Digital camera simulation

Joyce Farrell,[1,*] Peter B. Catrysse,[1,2] and Brian Wandell[1,3]
[1]Department of Electrical Engineering, Stanford University, Stanford, California 94305, USA
[2]Edward L. Ginzton Laboratory, Stanford University, Stanford, California 94305, USA
[3]Department of Psychology, Stanford University, Stanford, California 94305, USA
*Corresponding author: Joyce_Farrell@stanford.edu

Received 5 October 2011; revised 22 December 2011; accepted 23 December 2011;
posted 10 January 2012 (Doc. ID 156026); published 1 February 2012

We describe a simulation of 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 image rendered on a display. We show that there is a good correspondence between measured and simulated sensor performance. Through the use of simulation, we can quantify the effects of individual digital camera components on system performance and image quality. This computational approach can be helpful for both camera design and image quality assessment. © 2012 Optical Society of America

OCIS codes: 100.3000, 100.0100.

1. Introduction

Digital cameras are designed by a team of engineers and scientists who have different expertise and who use different analytical tools and language to characterize the imaging component they work on. Typically, these engineers work at different companies that specialize in the design and manufacturing of one imaging component, such as optical lenses, filters, sensors, processors, or displays.

Digital cameras are purchased by consumers who judge the image quality of the digital camera by viewing the final rendered output. Achieving a high quality output depends on 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 multidevice 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.

There has been much progress in modeling the individual components of a digital camera, including camera optics [1–3] and sensor noise [4–9]. Progress has also been made in modeling the spectral reflectances and illuminants in a scene [10–17]. There is also a very large literature on image processing algorithms that are part of any digital camera [18]. This is an opportune time to develop a system simulator that incorporates models for each of the system components. In this paper, we describe how to model and simulate the complete imaging 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.

Computer simulations have played an important role in evaluating remote imaging systems that are used to classify agricultural plants and materials and to detect and identify buildings, vehicles, and other targets [19–22]. Modeling the complete system for remote imaging systems includes (1) characterizing spectral properties of possible targets, (2) modeling atmospheric conditions, (3) characterizing the spectral transmissivity of filters, the sensitivity, noise, and spatial resolution of imaging sensors (typically one pixel sensors), and (4) implementing image processing operations, including exposure

1559-128X/12/040A80-11$15.00/0

control, quantization, and detection algorithms. The veridicality of the simulations is judged by how well the models predict target detection as quantified by receiver operating curves. The visual abilities of the human observer are also modeled when the final detector is a human [22,23].

We describe a simulation for consumer imaging that parallels this methodology. We (1) characterize the radiometric properties of scenes and illuminants, (2) model the optical properties of lenses, (3) characterize the sensitivity and noise of sensors, including spatial and spectral sampling of color filter arrays, (4) implement image processing algorithms, and (5) generate a radiometric representation of displayed images. We also model spatial and chromatic sensitivity of human observers for purposes of predicting the visibility of noise and sampling artifacts [24,25].

The digital camera simulations are comprised of an integrated suite of MATLAB software tools referred to as the Image Systems Evaluation Toolbox (ISET) [26]. ISET incorporates and extends the work that we and our colleagues have been doing over the past 15 years on modeling and evaluating the quality of imaging systems [24–35]. We first describe the computational modules in ISET and then validate the system by comparing simulated and measured sensor data obtained from a calibrated device in our laboratory.

2. Digital Camera Simulation

The digital camera imaging pipeline can be separated into a sequence of computational modules corresponding to the scene, optics, sensor, processor, and display. The **scene** module creates a radiometric description of the scene. The **optics** module converts scene radiance data into an irradiance image at the sensor surface. The **sensor** module converts the irradiance image into electrons. The **processor** module converts the digital values in the two-dimensional sensor array into a three-dimensional (RGB) image that can be rendered on a specified display. Finally, the **display** module generates a radiometric description of the final image as it appears on an LCD display. We describe these five modules in the sections below.

A. 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/s/nm/sr/m^2) at each pixel in the sampled scene.

There are several different sources of scene data. The simplest are synthetic scenes, such as the Macbeth ColorChecker, spatial frequency sweep patterns, intensity ramps, and uniform fields. These are the spectral radiance image data that arise from a single image plane at a specified distance from the optics. 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.

We can also create synthetic scenes using three-dimensional multispectral data generated by other software [36]. These three-dimensional rendering methods provide data in which both the spectral radiance and the depth are specified. Such data can be used to simulate the effect of optical depth of focus [37], synthetic apertures [38], and light-field cameras [39].

Another important source of scene data are measurements of natural scenes using multispectral imaging methods [40–44]. These data provide insights about the typical dynamic range and spectral characteristics of the likely scenes.

The ISET spectral radiance scene data can be stored in a compact wavelength format using a linear model for the spectral functions. Hence, a relatively small number (four–six) of chromatic samples—along with the modest overhead of the basis functions—can represent the full spectral radiance information. In addition, the illuminant spectral power distribution is typically stored in the scene representation; this enables simulation of illumination changes.

