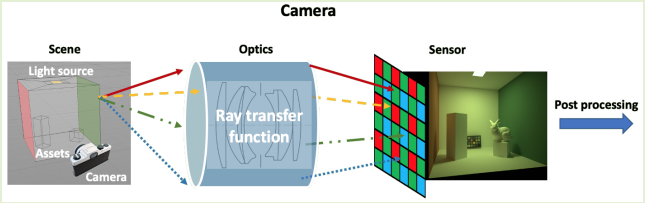
Validation of Physics-Based Image Systems Simulation With 3-D Scenes

Zheng Lyu, Thomas Goossens, Brian A. Wandell, and Joyce Farrell

***Abstract*—Image systems simulation software can accelerate innovation by reducing many of the time-consuming and expensive steps in designing, building, and evaluating image systems. To realize this potential, it is necessary to build trust in physics-based end-to-end image systems simulations. Toward this goal, we describe and experimentally validate an end-to-end physics-based image systems simulation of a digital camera. The simulation models the spectral radiance of 3-D scenes, the formation of the spectral irradiance by multielement optics, and the conversion of the irradiance to digital values (DVs) by the image sensor. We quantify the accuracy of the simulation by comparing real and simulated images of a precisely constructed, 3-D high-dynamic range (HDR) test scene.**



***Index Terms*— Computer graphics, end-to-end simulation, image systems, optics, physically based ray tracing, sensor.**

I. Introduction

COMPUTER simulation is a powerful tool for modeling and evaluating scientific ideas and engineering solutions. An image systems simulation programming environment makes it possible to understand how the system components work together to produce the final output and provides opportunities to experiment inexpensively with new designs and components.

The automotive industry is also using simulation software to generate synthetic camera image data for training neural networks for autonomous driving [1]. The advantages of such synthetic data are numerous. First, the objects in synthetic image data can be automatically labeled, eliminating the need for expensive and time-consuming hand labeling. Second, the ability to predict the images that a real camera would capture in different positions, under adverse weather conditions, and dangerous driving scenarios makes it possible to train neural networks without the risk of human injury, death, or property loss [2]. Third, image systems simulations make it possible to measure the effects that camera parameters have on the performance of neural networks [3]. Finally, simulations enable inexpensive exploration of new designs (e.g., soft prototyping) [4].

To realize the potential of image systems simulations, it is necessary to build trust in their accuracy through experimental validation and open-source software. It has been pointed out that there are very few papers that quantitatively compare real and simulated camera images of complex scenes [2]. This article starts to fill that gap by reporting on such an experiment.

We use the open-source image system engineer toolbox (ISET) software to model the complete image processing pipeline of an imaging system, including scene radiance, image formation by the optics, sensor capture, image processing, and display rendering [5], [6], [7]. These software tools were originally developed to support the codesign of the imaging components in digital cameras for consumer photography and mobile imaging [8], [9]. The software was validated by comparing simulated and real digital camera images using calibrated 2-D test charts [10], [11].

In recent years, we added the ability to use quantitative computer graphics to model 3-D scenes in the image systems simulations [12], greatly expanding the scope of applications. Using physically based ray tracing [13], we calculate the spectral irradiance image of complex, high-dynamic range (HDR) 3-D scenes. Ray-tracing algorithms capture the main effects of the lighting and inter-reflections [14], [15], [16], [17]. The use of ray tracing enables us to simulate natural scenes with complex lighting, HDR, occluding objects, and surface inter-reflections. We used these methods to design and evaluate imaging systems for a range of applications, including augmented reality/virtual reality (AR/VR) [18], underwater imaging [19], autonomous driving [20], [21], fluorescence imaging [22], [23], and the analysis of human image formation [24].

In the prior work, we carried out limited system validations by comparing a subset of the simulation measurements with real data. This article describes the first validation of the end-to-end simulation for a complex 3-D scene (Fig. 1). We make multiple quantitative comparisons between simulated and real camera images of a constructed Cornell box [14], [16].

