v} = (R, G, B)$, and the emitted light, referred to as the display gamma. We describe the static nonlinearity for the i^{th} sub-component as $g_i(v)$. The i^{th} gamma function converts the digital controller values, $\mathbf{v} = (R, G, B)$, into the intensity of each sub-pixel. Taking all of these assumptions together, we expect the spatial-chromatic image from a pixel, given a digital input, (R, G, B) , to be

$$\begin{aligned} p(x, y, \lambda) &= \sum_i g_i(v) p_i(x, y, \lambda) \\ &= \sum_i g_i(v) s_i(x, y) w_i(\lambda) \end{aligned} \quad (1.18)$$

These equations apply to the light emitted from a single pixel. We create the full display image by repeating this process across the array of display pixels. In so doing, we assume that the light emitted from a pixel is independent of the values at adjacent pixels. We refer to the spatial-spectral independence between pixels as display-independence [60].

Farrell et al [2] describe four conditions that are necessary for display-independence. First, the relative spectral power distribution of the display color primaries should be invariant as digital value increases (spectral homogeneity). Second, the spectral power distribution of any combination of pixel components can be predicted by the sum of the spectral power distribution of the individual pixel components measured separately (spectral additivity). Third, the relative spatial spread of each pixel component should be unchanged as digital values increase (spatial homogeneity). And fourth, the spatial distribution of light emitted by any combination of pixels is predicted by the sum of the spatial light distribution of the individual pixels (spatial additivity). Many LCD displays meet all four conditions. For CRT displays, however, spatial additivity holds only for vertically adjacent pixels and fails for horizontally adjacent pixels[2, 73]. The failure of spatial additivity for CRT displays can be explained by sample and hold circuitry. CRT manufacturers use sample and hold circuitry to compensate for the slew rate limitations of the electron beam as it moves horizontally across the screen.

It is possible to create a more complex model for CRTs. For example, the failures of additivity can be used as an empirical basis for modeling the distortion introduced by the sample and hold circuitry. Our measurements and display simulations show that this added complexity is necessary to adequately characterize these CRT displays and probably many others. But LCD displays are rapidly replacing CRT displays, both in the home and in the office. They are also an integral part of the imaging pipeline for digital cameras and camera phones. It is fortunate that most of the LCD devices we have tested satisfy the simple model properties. This makes it easier to calibrate, control and model these displays and thus predict their effect in the imaging pipeline.

Digital Camera Simulations

We have created software models for the scene, optics, sensor, processor and display in an integrated suite of Matlab® software tools - the Image Systems Evaluation Toolbox, or ISET[74]. The ISET simulation begins with scene data; these are transformed by the imaging optics into the optical image, an irradiance distribution at the image sensor array; the irradiance is transformed into an image sensor array response; finally, the image sensor array data are processed to generate a display image

In this section we use ISET to model the scene, optics and sensor of a 5 megapixel CMOS digital camera that we calibrated in our laboratory.

Simulation Parameters

The sensor manufacturer provided many of the simulation parameters. Other parameters, such as read noise, dark voltage, DSNU and PRNU were estimated from measurements in our laboratory (see Appendix A). We modeled the camera lens using a diffraction-limited model² with a lens f-number of 4 and a focal length of 3 mm. We simulated the effects of an optical diffuser that filters out signals above the Nyquist frequency limit of the imaging sensor. This was accomplished using a Gaussian filter with full-width half maximum equal to the pixel width.

Table 1 lists the sensor parameters along with the reference source for the data.

Table 1: Sensor parameters for ISET simulations		
Sensor Parameter	Parameter Value	Reference Source
Pixel width (μm)	2.2	manufacturer
Pixel height (μm)	2.2	manufacturer
CFA pattern	gbgr	manufacturer
Spectral quantum efficiencies		Measured (see Appendix B)
Dark voltage (V)	4.68 mV/sec	Measured (see Appendix A)
Read noise (mv)	0.89	Measured (see Appendix A)
DSNU (mv)	0.83 mV.	Measured (see Appendix A)
PRNU (%)	0.736	Measured (see Appendix A)
Fill factor	45%	manufacturer
Well capacity (electrons)	9000	manufacturer
Voltage swing (V)	1.8	manufacturer
Conversion Gain ($\mu\text{V}/\text{e}$)	2.0000e-004	manufacturer
Analog gain	1.0	software setting

² It possible to use lens specification data from lens design software programs such as ZEMAX® and CODE V®. When these data are available, a ray-trace method can be used to calculate point spread functions that are both wavelength-dependent and position-dependent (not shift-invariant). In our simulations, we analyze the central nxm portion of the sensor image where departures from diffraction-limited models are less evident.

Exposure duration (s)	100	measured
Scene luminance (cd/m^2)	61	measured
Lens f-number	4	lens setting
Lens focal length (mm)	3	lens calibration software[75]

The sensor spectral quantum efficiencies for the red, green and blue pixels were calculated by combining the effects of the lens transmittance, color filter arrays and photodiode quantum efficiency into one spectral sensitivity function for each red, green and blue pixels, respectively (Figure 15). Appendix B describes the laboratory measurements and calculations used to estimate these spectral curves.