B. Optics

The **optics** module converts scene radiance data into an irradiance image (photons/s/nm/m^2) 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 [45].

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.

1. Converting Radiance to Irradiance

The camera equation [46,47] defines a simple model for converting the scene radiance function, L_{scene}, to the optical irradiance field at the sensor, I_{image}. The camera equation is

$$I_{\text{image}}(x,y,\lambda) \cong \frac{\pi T(\lambda)}{4(f/\#)^2} L_{\text{scene}}\left(\frac{x}{m},\frac{y}{m},\lambda\right). \qquad (1)$$

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 lens transmissivity. The camera equation holds with good precision for the center of the image (i.e., on the optical axis). For all other image locations, we apply an off-axis (relative illumination) correction.

2. *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 [48]:

$$R(x,y,\lambda) = \cos^4\theta \approx \left(\frac{d}{S}\right)^4. \qquad (2)$$

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.

3. *Irradiance Image Blurring*

The irradiance image, $I_{\text{image}}(x,y,\lambda)$ cannot be a perfect replica of the scene radiance, $L_{\text{scene}}(x,y,\lambda)$. 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. A diffraction-limited system can be modeled as having a wavelength-dependent, shift-invariant point spread function (PSF) [49,50]. Diffraction-limited modeling uses a wave-optics approach to compute the blurring caused by a perfect lens with a finite aperture. The PSF of a diffraction-limited lens is quite simple, depending only on the f-number 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 PSF, which is also called the optical transfer function (OTF).

The formula for the diffraction-limited OTF is

$$\text{OTF} = \begin{cases} \frac{2}{\pi}a\ \cos(\rho) - \left(\rho\sqrt{1+\rho^2}\right), & \rho < 1 \\ 0, & \rho \geq 1 \end{cases}, \qquad (3)$$

where $\rho = f(A/(\lambda d))$ (normalized frequency), in which f = frequency in cycles/meter, A = aperture diameter in meters, λ = wavelength, and d = distance between the lens aperture and detector.

Shift-invariant image formation. The diffraction-limited PSF is a specific instance of a shift-invariant linear model. 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 PSF into the spatial frequency domain. The shift-invariant convolution in the space domain is a pointwise product in the spatial frequency domain. Hence, we have

$$\text{FT}\{I_{\text{image}}(x,y,\lambda)\} = \text{FT}\{\text{PSF}(x,y,\lambda)\} \cdot \text{FT}\{I_{\text{ideal}}(x,y,\lambda)\}, \qquad (4)$$

$$\text{FT}\{I_{\text{image}}(x,y,\lambda)\} = \text{OTF}(f_x,f_y,\lambda) \cdot \text{FT}\{I_{\text{ideal}}(x,y,\lambda)\}, \qquad (5)$$

$$I_{\text{image}}(x,y,\lambda) = \text{FT}^{-1}\{\text{OTF} \cdot \text{FT}\{I_{\text{ideal}}(x,y,\lambda)\}\}, \qquad (6)$$

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

Shift-variant image formation: ray-tracing. When measuring over large fields, in real systems the PSFs vary considerably. The ray-trace method model replaces the single, shift-invariant, PSF with a series of wavelength-dependent PSFs that vary as a function of field height and angle. In addition, ray tracing must account for the geometric distortion. This can be specified as a displacement field that varies as a function of input image position $[d(x,y)]$. In the ray-trace calculation, the displacement field is first applied and the result is blurred by the local PSF, wavelength by wavelength.

When using the ray-trace method, it is necessary to specify the geometric distortion and how the point spread varies with field height and wavelength. For real systems, these functions can be specified by the user or calculated using lens design software [51].

C. Sensor

The sensor module transforms 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 of 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/s/nm/m^2) is determined by the sensor spectral quantum efficiency ($S(\lambda)$, the e^-/photon), aperture function across space $A_i(x)$, and exposure time (T, s). For the ith photodetector, the number of electrons will be summed across the aperture and wavelength range:

$$e_i = T \iint\limits_{\lambda,x} S_i(\lambda) A_i(x) I(\lambda, x) \mathrm{d}\lambda \mathrm{d}x. \tag{7}$$

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 Eq. (7). Here, we outline the factors and computational steps that are incorporated in the simulation.

1. *Computing the Signal Current Density Image*

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 pixels in the sensor array are typically the same, each is covered by a color filter that permits certain wavelengths of light to pass more efficiently than 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 [52]. 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 redirection 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 [29].

2. *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.

3. *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 to 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 captures only a fraction of the incident photons, in part because of the material properties and in part because the photodetector occupies only a portion of the surface at the bottom of the pixel. The spectral quantum efficiency is a wavelength-dependent function that describes the expected fraction of 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 photoresponse 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 types of fixed pattern noise (FPN)—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) [53,54]

4. *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 the 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^{-1}).

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.

D. Processor

The processor module converts the digital values in the two-dimensional sensor array into an RGB image that can be rendered on a specified display. This is accomplished by controlling exposure duration, 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). There are many different approaches to autoexposure, demosaicking, and color balancing, and describing these methods is beyond the scope of this paper. ISET implements several algorithms that are in the public domain.