Manuscript received 13 May 2022; revised 5 August 2022; accepted 9 August 2022. Date of publication 26 August 2022; date of current version 14 October 2022. The associate editor coordinating the review of this article and approving it for publication was Dr. Ing. Emiliano Schena. *(Corresponding author: Zheng Lyu.)*

The authors are with the Stanford Center for Image Systems Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: zhenglv.felix@gmail.com).

Digital Object Identifier 10.1109/JSEN.2022.3199699

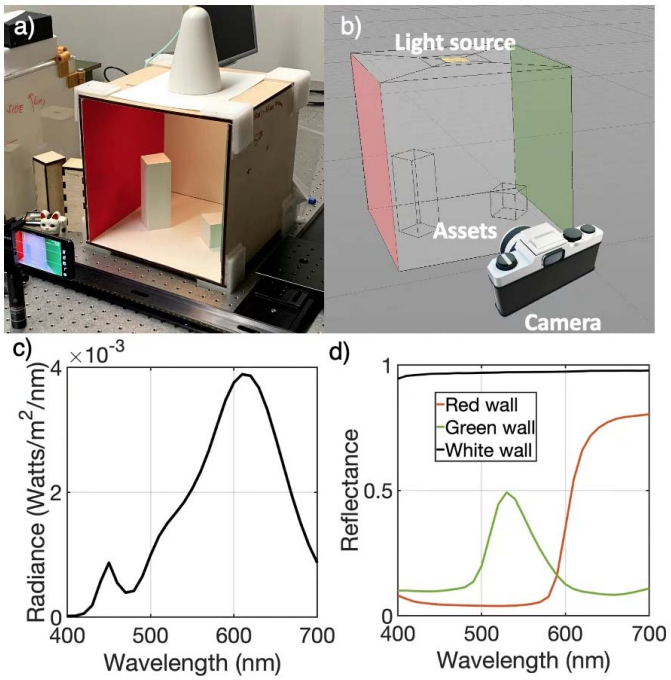


Fig. 1. Real and simulated Cornell box. (a) Image of the Cornell box and its light source in the laboratory. (b) Geometric representation of the Cornell box model and camera position in Cinema 4-D. (c) Measured spectral power distribution of the light source. (d) Measured spectral reflectance of the white, red, and green surfaces.

This box was designed to include important features found in complex natural scenes, such as surface-to-surface interreflections and shadows. In its original use, the Cornell box was used to judge the visual similarity between a real constructed box and computer graphics renderings of the box displayed on a cathode-ray tube (CRT). Here, we use the Cornell box to validate an end-to-end simulation that includes a physical description of the 3-D scene, optics, and the digital outputs of an electronic image sensor.

We provide a link to the open-source software that was used to generate the simulated camera images, along with the real camera image data that were used in this study [7]. The software contains the complete set of parameters used to generate the simulations. By sharing the software, we make it possible to check our work and reproduce the results we report.

II. Related Work

Our approach is similar to the end-to-end simulations that were used for customizing imaging systems for remote sensing [25], [26]. In remote sensing, image systems simulation is also referred to as end-to-end simulation [27], [28] and also as image chain analysis [29]. These terms emphasize the importance of evaluating individual imaging components in the context of the complete image systems.

Garnier *et al.* [30], [31] describe an end-to-end image systems simulation for infrared (IR) imaging. Their simulation traces rays from a scene through an optics model and to a sensor array. The optics model combines thin-lens equations and blurring kernels applied to the sensor image. The computational concepts are similar to the methods we use, and their paper is a good overview of the physics of image formation and sensor capture. Our effort differs from their work in several ways. First, we provide a solution to the problem, noted by Garnier *et al.* [30], [31], that lens manufacturers do not provide proprietary lens designs. This makes it impossible to trace rays through the lens. Our solution involves using data from a black box lens model, often provided by the lens manufacturer, to generate a polynomial ray-transfer function (RTF) that maps rays entering a lens, at any position and angle, into rays exiting the lens [32] (see Section III-B). This solution eliminates inaccuracies caused using the shift-invariant, wavelength-dependent, point spread functions to the postprocessed camera image [24]. Second, we simulate multiple color channels in the visible range. Third, we provide a more complete sensor model. Fourth, we quantitatively evaluate the accuracy of a simulation platform with real devices (Garnier *et al.* [30], [31] listed this evaluation goal as future work). Finally, the software that we use for our simulations is open source and freely available through GitHub [5], [6], [7]. We are unable to locate the Garnier *et al.*'s software.

