# Image Systems Simulation

***Joyce E. Farrell and Brian A. Wandell***
**Stanford University, California, USA**

## Abstract

*We describe a suite of software tools for image system simulation. The tools model the three-dimensional scene radiance, image formation by the optics, sensor transduction, image processing and display rendering.*

### Keywords

Simulation; digital cameras; displays; image quality.

## 1. Introduction

Imaging systems are designed to both capture and display information. Such systems comprise multiple components that impact performance and quality. Engineers typically evaluate individual system components, but customers judge performance by viewing the final output that depends on how the components work together. Consequently, understanding components in isolation, without accounting for the coordination of the system components, provides a limited assessment. A simulation environment can provide the engineer with useful tools that clarify how the system components work together to produce the final output.

Image systems simulation software should help the engineer improve the critical system components and provide opportunities to experiment with new designs and components. With these goals in mind, we developed the Image Systems Engineering Toolbox (ISET), an integrated suite of Matlab data structures and functions that can be used to characterize the variety of imaging components and simulate how they transform data along the imaging pipeline [1, 2].

ISET organizes the data structures into software objects corresponding to the scene, optics, sensor, processor, and display (Figure 1). The *scene* object is a radiometric description of the input data. The *optics* object defines the lens properties that convert the scene into an irradiance image at the sensor surface. The *sensor* object defines the properties of the pixels and sensor array that govern how the irradiance image is converted into electrons. The *image processor* (*IP*) object is a collection of algorithms that define how sensor data are transformed into display values. The *display* object is a radiometric description of the final image for any calibrated display.

The ISET functions act on these objects and their associated data. For example, there are functions that combine the scene and optics objects to calculate the sensor irradiance. Other functions combine the irradiance with the sensor object to calculate the sensor voltages. Additional functions implement image-processing algorithms, model display rendering and calculate the spectral and spatial properties of the displayed image.

## 2. Imaging Systems

The simulation software aims to follow two key principles:

1. Model the complete image processing pipeline
2. Use meaningful physical units

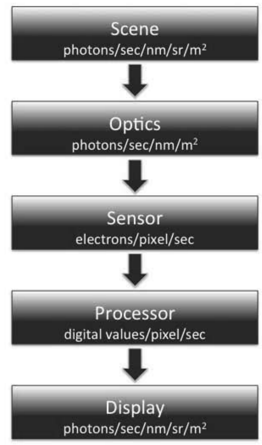


**Figure 1.** An image systems simulation environment. The software is organized around objects and associated data representing the scene, optics, sensor, processor and display.

The two principles are suggested in the flow chart in Figure 1. This chart shows key objects and the general units associated with important data attached to these objects. The need for a physically meaningful representation differs from the requirement in many computational photography and image coding applications, where knowledge of the digital image values is often sufficient. Image systems simulations must predict the effects of a lens, the responses in a sensor, or the image quality of a display. To do this, one must know the spectral radiance and irradiance and in some instances the spectral light field [3-5].

### *Scene representation*

A complete scene radiometric description, $L$, describes the rate of photons in every scene position ($x$, $y$, $z$), direction ($\theta$, $\phi$), wavelength ($\lambda$), polarization, and moment in time [3-5]. Ignoring polarization, the units of radiance are normalized per time interval (s), solid angle (sr), and area of the point source (photons/s/nm/sr/m$_2$).

But not every simulation requires the full scene description, and the complexity of the scene representation should match the simulation goals. A simple representation of scene radiance, $L(x, y, \lambda)$, is sufficient to predict the visibility of blur, noise, or color differences. A more complex representation, $L(x, y, z, \theta, \phi, \lambda)$,

 *ISSN 0097-966X/15/4501-0180-$1.00* 

is necessary to analyze transparency, depth of field, and synthetic apertures.