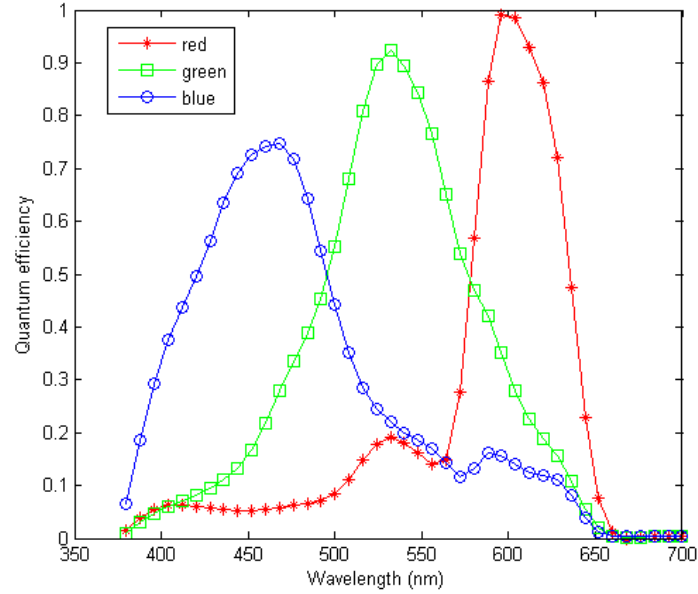


Figure 15. Spectral quantum efficiencies of red, green and blue channels for a calibrated imaging sensor.

Simulation Performance

To evaluate the quality of the simulation, we first measured the radiance image of a Macbeth ColorChecker using the methods described in the Scene section. We used these measured radiance data as the scene input in ISET simulations. Second, we acquired an image of this scene using the real camera. We compared the simulations with the real acquisition in several ways.

Figure 16 compares the predicted and simulated sensor images of the Macbeth ColorChecker after it is demosaicked using bilinear interpolation. These processed images (not color-balanced) illustrate the similarity between the measured and simulated sensor images. For a quantitative comparison, we calculated the mean and standard deviation of the pixels values for the 24 color patches in the sensor images of the Macbeth ColorChecker. A scatter plot of the signal-to-noise (mean divided by the standard deviation) in the measured and simulated images also shows good agreement for each of the twenty four patches in the Macbeth Color Checker (Figures 17 and 18).



Figure 16. Measured (a) and simulated (b) sensor images. The images have been demosaicked but not color-balanced.

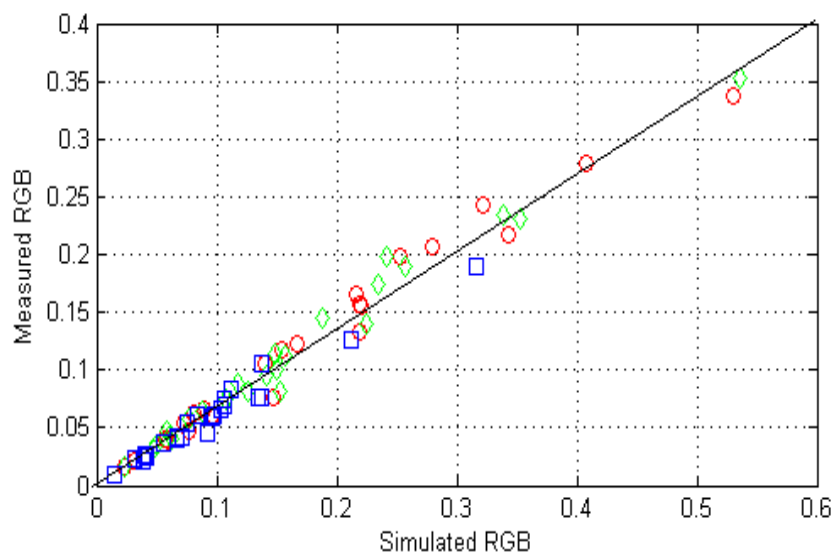


Figure 17. Mean pixel values averaged for each of the twenty four patches in the Macbeth Color Checker in the measured sensor images plotted against mean pixels values for the same patches in the simulated sensor images.

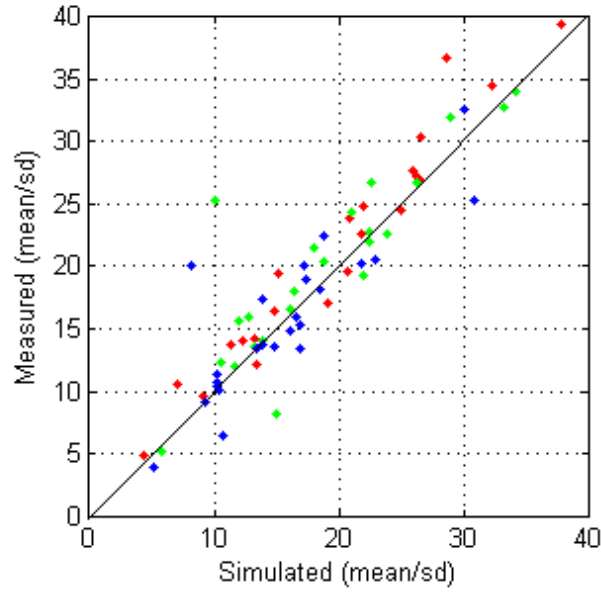


Figure 18. Standard deviation in pixel values for each of the twenty four patches in the Macbeth Color Checker in the measured sensor images plotted against standard deviation in pixels values for the same patches in the simulated sensor images.