E. Display

The display module generates a radiometric description of the final image as it appears on an LCD display. It is important to calculate the spatial–spectral radiance emitted from the displayed image because, unlike the digital image values generated by the processor, this is the stimulus that actually reaches the eye. Simplifying the process of modeling the radiance distribution makes it possible to use the radiance field as the input to objective image quality metrics based on models of the human visual system.

ISET uses three functions to predict the spatial–spectral radiance emitted by a display. First, the display gamma is used as a lookup table to convert digital values into a measure of the linear intensity. Second, pixel PSFs for each color component (subpixel PSF) are used to generate a spatial map of linear intensity for each of the display color primaries. Third, the spectral power distributions of the color primaries are used to calculate the spectral composition of the displayed image. These three functions—the display gamma, the subpixel PSFs, and the spectral power distributions—are sufficient to characterize the performance of linear displays with independent pixels.

More formally, the spatial–chromatic image from a pixel, given a digital input, (R,G,B), is

$$p(x,y,\lambda) = \sum_i g_i(v) s_i(x,y) w_i(\lambda), \tag{8}$$

where $g_i(v)$ represents the display gamma for each color primary, $s_i(x,y)$ is the spatial spread of the light for each color subpixel, and $w_i(\lambda)$ is the spectral power distribution of the color primary.

These equations apply to the light emitted from a single pixel. The full display image is created by repeating this process across the array of display pixels. This calculation assumes that the light emitted from a pixel is independent of the values at adjacent pixels. These assumptions are a practical starting point for display simulation, although they may not be sufficient for some displays [55].

3. Validation

We created software models for the scene, optics, sensor, processor, and display in an integrated suite of MATLAB software tools, the ISET [26]. 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 the next section, we use ISET to model the scene, optics, and sensor of a calibrated 5 megapixel CMOS digital camera and compare the simulated and measured sensor performance.

A. Simulation Parameters

The simulation parameters were derived from a few fundamental measurements that characterize sensor spectral sensitivity and electrical properties including dark current, read noise, DSNU, and photoreceptor nonuniformity. While we estimated these parameters from a modest set of calibration measurements (see Appendices A and B), other parameters, such as conversion gain and voltage swing, were provided by the sensor manufacturer.

We modeled the camera lens using a diffraction-limited model [56] with a lens *f*-number of 4 and a focal length of 3 mm. An optical diffuser that filters out signals above the Nyquist frequency limit of the imaging sensor was simulated using a Gaussian filter with full width at half-maximum equal to the pixel width.

Table 1 lists the sensor parameters along with the reference source for the data. 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, or blue pixel, respectively (Fig. 1). Appendix B describes the laboratory measurements and calculations used to estimate these spectral curves.

B. Simulation Performance

To evaluate the quality of the simulation, we first measured the radiance image of a Macbeth ColorChecker illuminated by a tungsten light using the methods described in Subsection 2.A. We then used the 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 2 compares the predicted and simulated sensor images of the Macbeth ColorChecker after they have been 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 pixel values for the 24 color patches in the sensor images of the Macbeth ColorChecker. A scatter plot of the signal-to-noise ratio (SNR, mean divided by the standard deviation) in the measured and simulated images also show good agreement for each of the 24 patches in the Macbeth Color Checker (Figs. 3 and 4).

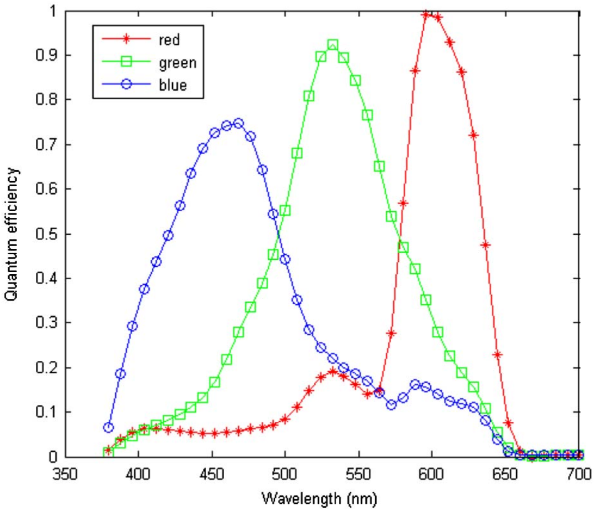


Fig. 1. (Color online) Spectral quantum efficiencies of red, green, and blue channels for a calibrated imaging sensor.