It is possible, though challenging, to measure natural scene radiances using hyperspectral and multispectral imaging methods, and several datasets can be downloaded from the web [6-9]. These data can provide insights about the typical dynamic range and spectral characteristics of likely scenes. The data size of the spectral representations can be substantial, particularly when both the surface and illuminant vary across the scene. The size of the spectral data can be reduced, without much loss in precision, by using linear models with a modest (6-8) number of basis functions [10].

### Image formation

In typical applications, the imaging optics converts scene radiance into an irradiance image at the sensor surface. The irradiance represents the sum of rays across all angles (photons/sec/nm/m$^2$). But for modeling light field cameras it is necessary to represent the intensity of rays at multiple angles.

Just as there are useful simplifications of the scene radiance, optics simulation can be carried out with varying degrees of complexity. To compute the optical irradiance we account for a number of factors. The minimum requirement is to account for the lens aperture, focal length, magnification, and lens shading (the fall-off in intensity with lens field height). More complex models specify geometric distortions and a wavelength-dependent but shift-invariant point spread function. The most complex models also incorporate the wavelength- and 3D space-varying point spread function.

### Sensor transduction

ISET uses the concept of a generalized sensor, a software object that includes descriptions of several system components that are integrated with the sensor chip. The essence of the generalized sensor is a *phenomenological model* of how irradiance is converted to sensor outputs (voltage). A phenomenological model is a set of mathematical formula that predicts properties, such as light sensitivity, spatial and temporal integration, fixed and temporal noise, and circuit properties (e.g., quantization). The parameters of the model depend on system components that are coupled closely coupled to the sensor - the color filter array, anti-aliasing filter, and microlens array - although these components are not part of the chip itself. The generalized sensor object includes the image system components whose properties are essential for an accurate prediction of the sensor response to any irradiance image.

In most digital image sensors, the transduction of photons to electrons is approximately linear: specifically, the photodetector response (either CCD or CMOS) increases linearly with the number incident photons up to saturation. 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/m$^2$) 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

$$e_i = T \iint_{\lambda,x} S_i(\lambda) A_i(x) I(\lambda,x) d\lambda dx \qquad (1)$$

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. For a more complete description see Ref. [5].

### Image processing

The **processor** module converts the sensor voltages in the two-dimensional sensor array into an RGB image that can be rendered on a specified display. The image systems pipeline includes algorithms to control exposure duration, interpolate missing RGB sensor values (demosaicking) and transform sensor RGB values into an internal color space for encoding and display (color-balancing and display rendering). There are many different approaches to auto-exposure, demosaicking and color balancing, and these are often proprietary. ISET includes basic algorithms that are in the public domain.

### Display rendering

ISET uses three functions to predict the spatial-spectral radiance emitted by a display. First, a look-up table that summarizes the display transduction (gamma function) converts digital values into a measure of the linear intensity. Second, pixel point spread functions for each color component (sub-pixel point spread function) are used to generate a spatial map of linear intensity for each of the display primaries. By modifying the point spread functions it is possible to model displays with unusual pixel layouts, such as RGBW, RGBG and PenTile patterns [11]. Third, the spectral power distributions of the primaries are used to calculate the spectral composition of the displayed image. These three functions – the display gamma, the sub-pixel point spread functions (psf) and the spectral power distributions (spd) of the primaries – are sufficient to characterize the performance of displays with independent pixels [12].

Accounting for these three properties of the display provides a practical starting point for many display simulations [12-13], although they may not be sufficient for some displays [14]. One of the values of the display simulation is to help engineers design and evaluate sub-pixel rendering algorithms for novel color displays without requiring a physical display prototype.

## 3. Summary

There are several ways in which simulation software can guide the design of imaging systems and lead to innovative solutions. First, simulation software can enhance communication and collaboration between people with different expertise and at different locations. As an example, consider an engineer who is inventing new image processing algorithms. Software simulation enables the person to generate inputs whose properties accurately reflect the expected inputs in a wide range of circumstances. These simulations provide meaningful data even though the imaging hardware is incomplete or unavailable because it is being developed at other locations and companies.