An important simulation objective is to understand how accurately a sensor can render a color scene. The color accuracy of the measured and simulated imaging sensors are characterized in Figures 19 and 20, respectively. In this example we render a sensor image of a Macbeth ColorChecker. To perform the rendering we specify the linear sRGB [76] display values of an image that match a Macbeth ColorChecker under D65 illumination. These are the *desired* sRGB values. Then, we find the 3x3 matrix that best transforms (in a least squares sense) the sensor RGB values into the desired sRGB values. We compare the transformed and desired sRGB values for the measured (Fig 19) and simulated sensor images (Fig 20). Histograms of the CIELAB color difference error (ΔE) are also shown. Again, the simulation predicts the color accuracy of the sensor quite well.

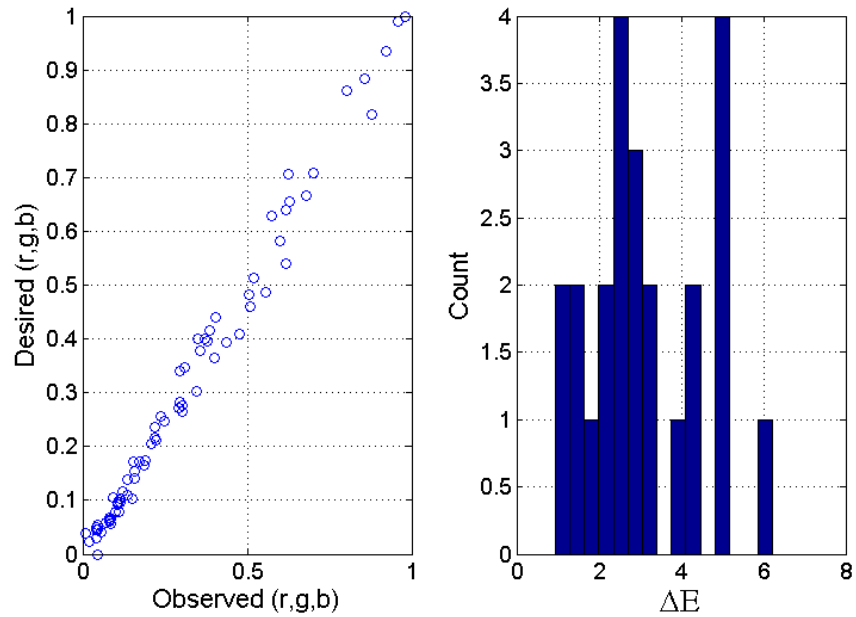


Figure 19. Color accuracy of measured imaging sensor. The graph on the left plots the desired sRGB values against the color-balance sRGB values derived from the measured sensor images. The histograms show the distribution of ΔE color errors for the 24 color patches in the Macbeth ColorChecker.

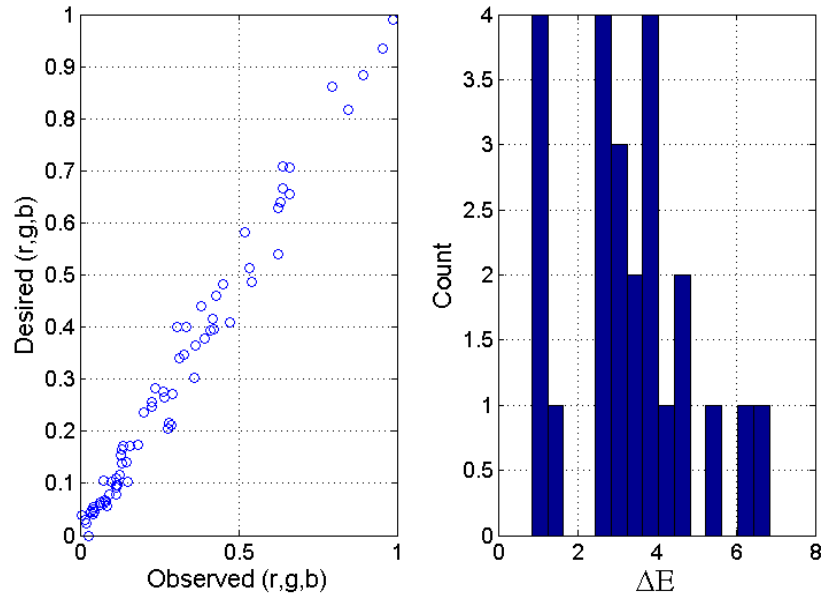


Figure 20. Color accuracy of simulated imaging sensor. The graph on the left plots the desired sRGB values against the color-balance sRGB values derived from the simulated sensor images. The histograms show the distribution of ΔE color errors for the 24 color patches in the Macbeth ColorChecker.

Figures 16 to 20 show that there is a good correspondence between measured and simulated sensor performance. The mean and variance in pixel values are nearly the same for simulated

and measured sensor images (see Figures 17 and 18). And both simulated and measured sensor images have comparable color accuracy (see Figures 19 and 20).