Another simulation platform (the "digital imaging and remote sensing image generation (DIRSIG)") was developed by Schott *et al.* [26]. Their platform was designed to support remote sensing applications, again in the IR domain. The system includes a ray tracer with a physically realistic material reflection model. Verification and validation studies [33], [34], [35] have focused on evaluating the accuracy of the spectral emission from the scene and comparing the performance of algorithms that use synthetic and real data. Less attention is paid to modeling the lens and sensor in an imaging system, and consequently, there are no quantitative validation studies that include these components. DIRSIG [36] is not open source, but it is available to U.S.-government employees and associates who are granted a license from the Rochester Institute of Technology (RIT).

An alternative approach to ray tracing is to use raster-based computer graphics to create synthetic camera images for autonomous driving applications (e.g., [1], [37], [38], [39], [40], [41], [42], [43]). These platforms take advantage of the software rendering packages that were built to create realistic appearing computer games and driving simulators. These raster-based computer graphics run in real time, because they use various approximations that make them less computationally expensive.

Because our focus is on system design and evaluation, our approach differs in multiple ways. First, we represent the spectral properties of lights and surfaces in a 3-D scene. Second, we use path-tracing software to model how light travels through the scene, including multiple bounces between surfaces in the scene. Third, we simulate the entire path through realistic camera lenses. Fourth, we model the spectral, spatial, and temporal properties of imaging sensors to calculate how each pixel converts photons into electrons, including a model of photon noise and electronic sensor noise.

These differences are significant for computer vision applications. Tsirikoglou *et al.* [44] and Zhang *et al.* [45] reported that networks trained on synthetic camera images generated using ray tracing performed better than networks that were trained on synthetic camera images generated by raster-based methods. Liu *et al.* [3] extended this work by including simulations of digital cameras and demonstrating that networks trained on their synthetic camera images perform almost as well as neural networks trained on real camera images. The fact that neural networks trained on synthetic

camera images generalized to real camera images supports the use of image systems simulation for autonomous driving applications.

There is one report that compares simulated and real camera images for advanced driving assistance systems (ADASs). Grapinet *et al.* [46] and Gruyer *et al.* [47] used a commercially available software platform (Pro-SiVIC) that combines Open Graphics Library (Open-GL) raster-based computer graphics with some ray-tracing extensions. An important difference between their work and ours is that they use ray tracing in the red, green, blue (RGB) domain, and the lens and sensor models are implemented as filters that are applied to the rendered RGB image. In our work, we use ray tracing in the spectral domain to model critical physical characteristics of the scene and the light capture. Grapinet *et al.* [46] and Gruyer *et al.* [47] also report comparisons of simulated and real camera images of 2-D, but not 3-D, test scenes. In this article, we compare simulated and real camera images of a constructed and calibrated 3-D scene.

III. Simulation Pipeline Modeling

This section reviews the image systems simulation pipeline modeling. Section III describes the validation measurements. Section IV reviews the results. Sections V and VI describes ongoing and future work.

The image systems simulation validation includes three main parts: 1) a 3-D scene radiance description that models light sources, asset geometry, and material properties; 2) an optical model that maps rays from the scene onto the sensor; and 3) a model of how the irradiance at the sensor is converted into unprocessed digital values (DVs).

The validation measures how closely the simulation predicts the minimally processed sensor data obtained from the Google Pixel 4a. We focus on matching the unprocessed sensor data, because remaining components of the image processing are implemented in proprietary software.