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 Figs. 5 and 6, respectively. In this example, we render a sensor image of a Macbeth ColorChecker. To perform the rendering, we specify the linear sRGB [58] display values of an image that match a Macbeth ColorChecker under D65 illumination. These are the

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	gbrg	manufacturer
Spectral quantum efficiencies	—	measured (see Appendix B)
Dark voltage (V)	4.68 mV/s	measured (see Appendix A)
Read noise (mV)	0.89	measured (see Appendix A)
DSNU (mV)	0.83	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 (μV/e)	2.0000e – 004	manufacturer
Analog gain	1.0	software setting
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 [57]

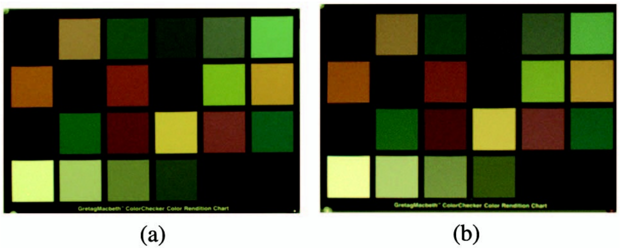

(a) (b)

Fig. 2. (a) Measured and (b) simulated sensor images. The images have been demosaicked but not color balanced.

desired sRGB values. Then, we find the 3×3 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. 5) and simulated sensor images (Fig. 6). Histograms of the CIELAB color difference error (ΔE) are also shown. Again, the simulation predicts the color accuracy of the sensor quite well.

Figures 2–6 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 Figs. 3 and 4). And both simulated and measured sensor images have comparable color accuracy (see Figs. 5 and 6).

4. Discussion

We described 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 laboratory measurements conform well to the predictions from the simulation.

Two independent studies also show that simulation accurately models real systems [30,59]. These studies simulated imaging sensors in development and reported a close correspondence between the distribution of pixel values for measured and simulated images of the Macbeth ColorChecker. One investigation [30] used the ISET ray-trace modeling tools to simulate the effects of the lens on the sensor irradiance image. The authors of this study compared simulated and actual sensor images of a spatial resolution chart and calculated the system modulation transfer function (MTF) using the ISO 12233 method implemented in ISET. They report that the MTF for the simulated sensor was slightly higher than the MTF for the measured sensor. The authors explain this difference by noting that they had more control over the focus setting for a simulated than for a measured sensor. Given the results of their study, the authors conclude that the simulator "is capable of generating results that are close to those of the actual product" and that they now use the simulator "for commercial image sensor design."

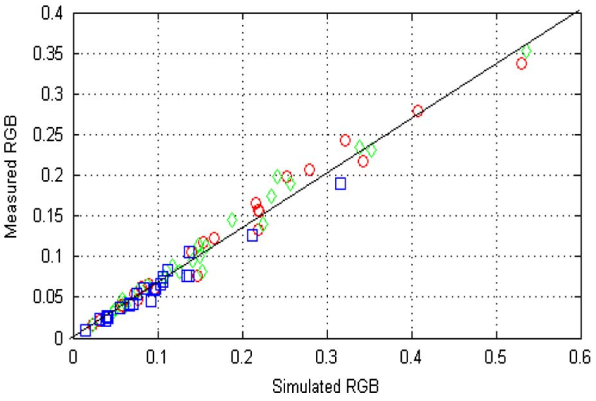


Fig. 3. (Color online) Mean pixel values averaged for each of the 24 patches in the Macbeth ColorChecker in the measured sensor images plotted against mean pixel values for the same patches in the simulated sensor images.

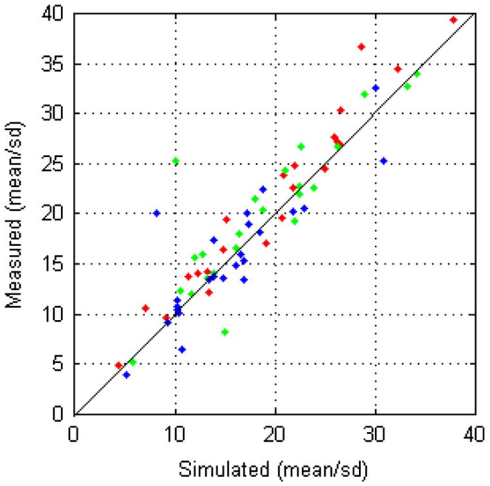


Fig. 4. (Color online) Mean divided by the standard deviation (SNR) in pixel values for each of the 24 patches in the Macbeth ColorChecker in the measured sensor images plotted against standard deviation in pixel values for the same patches in the simulated sensor images.

The complexity of the digital imaging pipeline, coupled with a moderate array of image quality metrics, limits our ability to offer closed-form mathematical solutions to design questions. In such cases, simulation technology can be a helpful guide for engineers who are selecting parts, designing algorithms, and inventing new imaging sensors. Simulation of image system performance complements and extends explorations using specific hardware prototypes [60] in several ways.