The simulation parameters were derived from a few fundamental measurements that characterize sensor spectral sensitivity and electrical properties including dark current, read noise, dark signal non-uniformity and photoreceptor non-uniformity. While we estimate these parameters from a modest set of calibration measurements (see Appendix A and B), these sensor parameters are often provided by the sensor manufacturer in the form of a product data sheet.

Applications

Once we have confidence in the simulations, it is further possible to estimate how the sensor will perform when acquiring scenes that are difficult to recreate or calibrate (high dynamic range, low light levels, and so forth). It is also possible to evaluate the effects that optical and sensor components have upon perceived image quality. As a simple example, Figure 21 shows the effects of reducing scene luminance; Figure 22 illustrates the effect that pixel size has on image quality. At any fixed exposure duration, the number of photons captured by the imaging sensor decreases with pixel size (Figure 22) and the results will be equivalent to reducing scene luminance, as illustrated in Figure 21.

We have used ISET simulations to investigate the tradeoff between pixel size and light sensitivity[77], the effects of camera motion [78], different sources of sensor noise [79], novel color filter arrays [80], and many other system parameters. Continuing validation of the simulation technology and methods for estimating the sensor characteristics should lead to high efficiencies for sensor evaluation. Ultimately, as we gain increasing confidence in the simulation, it can be used to design novel sensor designs as part of the manufacturing process.



Figure 21. Processed (demosaicked and color-balanced) images based on ISET simulations. The left image shows the results when the mean scene luminance is 5 cd/m^2 and the exposure duration is 20 msec. The right image shows the results when the mean scene luminance is 100 cd/m^2 and exposure duration is 20 msec. The pixel size is 2.2 micron for both simulations. Other sensor parameters are listed in Table 1.

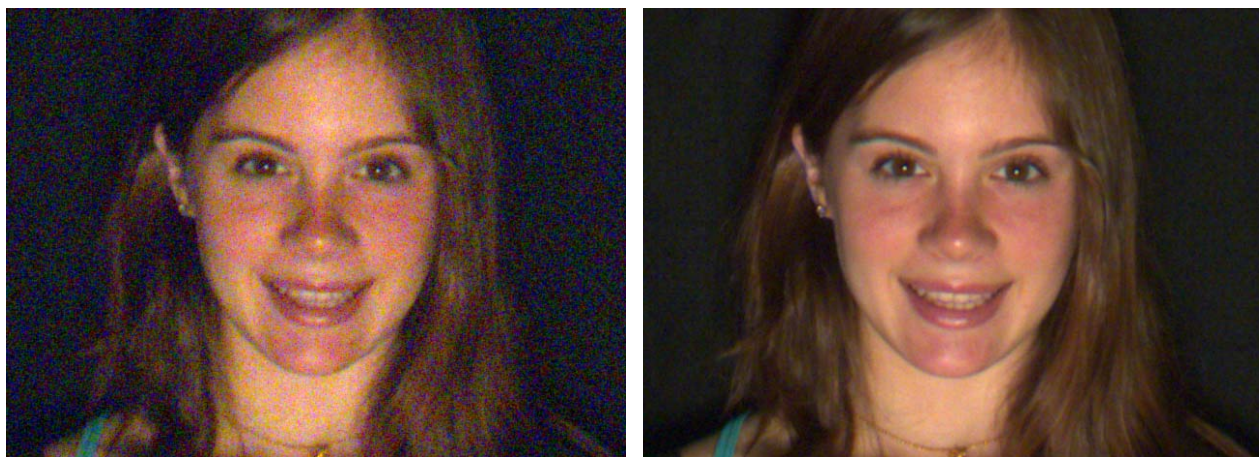


Figure 22. Processed (demosaicked and color-balanced) images based on ISET simulations of a sensor with 2.2 micron pixels with 45% fill factor (left) and a sensor with 4.4 micron pixels with 90% fill factor (right). The mean scene luminance is 5 cd/m^2 and the exposure duration is 20 msec. in both simulations. All other sensor parameters are listed in Table 1.

Summary

In this chapter we describe how to model and simulate the complete image processing pipeline of a digital camera, beginning with a radiometric description of the scene captured by the camera and ending with a radiometric description of the final image rendered on an LCD display. The ability to simulate sensor performance provides several advantages compared to a simulation entirely on the experimental bench.

First, creating and calibrating images to test components is essential; yet this is a major bottleneck in the laboratory. The availability of digitally accurate radiometric scenes for use in a simulator enables the user to predict sensor performance for a wide range of scenes that are difficult to create in the laboratory.

Second, a simulator makes it possible to evaluate the effects that different optical and sensor components have upon image quality, as well as different post-processing algorithms[80]. For example, the simulator could be used to verify the properties of a particular sensor in different imaging conditions or when coupled with different types of lenses.

Third, a simulator offers a common standard between laboratories. In principle, different laboratories can communicate about a sensor by simply sending the data files that characterize the sensor to one another. In summary, accurate simulations can determine which design changes will meet customers' expectations of image quality, without incurring the cost of building a new module.