A. Scene Construction

We constructed a Cornell box [Fig. 1(a)] and created a computer graphics model of the box [Fig. 1(b)]. The Cornell box was constructed using wood and covered with a white matte paper. We added red and green matte paper to the left and right walls, respectively. The meshes and their positions are exported as a set of text files that can be read by physically based rendering toolbox (PBRT).

We placed a light with a diffusing filter at an opening at the top of the box. We model the light source as an area light, and we model the surface materials as primarily Lambertian, but with a small specular term [48]. The spectral power distribution of the light source and the spectral reflectances of the surfaces in the Cornell box were measured using a PR670 spectroradiometer [Fig. 1(c) and (d)]. In some experiments, we added a calibrated miniature Gretag color chart, a slanted bar target, and a 3-D printed Stanford Bunny [49]. All the scene and simulation parameters are available in the source code we provide.

The Cornell box is a good compromise between simplicity and complexity: it includes a light source and a set of objects within a closed environment that includes many inter-reflections and shadows. There are only a few materials,

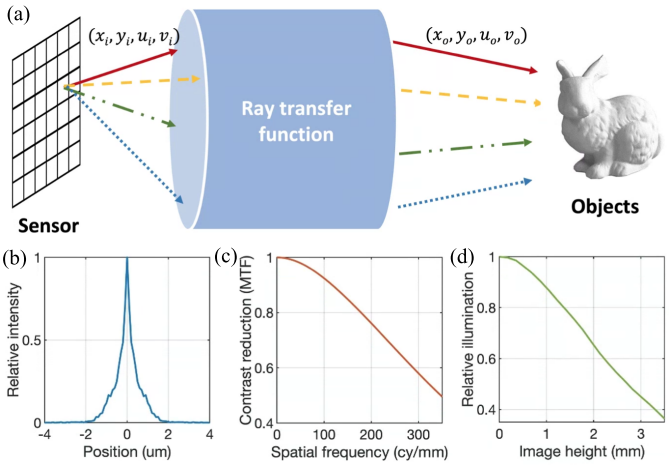


Fig. 2. Lens modeling method using a ray-transfer function. (a) We use a Zemax "black-box model" to estimate the ray-transfer function. This function describes the mapping between rays from the sensor plane that are incident at the entrance pupil, through the unknown optics, to rays in a plane after exiting from the optics (see text for details). (b)–(d) Zemax model generates an LSF, MTF, and relative illumination function, all at 550 nm. The ray-transfer function implementation in PBRT matches the Zemax calculations on these functions [32]. We evaluate the model accuracy by comparing measurements with these simulations.

making it possible to accurately model the scene. Because the light and objects within the box create significant shadows, the scene has a relatively HDR. Thus, the scene is useful for testing complex rendering, dynamic range, and sensor noise. The surfaces with different reflectances can be used to evaluate sensor chromatic responses and surface inter-reflections. For example, we were able to assess the estimated spectral quantum efficiency (QE) of the sensor by adding the Gretag color chart to the scene. Similarly, we assessed the optics model by varying the position of the slanted bar target added to the scene.

B. Optics Modeling

Optics are a critical component of cameras and determine important properties, such as lens shading (vignetting), spatial resolution, and depth of field. It is possible to use standard ray tracing to quantify these properties when the shapes, positions, and indices of refraction of the optical components are known. In some cases, including the project analyzed in this article, the lens design is proprietary.

To solve this problem, we used the approach described in [32], who point out that for ray-tracing purposes, we only need to know the "ray-transfer function" of the optics. This function maps rays entering the optics into outgoing rays. We can simulate a proprietary lens by finding an equivalent ray-transfer function using a "black box" model provided by the vendor. This model enables a customer to calculate how a ray in the incident light field (at a position and angle in the aperture) is mapped to a position and angle of a ray in the exit pupil, without revealing the lens design [Fig. 2(a)].