First, creating and calibrating images to test components is essential for system design and evaluation; obtaining an adequate range of calibrated test images 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 (high dynamic range, low light levels, and so forth). For example, to study the trade-off between pixel size and light sensitivity [31], we used the ISET simulator to parametrically vary scene light levels and sensor pixel

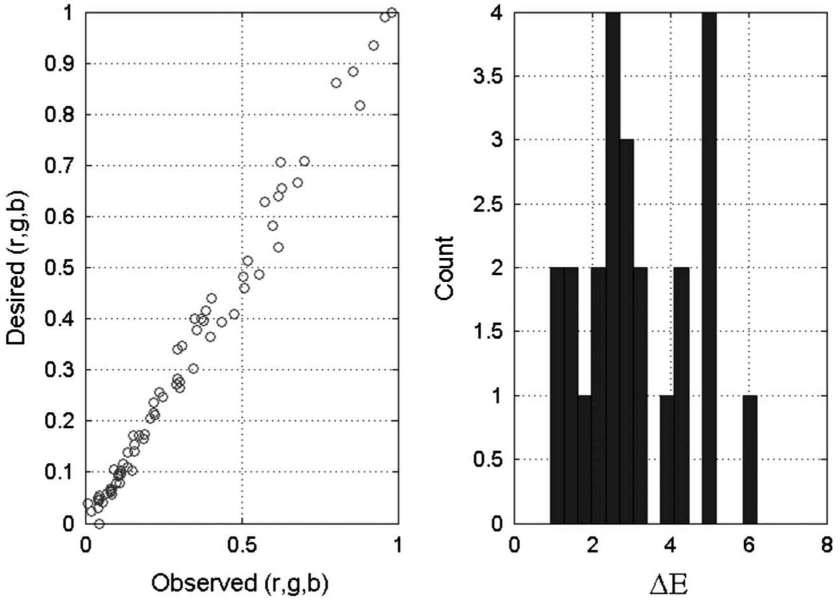
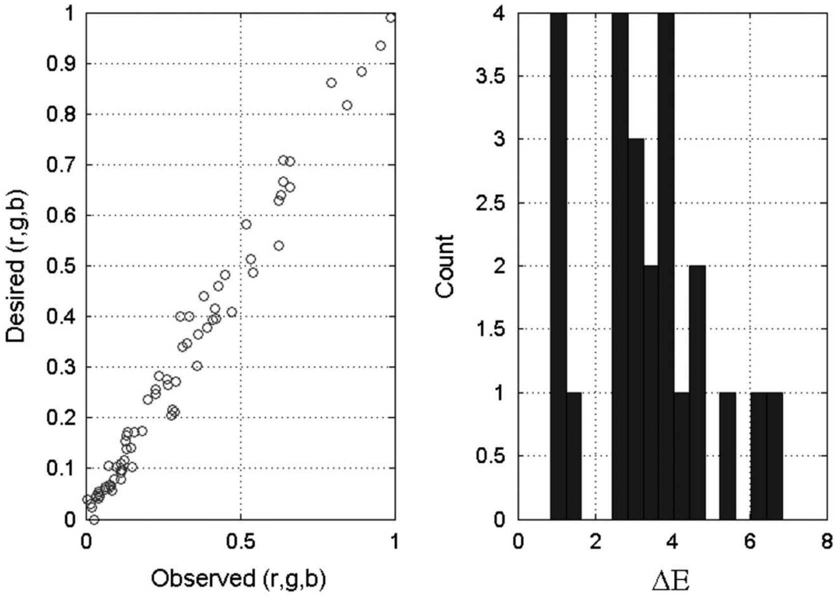


Fig. 5. Color accuracy of measured imaging sensor. The graph on the left plots the desired sRGB values against the color-balanced sRGB values derived from the measured sensor images. The histograms show the distribution of ΔE color errors for the 24 color patches in the Macbeth ColorChecker.

size. And to characterize the effects of camera motion [36] on image quality, we varied scene intensity, pixel size, exposure duration, and camera motion.

Second, simulators make it possible to model and evaluate imaging components that are difficult or even impossible to manufacture with current technologies. This includes novel pixel designs with integrated color filters based on nanopatterned metal layers [61], transverse field detectors with pixel electrically tunable spectral sensitivities [62], and imaging sensors with novel color pixel arrays [33,63,64], as well as sensors for other applications, including microscopy, endoscopy [65], and high-speed document sensing [66].

Third, a simulator can enhance communication and collaboration among people with different types of expertise and at different locations. For example, an engineer working on one part of the system—say, demosaicking—need not be familiar with the physical simulation of the sensor itself. Similarly, the



(a) (b)

Fig. 6. Color accuracy of simulated imaging sensor. (a) plots the desired sRGB values against the color-balanced sRGB values derived from the simulated sensor images. The histograms in (b) show the distribution of ΔE color errors for the 24 color patches in the Macbeth ColorChecker.

engineer working to reduce circuit noise need not be an expert in demosaicking algorithms. A simulator provides both users with a framework that implements the less familiar parts of the pipeline. Different laboratories can communicate about a sensor by simply sending the data files that characterize the sensor to one another.

Finally, a simulator can produce fully rendered images so that the user can see the effects of the parameter changes. The combination of visualization and quantification increases the user's confidence that sensible design and evaluation decisions will be made.

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 (DVs) over time is proportional to the dark voltage. The dark voltage (volts/second) 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 darsk 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 nonuniformity (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.

Photoresponse nonuniformity (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 that 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 [67]

Figure 7 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. 7(a)]. 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 and M is the number of spectral lights.