Second, simulation software can be used to create calibrated test images that are valuable for evaluating performance. The engineer can simulate sensor data over a wide range of conditions that are difficult to create in the laboratory (high dynamic range, low light levels, and so forth). For example, to characterize the tradeoff between spatial resolution and light sensitivity, we used the ISET simulator to parametrically vary scene light levels and sensor pixel size [15]. To characterize the effects of camera motion on spatial resolution, we varied scene intensity, pixel size, exposure duration, and camera motion [16]. It would have been very difficult, or even impossible, to systematically vary and measure all of these parameters.

Third, simulation software can model the integration of imaging components that are difficult or even impossible to manufacture with current technologies. For example, ISET has been used to design pixels with integrated color filters based on nano-patterned metal layers [16-17], transverse field detectors with pixels electrically tunable spectral sensitivities [18-19], and imaging sensors with novel color pixel arrays [20-22]. These simulations led to inventions that are now being developed in academic and industry laboratories.

There is a great deal of room for development of image systems simulation software. There are opportunities to expand simulations to include systems that use multiple sensors, and even sensors of different types, such as specialized components that measure depth, motion, and location. As imaging systems become more complex and more deeply integrated with other technologies, such as automobiles, the need for simulations will continue to grow.

## 4. Acknowledgements

We thank David Brainard, Peter Catrysse, Haomiao Jiang , Andy Lai Lin and Feng Xiao for their contributions to our work on image systems simulation software. We thank Henryk Blasinski, Sam Kavusi, Mike Okincha, Manu Parmar, Steve Lansel, and Qiyuan Tian for demonstrating the efficacy of ISET simulations for many different types of imaging applications.

## 5. References

[1] Farrell, J. E., Xiao, F., Catrysse, P.B., and Wandell B.A. (2014) *A simulation tool for evaluating digital camera image quality.* Proc. SPIE, **5294**: p. 124-131.

[2] Farrell, J.E., Catrysse, P.B. and Wandell. B. A. (2012) *Digital camera simulation.* Journal of Applied Optics, **51**(4).

[3] Gershun, A. *The Light Field*. (1936) Translated by P. Moon and G. Timoshenko in Journal of Mathematics and Physics **18**, pp. 55 - 151, 1939. p. 51-151.

[4] Adelson, E.H. and Bergen, J. R. (1991) *The Plenoptic Function and the Elements of Early Vision*, in *Computational Models of Image Processing*, M.L.a.J.A. Movshon, Editor. 1991, MIT Press: Cambridge, MA. p. 3 - 20.

[5] Farrell, J.E. and Wandell, B.A. (2015) *Image Systems Simulation*, in Handbook of Digital Imaging, Volume 1 (ed Michael Kriss), John Wiley & Sons, Ltd., Chichester, pp. 373–400.

[6] Skauli, T. and Farrell, J. E. (2013) *A collection of hyperspectral images for imaging systems research. Proc. SPIE* **8660**,; doi:10.1117/12.2007097, https://scien.stanford.edu/index.php/hyperspectral-image-data/

[7] Parmar, M., Imai, F., Park, S. H. and Farrell, J. E. (2008) A database of high dynamic range visible and near-infrared multispectral images. Proc. SPIE **6817**; doi: 10.1117/12.767830 http://imageval.com/scene-database/

[8] Foster, D.H., Nascimento, S.M.C. , and Amano, K. (2004) *Information limits on neural identification of coloured surfaces in natural scenes.* Visual Neuroscience, **21**: p. 331-336. http://personalpages.manchester.ac.uk/staff/d.h.foster/Hyperspectral_images_of_natural_scenes_04.html

[9] F. Yasuma, Mitsunaga, T., Iso, D. and Nayar, S.K. (2008) *Generalized Assorted Pixel Camera: Post-Capture Control of Resolution, Dynamic Range and Spectrum,* Technical Report, Department of Computer Science, Columbia University CUCS-061-08, http://www.cs.columbia.edu/CAVE/databases/multispectral

[10] Marimont, D.H. and Wandell, B. A. (1992) *Linear models of surface and illuminant spectra.* Journal of the Optical Society of America A, **9**(11): p. 1905-1913.