We calculated the polynomials that describe the Google Pixel 4a ray-transfer function. Conceptually, four input ray parameters are related to four output ray parameters by four polynomials. The positions of the incident and output rays are described by the 4-D vectors (x_i, y_i, u_i, v_i) and (x_o, y_o, u_o, v_o). The first two entries define the position of the ray in the aperture, and the second two entries define

the ray direction. We use Zemax to calculate 128 561 input–output ray pairs. We then fit a set of four polynomials relating the four input ray parameters to each of the four output ray parameters (1).

These polynomials can be calculated for each sampled wavelength

$$\begin{cases} x_o = \text{poly}_1(x_i, y_i, u_i, v_i) \\ y_o = \text{poly}_2(x_i, y_i, u_i, v_i) \\ u_o = \text{poly}_3(x_i, y_i, u_i, v_i) \\ v_o = \text{poly}_4(x_i, y_i, u_i, v_i). \end{cases} \tag{1}$$

The x, y values describe the position of the rays within the input or output aperture of the optics. The u, v values are the first two elements of the unit vector that describes the direction of the ray with respect to the input or output aperture. Only two components are needed, because the direction is a unit vector with a fixed sign for its z-component (for most lenses). As described in [32], this formulation can be simplified for the common case when the optics are rotationally symmetric, and we assumed rotational symmetry here.

The ray-transfer function can be used to calculate many summary measures of the optics, including the line-spread function (LSF), modulation transfer function (MTF), and the relative intensity map [Fig. 2(b)–(d)]. It is convenient to assess the accuracy of the model by comparing these predictions with measurements, and we do so in Section IV.

C. Sensor Modeling

The optical irradiance at the sensor surface is converted to voltages and a digital output value using a sensor model. The Google Pixel 4a uses a Sony IMX363 sensor; we summarize the sensor specifications (e.g., pixel size, fill factor, and sensor resolution) in Appendix II. Some of these values were published by Sony, and others were estimated in laboratory experiments described in the following. To perform these experiments, we used OpenCamera [50], a free and open-source software application that controls camera gain and exposure duration, to obtain the nearly unprocessed digital values from the sensor.

1) Sensor Spectral QE: To calibrate the sensor spectral QE, we captured the images of a Macbeth color checker (MCC) under three different illuminants. The spectral power distribution of the illuminants and the reflectance functions of the 24 MCC patches are plotted in Fig. 3(a) and (b). As part of the estimation, it is necessary to account for relative illumination (vignetting) effects. We did this by measuring and correcting for the relative illumination and by placing the MCC in the center of the camera image where the change in relative illumination is small.

We began the calibration by comparing the measured RGB values with those predicted using the published sensor color channel QEs and the transmissivity of an near infrared (NIR) filter. There is a substantial mismatch for this prediction which we attribute to optical and electrical crosstalk and channel gains. To account for these factors, we found a positive 3×3 matrix, M, that transforms the spectral QE. The matrix was estimated by minimizing the least square error between the measurement (r', g', b') and the original prediction (r, g, b)

$$\min_M \left\| \begin{bmatrix} r' & g' & b' \end{bmatrix} - \begin{bmatrix} r & g & b \end{bmatrix} M \right\|^2 \tag{2}$$