Second, place the camera at the same location used for the spectrophotometer measurement [Fig. 7(b)]; 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

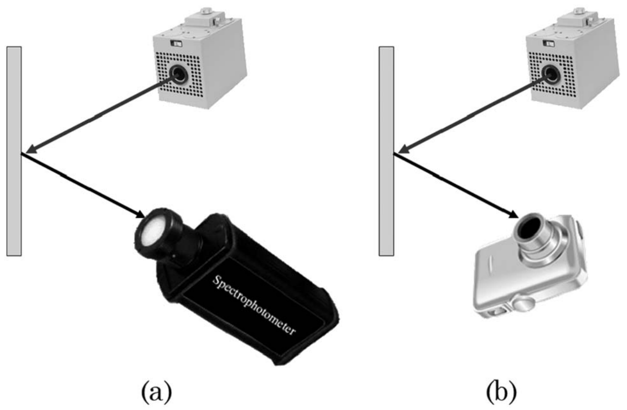


Fig. 7. (a) Illuminate a Lambertian surface with narrowband lights spanning 400–800 nm using a monochromator. Measure the spectral radiance of each of the narrowband lights using a spectrophotometer. (b) Place the camera at the same location used for the spectrophotometer measurement and capture an image of each of the narrowband lights.

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 $C = S'L$.

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.

We gratefully acknowledge support from Canon, Hewlett Packard, Logitech International, Microsoft Corporation, Nikon Corporation, Olympus Corporation, Philips Semiconductors, and the Samsung Advanced Institute of Technology. We thank Professor Bernd Girod for supporting this work. We thank Norihiro Aoki from Nikon for providing us with digital cameras and lenses, Jeffrey Mackey from Aptina for sensor reference design kits, and Francisco Imai, formerly from Samsung, for calibration equipment. We thank our colleagues who have contributed to this work, including David Brainard, Ting Chen, Jeff DiCarlo, Keith Fife, Abbas El Gamal, Kunal Ghosh, Francisco Imai, Sam Kavusi, Max Klein, Peter Maeda, Ricardo Motta, Mike Okincha, Amnon Silverstein, Poorvi Vora, Feng Xiao, and Xuemei Zhang.