Acknowledgements

We gratefully acknowledge support from Hewlett Packard, Logitech International, Olympus Corporation, Microsoft Corporation, Nikon Corporation and the Samsung Advanced Institute of Technology. We thank Norihiro Aoki from Nikon for providing us with many wonderful Nikon cameras and lenses and Bernd Girod for providing us with lab facilities. And we thank our many colleagues who have contributed to this work, including David Brainard, Tien Chen, Jeff DiCarlo, Keith Fife, Abbas El Gamal, Kunal Ghosh, Francisco Imai, Max Klein, Ricardo Motta, Amnon Silverstein, Poorvi Vora, Feng Xiao and Xuemei Zhang.

References

1. Catrysse, P., *Imaging Optics*, in *Handbook of Digital Imaging*, M. Kriss, Editor. 2008, John Wiley & Sons, Ltd.
2. Farrell, J., G. Ng, X. Ding, K. Larson, and B. Wandell, *A Display Simulation Toolbox for Image Quality Evaluation*, "IEEE/OSA Journal of Display Technology, 2008. **4**(2): p. 262-270.
3. Chen, T., P. Catrysse, A.E. Gamal, and B. Wandell. *How small should pixel size be?* in *SPIE Electronic Imaging Conference*. 2000. San Jose, CA.
4. Wandell, B.A., P. Catrysse, J.M. DiCarlo, D. Yang, and A. El Gamal. *Multiple Capture Single Image with a CMOS Sensor*. in *Chiba Conference on Multispectral Imaging*. 1999. Japan.
5. S. Kleinfelder, S.L., X. Liu, and A. El Gamal, *A 10,000 Frames/s CMOS Digital Pixel Sensor*. IEEE Journal of Solid State Circuits, 2001. **36**(12): p. 2049-2059.
6. Yang, D., A.E. Gamal, B. Fowler, and H. Tian, *A 640*512 CMOS image sensor with ultra wide dynamic range floating-point pixel-level ADC*. IEEE Journal of Solid State Circuits, 1999. **34**(12): p. 1821-1834.
7. Debevec, P. and J. Malik, *Recovering High Dynamic Range Radiance Maps from Photographs*. SIGGRAPH, 1997. **1997**.
8. Xiao, F., *A system study of high dynamic range imaging* Ph.D thesis, Stanford University. , 2003.
9. Yang, D., A. El Gamal, B. Fowler, and H. Tian, *A 640x512 CMOS Image Sensor with Ultrawide Dynamic Range Floating-Point Pixel-Level ADC*. IEEE Journal of Solid-State Circuits, 1999. **34**(12): p. 1821-1834.
10. Wandell, B.A., P.B. Catrysse, J.M. DiCarlo, D.X.D. Yang, and A. El Gamal, *Multiple Capture Single Image Architecture with a CMOS Sensor*, in *International Symposium on Multispectral Imaging and Color Reproduction for Digital Archives*. 1999, Society of Multispectral Imaging of Japan: Chiba, Japan. p. 11-17.
11. Martinez, K., J. Cupitt, and D. Saunders, *High resolution colorimetric imaging of paintings*. Proceedings of the SPIE, 1993. **1901**: p. 25-36.

12. Martinez, K., J. Cupitt, D. Saunders, and R. Pillay, *Ten years of Art Imaging Research*. Proceedings of the IEEE, 2002. **90**(1): p. 28-41.
13. Imai, F. and R. Berns, *Spectral estimation using trichromatic digital cameras*, in *International Symposium on Multispectral Imaging and Color Reproduction for Digital Archives* 1999, Society of Multispectral Imaging of Japan: University of Chiba, Japan. p. 42-49.
14. Tominaga, S., *Multichannel vision system for estimating surface and illumination functions*. J. Opt. Soc. Am A, 1996. **13**: p. 2163–2173.
15. Vora, P.L., J.E. Farrell, J.D. Tietz, and D.H. Brainard, *Image Capture: Simulation of Sensor Responses from Hyperspectral Images*. IEEE Transactions on Image Processing, 2001. **10**: p. 307-316.
16. Farrell, J.E., J. Cupitt, D. Saunders, and B. Wandell, *Estimating spectral reflectances of digital images of art*, in *The International Symposium on Multispectral Imaging and Colour Reproduction for Digital Archives*. 1999, Society of Multispectral Imaging of Japan: Chiba University, Japan.
17. Wandell, B.A., *Computational methods for color constancy*, in *Frontiers of Visual Science*, N. The Committee on Vision, Editor. 1987: Washington, D.C. p. 109-118.
18. Marimont, D.H. and B.A. Wandell, *Linear models of surface and illuminant spectra*. Journal of the Optical Society of America A, 1992. **9**(11): p. 1905-1913.
19. Maloney, L.T. and B. Wandell, *Color constancy: a method for recovering surface spectral reflectance*. J. Opt. Soc. Am. A, 1986. **1**: p. 29-33.
20. Buchsbaum, G., *A spatial processor model for object color perception*. J Franklin Institute, 1980. **310**: p. 1-26.
21. Horn, B.K.P., *Exact reproduction of colored images*. Computer Vision, Graphics and Image Processing, 1984. **26**: p. 135-167.
22. Farrell, J., *Spectral based color Image editing*. Proceedings of the Fourth IST and SID Conference on Color Imaging, 1996.
23. Parkkinen, J.P.S., J. Hallikainen, and T. Jaaskelainen, *Characteristic spectra of Munsell colors*. Journal of the Optical Society of America, 1989. **6**: p. 318-322.
24. Judd, D.B., D.L. MacAdam, and G.W. Wyszecki, *Spectral distribution of typical daylight as a function of correlated color temperature*. J. Opt. Soc. Am., 1964. **54**: p. 1031.
25. Cohen, J., *Dependency of the spectral reflectance curves of the Munsell color chips*. Psychon. Sci, 1964. **1**: p. 369-370.
26. Krinov, E.L., *Surface reflectance properties of natural formations*. National Research Council of Canada: Technical Translation, 1947. **TT-439**.
27. Parmar, M., F. Imai, S.H. Park, and J. Farrell, *A database of high dynamic range visible and near-infrared multispectral images*. Proceedings of the SPIE, 2008 **6817**.
28. Foster, D.H., S.M.C. Nascimento, and K. Amano, *Information limits on neural identification of coloured surfaces in natural scenes*. Visual Neuroscience, 2004. **21**: p. 331-336.
29. Parmar, M., M. Klein, J. Farrell, and B. Wandell, *An LED based multispectral imaging system*. 2009, Paper in preparation.
30. Wilburn, B., J. Joshi, V. Vaish, E. Talvala, E. Antunez, A. Barth, A. Adams, M. Levoy, and M. Horowitz, *High Performance Imaging Using Large Camera Arrays*. ACM Transactions on Graphics, 2005. **24**(3).