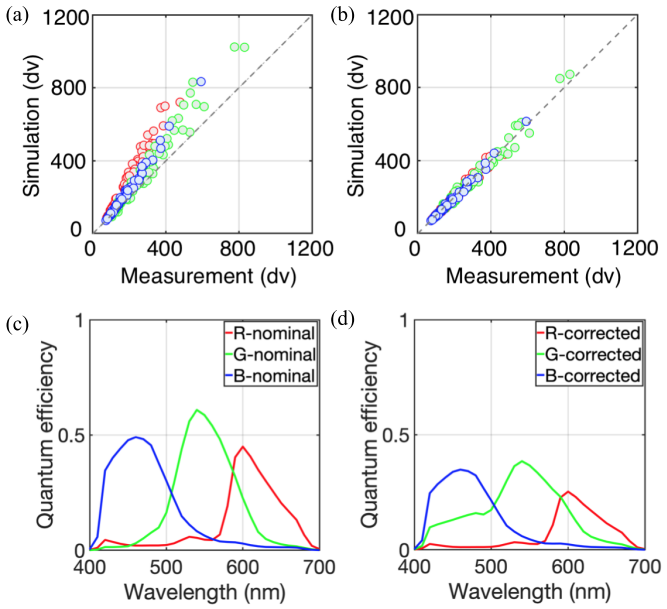


Fig. 3. Sensor spectral QE calibration. (a) Scatter plot comparing measured and simulated RGB values of the MCC based on nominal color filter QE. (b) Scatter plot comparing measured and simulated RGB values after correction for crosstalk and channel gain. (c) Nominal color filter QE and (d) QE after correction.

where $\begin{bmatrix} r', g', b' \end{bmatrix}$ and $\begin{bmatrix} r, g, b \end{bmatrix}$ are 72×3 (24 patches under three illuminants). We solve for M subject to a non-negativity constraint

$$M(i, j) \geq 0, \quad 1 \leq i, \ j \leq 3. \tag{3}$$

The fit matrix is

$$M = \begin{bmatrix} 0.5636 & 0.0807 & 0.0069 \\ 0 & 0.5917 & 0 \\ 0 & 0.2470 & 0.7098 \end{bmatrix}.$$

The diagonal entries of M are channel gains. There is one significant off-diagonal value that represents crosstalk between the blue and green channels. After incorporating the correction matrix, the simulated sensor data match the measurement [Fig. 3(d)] within a few percent (8.9%). The three spectral QE curves for the channels are shown before and after correction in Fig. 3(c) and (d), respectively.

2) Sensor Noise Model: We estimated the following noise sources [51].

1) *Photon Noise (or Shot Noise):* It is a natural consequence of the photoelectric effect. The number of electrons generated is Poisson-distributed.
2) *Dark Current Noise:* It is the leakage current within each pixel in the absence of light. The electrons are generated by thermal effects and crystallographic defects in the depletion region.
3) *Dark Signal Nonuniformity (DSNU):* It is random distribution of offset levels across pixels measured at short exposure durations in the absence of light.
4) *Photoresponse Nonuniformity (PRNU):* It describes the variation of the gain (slope) when measuring the electrical signal as a function of incident photons.
5) *Read Noise:* It arises from electrical noise in the circuitry that reads out the pixel value.

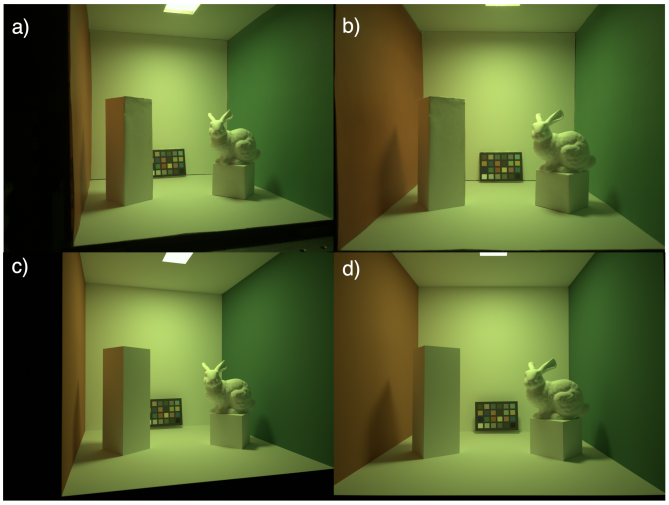


Fig. 4. Qualitative comparisons between the simulation and measurement from two camera positions. (a) and (b) Measurements. (c) and (d) Simulations.

6) *Reset Noise:* It arises from small fluctuations in the voltage level achieved when resetting the pixel prior to an acquisition.

Some types of noise differ with each acquisition (temporal noise). Other types of noise are variations across the sensor surface that remain consistent across acquisitions (fixed pattern). Certain potential sources of variation, such as column fixed pattern noise, are very small and not included in IMX363 sensor modeling.