References and Notes

1. C. Kolb, D. Mitchell, and P. Hanrahan, “A realistic camera model for computer graphics,” in *SIGGRAPH ’95 Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques* (ACM, 1995), pp. 317–324.
2. M. Potmesil and I. Chakravarty, “A lens and aperture camera model for synthetic image generation,” in *SIGGRAPH ’83 Proceedings of the 10th Annual Conference on Computer Graphics and Interactive Techniques* (ACM, 1983), Vol. **15**, No. 3, pp. 297–305.
3. B. Barsky, D. R. Horn, S. A. Klein, J. A. Pang, and M. Yu, “Camera models and optical systems used in computer graphics: Part I, object based techniques,” in *Proceedings of the 2003 International Conference on Computational Science and its Applications (ICCCSA ’03)*, Lecture Notes in Computer Science (Springer-Verlag 2003), pp. 246–255.
4. R. Gow, D. Renshaw, and K. Findlater, “A comprehensive tool for modeling CMOS image-sensor-noise performance,” IEEE Trans. Electron Devices **54**, 1321–1329 (2007).
5. P. E. Haralabidis and C. Pilinis, “Linear color camera model for a skylight colorimeter with emphasis on the imaging pipeline noise performance,” J. Electron. Imaging **14**, 043005 (2005).
6. R. Constantini and S. Susstrunk, “Virtual sensor design,” Proc. SPIE **5301**, 408–419 (2004).
7. H. B. Wach and J. E. R. Dowski, “Noise modeling for design and simulation of computational imaging systems,” Proc. SPIE **5438**, 159–170 (2004).
8. B. F. A. El Gamal, H. Min, and X. Liu, “Modeling and estimation of FPN components in CMOS image sensors,” Proc. SPIE **3301**, 168–177 (1998).
9. B. F. H. Tian and A. El Gamal, “Analysis of temporal noise in CMOS APS,” IEEE J. Solid-State Circuits **36**, 92–101 (2001).
10. B. A. Wandell, “The synthesis and analysis of color images,” IEEE Trans. Pattern Anal. Machine Intell. **9**, 2–13 (1987).
11. L. T. Maloney, “Evaluation of linear models of surface spectral reflectance with small numbers of parameters,” J. Opt. Soc. Am. A **3**, 1673–1683 (1986).
12. D. H. Marimont and B. A. Wandell, “Linear models of surface and illuminant spectra,” J. Opt. Soc. Am. A **9**, 1905–1913 (1992).
13. S. Tominaga and B. A. Wandell, “Component estimation of surface spectral reflectance,” J. Opt. Soc. Am. A **7**, 312–317 (1990).
14. B. K. P. Horn and R. W. Sjoberg, “Calculating the reflectance map,” Appl. Opt. **18**, 1770–1779 (1979).
15. D. B. Judd, D. L. MacAdam, and G. W. Wyszecki, “Spectral distribution of typical daylight as a function of correlated color temperature,” J. Opt. Soc. Am. **54**, 1031–1036 (1964).
16. J. P. S. Parkkinen, J. Hallikainen, and T. Jaaskelainen, “Characteristic spectra of Munsell colors,” J. Opt. Soc. Am. A **6**, 318–322 (1989).
17. J. Cohen, “Dependency of the spectral reflectance curves of the Munsell color chips,” Psychonomic Sci. **1**, 369–370 (1964).
18. B. Gunturk, J. Glotzbach, Y. Altunbasak, R. Schafer, and R. M. Mersereau, “Demosaicking: color filter array interpolation,” IEEE Signal Process. Mag. **22**, 44–54 (2005).
19. J. P. Kerekes and J. E. Baum, “Spectral imaging system analytical model for subpixel target detection,” IEEE Trans. Geosci. Remote Sens. **40**, 1088–1101 (2002).
20. M. Bernhardt, P. Clare, C. Cowell, and M. Smith, “A hyperspectral model for target detection,” Proc. SPIE **6565**, 65650F (2007).
21. J. R. Schott, *Remote Sensing: The Image Chain Approach*, 2nd ed. (Oxford University, 2007).
22. R. D. Fiete, *Modeling the Imaging Chain of Digital Cameras Tutorial Texts in Optical Engineering* (SPIE Press, 2010), Vol. TT92.
23. X. Wang, J. Zhang, Z. Feng, and H. Chang, “Equation-based triangle orientation discrimination sensor performance model based on sampling effects,” Appl. Opt. **44**, 498–505 (2005).
24. J. Farrell, M. Okincha, M. Parmar, and B. Wandell, “Using visible SNR (vSNR) to compare the image quality of pixel binning and digital resizing,” Proc. SPIE **7537**, 75370C (2010).
25. J. Xu, R. Bowen, J. Wang, and J. Farrell, “Visibility of uncorrelated image noise,” Proc. SPIE **7537**, 753703 (2010).
26. J. Farrell, F. Xiao, P. B. Catrysse, and B. Wandell, “A simulation tool for evaluating digital camera image quality,” Proc. SPIE **5294**, 124–131 (2004).
27. P. L. Vora, J. E. Farrell, J. D. Tietz, and D. H. Brainard, “Image capture: simulation of sensor responses from hyperspectral images,” IEEE Trans. Image Process. **10**, 307–316 (2001).
28. P. Longere and D. H. Brainard, “Simulation of digital camera images from hyperspectral input,” in *Vision Models and Applications to Image and Video Processing*, C. v. d. B. Lambrecht, ed. (Kluwer, 2001), pp. 123–150.
29. P. B. Catrysse and B. A. Wandell, “Optical efficiency of image sensor pixels,” J. Opt. Soc. Am. A **19**, 1610–1620 (2002).
30. J. Chen, K. Venkataraman, D. Bakin, B. Rodricks, R. Gravelle, P. Rao, and Y. Ni, “Digital camera imaging system simulation,” IEEE Trans. Electron Devices **56**, 2496–2505 (2009).
31. J. Farrell, M. Okincha, and M. Parmar, “Sensor calibration and simulation,” Proc. SPIE **6817**, 68170R (2008).
32. J. Farrell, F. Xiao, and S. Kavusi, “Resolution and light sensitivity tradeoff with pixel size,” Proc. SPIE **6069**, 60690N (2006).
33. M. Parmar and B. A. Wandell, “Interleaved imaging: an imaging system design inspired by rod-cone vision,” Proc. SPIE **7250**, 725008 (2009).
34. F. Xiao, A. Silverstein, and J. Farrell, “Camera motion and effective spatial resolution,” in *Proceedings of the International Congress of Imaging Science* (Society for Imaging Science and Technology, 2006), pp. 33–36.
35. F. Xiao, J. E. Farrell, P. B. Catrysse, and B. Wandell, “Mobile imaging: the big challenge of the small pixel,” Proc. SPIE **7250**, 72500K (2009).
36. D. H. Brainard and C. Broussard, “Rendertoolbox Wiki,” http://rendertoolbox.org.
37. B. Barsky and T. Kosloff, “Algorithms for rendering depth of field effects in computer graphics,” in ICCOMP ’08 Proceedings of the 12th WSEAS International Conference on Computers (World Scientific and Engineering Academy and Society, 2008), 999–1010.