31. Martinez, K., S. Perry, and J. Cupitt, *Object browsing using the Internet Imaging Protocol*. Computer Networks, 2000. **33**: p. 800-810.
32. Driggers, R.G. Encyclopedia of Optical Engineering, ed. R.G. Driggers. Vol. 1. 2003: CRC Press,.
33. Goodman, J., *The frequency response of a defocused optical system*. Proceedings of the Royal Society A, 1955. **231**: p. 91-103.
34. Goodman, J.W., *Introduction to Fourier optics*. 2nd ed. 1996, New York: McGraw-Hill.
35. Tian, H. and A. El Gamal, *Analysis of 1/f noise in CMOS APS*. Proc. SPIE 2000. **3965**.
36. Janesick, J.R., *Scientific Charge-Coupled Devices*. Vol. PM83. 2001: SPIE Publications.
37. Kremens, R., N. Sampat, S. Venkataraman, and T. Yeh, *System implications of implementing auto-exposure on consumer digital cameras*. Proceedings of the SPIE Electronic Imaging, 1999. **3650**.
38. Kuno, T., *A new automatic exposure system for digital still cameras*. IEEE Transactions on Consumer Electronics, 1998. **44**(1): p. 192-199.
39. Shimizu, S., *A new algorithm for exposure control based on fuzzy logic for video cameras*. IEEE Transactions on Consumer Electronics, 1992. **38**: p. 617-623.
40. Johnson, B.K., *Photographic exposure control system and method*, in U.S. Patent 4,423,936. 1984.
41. Muramatsu, M., *Photometry device for a camera*, in U.S. Patent 5,592,256. 1997.
42. Takagi, T., *Auto-exposure device of a camera*, in U.S. Patent 5,596,387. 1997.
43. Gunturk, B.K., J. Glotzbach, Y. Altunbasak, R.W. Schafer, and R.M. Mersereau, *Demosaicking: Color filter array interpolation in single-chip digital cameras*. IEEE Signal Processing Magazine, 2005. **22**(1): p. 44-54.
44. Taubman, D., *Generalized Wiener reconstruction of images from colour sensor data using a scale invariant prior*. Proceedings of the IEEE International Conference on Image Processing, 2000. **3**: p. 801-804.
45. Kapah, O. and H.Z. Hel-Or, *Demosaicking using artificial neural networks*. Proceedings of the SPIE 2000. **3962**(112).
46. Hel-Or, Y. and D. Keren, *Demosaicking of Color Images using Steerable Wavelets*. HP Labs Technical Report HPL-2002-206R1, Aug. 2002.
47. Brainard, D.H., *Bayesian method for reconstructing color images from trichromatic samples*. , in *Proceedings of the IS&T 47th Annual Meeting*. 1994: Rochester, NY. p. 375-380.
48. Mukherjee, J., R. Parthasarathi, and S. Goyal, *Markov random field processing for color demosaicking*. Pattern Recognition Letters, 2001. **22**(339-351).
49. Gunturk, B.K., Y. Altunbasak, R.M. Mersereau, and . *Color plane interpolation using alternating projections*,. IEEE Transactions on Image Processing, 2002. **11**(9): p. 997-1013.
50. Hirakawa, K. and T.W. Parks, *Adaptive homogeneity-directed demosaicking algorithm*. IEEE Transactions on Image Processing, 2005. **14**(3): p. 360-369.
51. Glotzbach, J.W., R.W. Schafer, and K. Illgner, ., , *A method of color filter array interpolation with alias cancellation properties*. IEEE Transactions on Image Processing, 2001. **1**: p. 141-144.
52. Adams, J.E. and J.F. Hamilton, *Design of practical color filter array interpolation algorithms for digital cameras*. Proceedings of the SPIE, 1997. **3028**: p. 117-125.