[11] Elliott, C. H. Brown, Credelle, T. L., Han, S., Im, M. H., Higgins, M. F. and Higgins, P. (2003) *Development of the PenTile Matrix™ color AMLCD subpixel architecture and rendering algorithms*, Journal of the Society for Information Display, **11**(1), pp. 89-98.

[12] Farrell, J.E., Ng, G., Ding, X., Larson, K. and Wandell, B. A. (2008) *A display simulation toolbox for image quality evaluation*, IEEE/OSA Journal of Display Technology, **4**(2), pp. 262-270.

[13] Cooper, E. A., Jiang, H., Vildavski, V., Farrell, J. E. and Norcia, A. M. (2013), *Assessment of OLED displays for vision research*, Journal of Vision, **13**(16), doi: 10.1167/13.12.16

[14] Lyons, N. P and Farrell, J. E. (1989) *Linear systems analysis of CRT displays*, SID Digest, **10**, pp. 220-223.

[15] Farrell, J., Xiao, F. and Kavusi, S. *Resolution and light sensitivity tradeoff with pixel size.* Proc, SPIE., **6069**.

[16] Xiao, F., Farrell, J. E., Catrysse, P. B. and Wandell, B. A. (2009) *Mobile Imaging: The Big Challenge of the Small Pixel.* Proc. SPIE, **7250**.

[17] Catrysse, P.B., Wandell, B. A., and El Gamal, A. (2001) *An Integrated Color Pixel in 0.18 mu m CMOS Technology*. Technical Digest International Electron Devices Meeting, pp. 24 4 1–4.

[18] Catrysse, P.B. and Wandell, B.A. (2003) *Integrated color pixels in 0.18-um complementary metal oxide semiconductor technology*. Journal of the Optical Society of America A, **20**(12), 2293–2306.

[19] Langfelder, G. (2012) *Spectrally reconfigurable pixels for dual-color-mode imaging sensors*. Journal of Applied Optics, **51**(4), 91–98.

[20] Langfelder, G., Zaraga, F., Longoni, A. (2009) *Tunable spectral responses in a color-sensitive CMOS pixel for imaging applications*. IEEE Transactions on Electron Devices, **56**(11), 2563–2569.

[21] Parmar, M. and B.A. Wandell, (2009) *Interleaved Imaging: An Imaging System Design Inspired by Rod-Cone Vision.* Proc. SPIE, **7250**

[22] Lansel, S. and Wandell, B. A, (2011) *Local linear learned image processing pipeline*, presented at the OSA Conference on Imaging Systems and Applications, Toronto, Canada

[23] Tian, Q., Lansel, S., Farrell, J.E. and Wandell, B. A. (2014) *Automating the design of image processing pipelines for novel color filter arrays: local, linear, learned (L3) method*, Proceedings of the SPIE **9023**; doi: 10.1117/12.2042565

[24] Tian, Q., Blasinski, H., Lansel, S., Jiang, H., Fukunishi, M., Farrell, J. E. and Wandell, B. A. (2015) *Automatically designing an image processing pipeline for a five-band camera prototype using the Local, Linear, Learned (L3) method,* Proc. SPIE **9404**; doi:10.1117/12.2083435

[25] Dai, S., Tian, Y. and J. Farrell, J. E. (2013) *Design, Simulation and Evaluation of Imaging Oximeter*, Proc. SPIE **8660**,

[26] Breneman, J. Blasinski, H. and Farrell, J.E. (2104) *The color of water: using underwater photography to estimate water quality*, in Proc SPIE, **9023**,

[27] Blasinski, H., Breneman, J. and Farrell, J. E. (2014) *A model for estimating spectral properties of water from RGB images*, IEEE International Conference on Image Processing (ICIP), doi 10.1109/ICIP.2014.7025122