38. D. Lanman, R. Raskar, and G. Taubin, “Modeling and synthesis of aperture effects in camera,” in *Proceedings of the International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging* (CAe, 2008), pp. 81–88.
39. R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, and P. Hanrahan, “Light field photography with a hand-held plenoptic camera,” Stanford Tech. Report CTSR 2005-02 (2005).
40. K. Martinez, J. Cupitt, and D. Saunders, “High resolution colorimetric imaging of paintings,” Proc. SPIE **1901**, 25–36 (1993).
41. P. L. Vora, M. L. Harville, J. E. Farrell, J. D. Tietz, and D. H. Brainard, “Image capture: synthesis of sensor responses from multispectral images,” Proc. SPIE **3018**, 2–11 (1997).
42. P. D. Burns and R. S. Berns, “Analysis of multispectral image capture,” in *Proceedings of the 4th IS&T/SID Color Imaging Conference* (IS&T, 1996), pp. 19–22.
43. S. M. C. Nascimento, F. P. Ferreira, and D. H. Foster, “Statistics of spatial cone-excitation ratios in natural scenes,” J. Opt. Soc. Am. A **19**, 1484–1490 (2002).
44. M. Parmar, F. Imai, S. H. Park, and J. Farrell, “A database of high dynamic range visible and near-infrared multispectral images,” Proc. SPIE **6817**, 68170N (2008).
45. P. B. Catrysse is preparing a manuscript to be called “Imaging optics.”
46. R. G. Driggers, *Encyclopedia of Optical Engineering* (CRC, 2003), Vol. 1.
47. P. B. Catrysse and B. A. Wandell, “Roadmap for CMOS image sensors: Moore meets Planck and Sommerfeld,” Proc. SPIE **5678**, 1–13 (2005).
48. W. J. Smith, *Modern Optical Engineering: the Design of Optical Systems*, 4th ed., Optical and Electro-Optical Engineering Series (McGraw-Hill, 2007).
49. J. Goodman, “The frequency response of a defocused optical system,” Proc. R. Soc. Lond. Ser. A **231**, 91–103 (1955).
50. J. W. Goodman, *Introduction to Fourier Optics*, 2nd ed. (McGraw-Hill, 1996).
51. P. Maeda, P. B. Catrysse, and B. A. Wandell, “Integrating lens design with digital camera simulation,” Proc. SPIE **5678**, 48–58 (2005).
52. P. B. Catrysse, X. Liu, and A. El Gamal, “Quantum efficiency reduction due to pixel vignetting in CMOS image sensors,” Proc. SPIE **3965**, 420–430 (2000).
53. H. Tian and A. El Gamal, “Analysis of $1/f$ noise in CMOS APS,” Proc. SPIE **3965**, 168–176 (2000).
54. J. R. Janesick, *Scientific Charge-Coupled Devices* (SPIE, 2001), Vol. PM83.
55. J. Farrell, G. Ng, X. Ding, K. Larson, and B. Wandell, “A display simulation toolbox for image quality evaluation,” J. Disp. Technol. **4**, 262–270 (2008).
56. It is 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 PSFs that are both wavelength dependent and position dependent (not shift invariant). In our simulations, we analyze the central portion of the sensor image, where departures from diffraction-limited models are less evident.
57. J.-Y. Bouguet, “Camera calibration toolbox for Matlab,” http://www.vision.caltech.edu/bouguetj/calib_doc/.
58. M. Stokes, M. Anderson, S. Chandrasekar, and R. Motta, “Standard default color space for the internet—sRGB,” http://www.w3.org/Graphics/Color/sRGB.html.
59. C. Mornet, J. Vaillant, T. Decroux, D. Herault, and I. Schanen, “Evaluation of color error and noise on simulated images,” Proc. SPIE **7537**, 75370Y (2010).
60. A. Adam, E. Talvala, S. H. Park, D. E. Jacobs, B. Ajdin, N. Gelfand, N. Dolson, J. Vaquero, J. Baek, M. Tico, H. P. Lensch, W. Matusik, K. Pulli, M. Horowitz, and M. Levoy, “The Frankencamera: an experimental platform for computational photography,” in *SIGGRAPH ’10 ACM SIGGRAPH 2010 Papers* (ACM, 2010), pp. 1–12.
61. P. B. Catrysse and B. A. Wandell, “Integrated color pixels in 0.18-μm complementary metal oxide semiconductor technology,” J. Opt. Soc. Am. A **20**, 2293–2306 (2003).
62. G. Langfelder, F. Zaraga, and A. Longoni, “Tunable spectral responses in a color-sensitive CMOS pixel for imaging applications,” IEEE Trans. Electron Devices **56**, 2563–2569 (2009).
63. S. Lansel and B. Wandell, “Local linear learned image processing pipeline,” in *Imaging Systems Applications*, OSA Technical Digest (CD) (Optical Society of America, 2011), paper IMC3.
64. F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar, “Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum,” IEEE Trans. Image Process. **19**, 2241–2253 (2010).
65. K. K. Ghosh, L. D. Burns, E. D. Cocker, A. Nimmerjahn, Y. Ziv, A. El Gamal, and M. J. Schnitzer, “Miniaturized integration of a fluorescence microscope,” Nat. Methods **8**, 871–878 (2011).
66. G. Leseur, N. Meunier, G. Georgiadis, L. Huang, B. Wandell, P. B. Catrysse, and J. M. DiCarlo, “High-speed document sensing and misprint detection in digital presses,” Proc. SPIE **7536** (2010).
67. J. M. DiCarlo, E. Montgomery, and S. W. Trovinger, “Emissive chart for imager calibration,” in *Twelfth Color Imaging Conference: Color Science and Engineering Systems* (The Society for Imaging Science and Technology, 2004), pp. 295–301.