53. Kimmel, R., *Demosaicing: image reconstruction from color CCD samples*. IEEE Transactions on Image Processing, 1999. **8**(9): p. 1221-8.
54. Parmar, M., *Bayesian Restoration of Color Images using a Non-Homogenous Cross-Channel Prior*, in *Proceedings of the International Conference on Image Processing*. 2007. p. 505-508.
55. Alleysson, D., S. Susstrunk, and J. Herault, *Linear color demosaicing inspired by the human visual system*. IEEE Transactions on Image Processing, 2005. **14**(1): p. 439-449.
56. Dubois, E., *Frequency-domain methods for demosaicking of Bayer-sampled color images*. IEEE Signal Processing Letters 2005. **12**(847-850).
57. Hirakawa, K. and P.J. Wolfe, *Spatio-Spectral Color Filter Array Design for Enhanced Image Fidelity*. Proceedings of the International Conference on Image Processing, 2007. **2**: p. 81-84.
58. Wandell, B.A., *Foundations of Vision*. 1995: Sinauer Associates, Inc.
59. Wyszecki, G. and W.S. Stiles, *Color science: concepts and methods, quantitative data and formulae*. 1982.
60. Brainard, D.H., *Calibration of a computer controlled color monitor*. Color Research and Application, 1989. **14**: p. 23-34.
61. Wandell, B.A., *Foundations of vision*. 1995, Sunderland: Sinauer Associates. 476.
62. Tominaga, S. and B.A. Wandell, *Natural scene-illuminant estimation using the sensor correlation*. Proceedings of the IEEE, 2002. **90**(1): p. 42-56.
63. Tominaga, S. and B. Wandell, *Standard surface-reflectance model and illuminant estimation*. Journal of the Optical Society of America A., 1989. **6**(4): p. 576-584.
64. Tominaga, S., S. Ebisui, and B.A. Wandell, *Scene illuminant classification: brighter is better*. Journal of the Optical Society of America A (Optics, Image Science and Vision), 2001. **18**(1): p. 55-64.
65. Brainard, D. and W. Freeman, *Bayesian color constancy*. Journal of the Optical Society of America A, 1997. **14**(7): p. 1393-1411.
66. DiCarlo, J., P. Catrysse, F. Xiao, and B. Wandell, *System and Method for Estimating Physical Properties of Objects and Illuminants in a Scene using Temporally Modulated Light Emission*. Patent pending, Stanford University: USA.
67. DiCarlo, J.M. and B.A. Wandell, *Illuminant Estimation: Beyond the Bases*, in *Eighth Color Imaging Conference*. 2000: Scottsdale, AZ. p. 91-96.
68. Finlayson, G.D., P.M. Hubel, and S. Hordley, *Color by correlation*, in *Proceedings of the Fifth Color Imaging Conference*. 1997, IS&T: Springfield, VA. p. 6-11.
69. Maloney, L.T. and B.A. Wandell, *Color constancy: a method for recovering surface spectral reflectance*. Journal of the Optical Society of America A, 1986. **3**(1): p. 29-33.
70. Lee, H.-C., *Method for computing the scene-illuminant chromaticity from specular highlights*. Journal of the optical Society of America A., 1986. **3**: p. 1694-1699.
71. Brainard, D.H. and B.A. Wandell, *Analysis of the retinex theory of color vision*. Journal Optical Society of America A, 1986. **3**(10): p. 1651-1661.
72. Kimmel, R., M. Elad, D. Shaked, R. Keshet, and I. Sobel, *A Variational Framework for Retinex*. Journal of Computer Vision, 2003. **52**: p. 7-23.
73. Lyons, N.P. and J.E. Farrell, *Linear systems analysis of CRT displays*, in *1989 SID International Symposium. Digest of Technical Papers*. 1989, Soc. Inf. Display: Playa del Ray, CA, USA. p. x+440.

74. Farrell, J., F. Xiao, P. Catrysse, and B. Wandell, *A simulation tool for evaluating digital camera image quality*. Proceedings of the SPIE, 2004. **5294**: p. 124-131.
75. Bouguet, J.-Y. *Camera calibration toolbox for Matlab*. 2008 June 6, 2008 [cited; Available from: http://www.vision.caltech.edu/bouguetj/calib_doc/].
76. Stokes, M., M. Anderson, S. Chandrasekar, and R. Motta. *Standard Default Color Space for the Internet - sRGB*. 1996 [cited; Available from: <http://www.w3.org/Graphics/Color/sRGB.html>].
77. Farrell, J., F. Xiao, and S. Kavusi, *Resolution and light sensitivity tradeoff with pixel size*. Proceedings of the SPIE 2006. **6069**.
78. Xiao, F., A. Silverstein, and J. Farrell, *Camera motion and effective spatial resolution*. Proceedings of the International Congress of Imaging Science, 2006.
79. Farrell, J., M. Okincha, and M. Parmar, *Sensor calibration and simulations*. Proceedings of the SPIE, 2008. **6817**.
80. Parmar, M. and B.A. Wandell, *Interleaved Imaging: Improving Sensitivity with Simultaneous Acquisition of Wideband and RGB Channels*. SPIE/IS&T Conference on Electronic Imaging, 2009, 2009.
81. DiCarlo, J.M., E. Montgomery, and S.W. Trovinger. *Emissive chart for imager calibration*. in *Twelfth Color Imaging Conference: Color Science and Engineering Systems*. 2004: The Society for Imaging Science and Technology.

Appendix A: Characterizing sensor noise

We describe how to estimate several types of sensor noise from unprocessed (“raw”) sensor data.

Dark voltage or dark current is thermally-generated electron noise in the absence of light. To characterize dark voltage, capture many images in the dark at a set of different exposure durations. The rate of increase in pixel digital values (DV) over time is proportional to the dark voltage. The dark voltage (volts/sec) can be derived from the DV/time data using the voltage swing and number of quantization levels.

Read noise is the variance in digital values from repeated reads of the same pixel. To measure read noise, capture many images in the dark with the same very short exposure duration. The exposure duration should be as short as possible in order to avoid contributions from dark voltage. Read noise is the standard deviation in the multiple measurements obtained in the dark with the same (short) exposure duration. It has units of volts.

Dark signal non-uniformity (DSNU) is the variability across pixels in dark voltage. DSNU can be estimated by averaging multiple measurements in the dark with constant exposure duration and then calculating the standard deviation of the mean pixel value across the array of pixels. Averaging many dark images reduces the contribution of read noise. DSNU has units of volts.

Photoreceptor non-uniformity (PRNU) is the standard deviation in sensitivity across pixels. PRNU can be estimated by analyzing raw sensor images of a uniform light field captured at a series of exposure durations. Do not include sensor images that are dominated by noise at short durations or saturated at long exposure durations. Then, for each pixel, calculate the increase in mean digital value as exposure duration increases. The slope differs between color channels because each has its own light sensitivity. The standard deviation of the slope, measured as a proportion of the mean slope for that channel, is the same across the colored pixels. This standard deviation is the PRNU and is dimensionless.

Appendix B: Characterizing channel spectral efficiency

The spectral sensitivity of a color channel depends on the spectral quantum efficiency of the photodetector and the spectral transmittance of optical elements and filters in the imaging path. For instrument design, one would like to measure the spectral transmission of all of these. Measuring the components, however, requires access prior to assembly or disassembling the device; often neither of these options is practical. Here, we describe how to measure the combined effect of these components which is called the spectral efficiency of the channel.

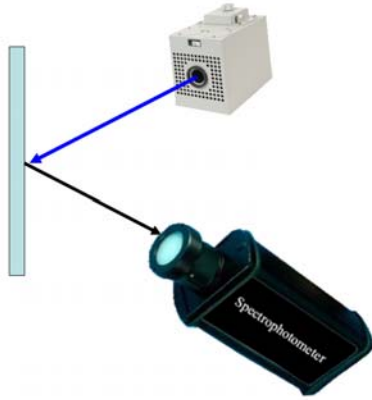
To measure the channel spectral efficiency we record the channel response to a series of narrowband lights. The wavelength range of these lights should span 400 and 700 nm, matching the range of human vision and the typical range of consumer cameras. One method of creating such lights is to use a monochromator which separates broadband light into monochromatic wavelength components. Another way to create narrowband lights is to use a set of LEDs with different peak wavelengths that sample the visible range [81]

Figure 23 illustrates a procedure for measuring the spectral efficiency of the rgb color channels in a digital camera. We illuminate a surface with a Lambertian reflectance, such as a flat piece of magnesium-oxide chalk, with narrowband light from a monochromator. Using a spectrophotometer, we measure the spectral radiance of each of the narrowband lights (Fig 23a). The spectral radiance measured for each of the narrowband lights is combined into a matrix, L . The matrix is $N \times M$, where N is the number of wavelength samples M is the number of spectral lights.

Second, place the camera at the same location used for the spectrophotometer measurement (Fig 23b); capture an image of each of the narrowband lights. For each wavelength band, use the camera exposure duration that yields the highest SNR. Normalize the linear camera values (i.e., correct for differences in exposure duration by dividing the rgb values to obtain a total response per unit time). To avoid the effects of chromatic aberration, calculate the mean r, g and b values for the central region of the normalized camera image. Place the corrected rgb values for each of the M lights in the columns of a $3 \times M$ matrix, C . We estimate the channel spectral sensitivity by using a robust method to solve for S in the linear equation: $C = S^T L$.

We recommend measuring the lens spectral transmissivity independently. Manufacturers use unique lens coatings that can strongly influence the channel spectral efficiency. Hence, the channel spectral efficiency can depend strongly on the imaging lens.

A



B

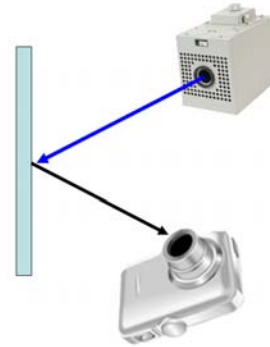


Figure 23. A) Illuminate a Lambertian surface with narrowband lights spanning 400 – 800 nm using a monochromator. Measure the spectral radiance of each of the narrowband lights using a spectrophotometer. B) Place the camera at the same location used for the spectrophotometer measurement and capture an image of each of the narrowband lights.