3) Combined Noise Model Estimates: The combined effect of different noise sources is present in the sensor digital values. The expected noise is signal-dependent, largely because the photon noise is signal-dependent. To evaluate the accuracy of the simulated noise, we compared the standard deviation of the simulation and measurement in multiple regions that span a range of signal levels (Section IV).

IV. Results

A. Qualitative Appearance Comparison

The simulated and measured sensor image data appear quite similar (Fig. 4). Both represent the HDR of the scene. For example, the area light source is saturated, while the shadows and corners are very dark. Light reflected from the colored walls can be seen on the sides of the cubes.

We do not expect to observe a pixel-by-pixel match between the simulated and measured camera images. This is partly due to small differences in the positions of the objects and camera in the simulated and real scenes. Furthermore, we modeled the scene illumination as a uniform area light, and this approximation also introduces some differences. Hence, to characterize the simulation accuracy, we make a series of quantitative summary measures. In Sections IV-B–IV-F, we use these measures to evaluate relative illumination, optical blur, depth of field, chromatic channel responses, and sensor noise.

B. Relative Illumination

In many cell phone cameras, the optics introduce a substantial change in the relative illumination with field height. Using the Zemax black-box model of the Google Pixel 4a lens, we expect a falloff from the center to the 3-mm field height to be ≈60% (Fig. 5). The relative illumination, calculated using the ray-transfer method described in [32], matches the Zemax prediction very closely. The measured relative illumination differs, falling off by ≈70%. The most likely cause of the difference is that we did not model the microlens array placed on the surface of nearly all commercial sensors. This information is not included in the Zemax black box model, and we did not have access to the microlens description. This effect is described in the literature [52]. In subsequent calculations, we use the measured relative illumination in the simulation.

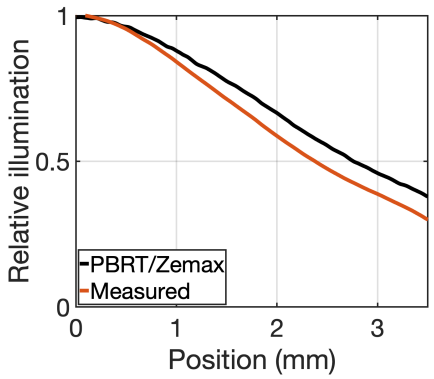


Fig. 5. Comparison of simulated and measured relative illumination. The curves show the relative illumination from the center of the sensor to a 3.5-mm field height (the sensor diagonal is 7 mm). The Zemax black box model relative illumination and ray-transfer prediction in PBRT are nearly identical (black). Neither closely matches the measured relative illumination (red), which falls off more rapidly.

C. MTF and Depth of Field

The LSF measures how light from a thin line is spread across the sensor surface; it serves as a summary measure of spatial blur. The LSF is narrow for a line at the focal distance, broadening for lines placed closer or farther. We measured the LSF function of the camera using the methods defined by the International Standard Organization (ISO) 12 233 standard [53]. To validate the RTF, two slanted edge targets were placed in the Cornell box at 0.3 and 0.5 m from camera position. Two images were acquired, one with the focal length set to each of these distances. We expect that when one slanted edge target is in focus, the LSF for the other slanted edge target will be wider.

The simulated images in Fig. 6 illustrate the scene geometry and regions of interest. The LSFs computed from the measured and simulated images are shown in Fig. 7. Fig. 7(a) compares the LSFs with the camera focused at 0.3 m, and Fig. 7(b) compares the LSFs with the camera focused at 0.5 m. As expected, in both cases, the LSF is broader when the line is away from the focal distance.

To further compare the data, we transformed the LSFs to MTFs in Fig. 7(c) and (d). The MTF describes the reduction in image contrast as a function of image spatial frequency. The narrow LSF becomes a broader MTF. Both the MTF and LSF measure the depth-dependence, which defines the depth of field. This will vary with the lens and its aperture opening. The Google Pixel 4a does not adjust the aperture, and for this reason, we compared using focal distance.

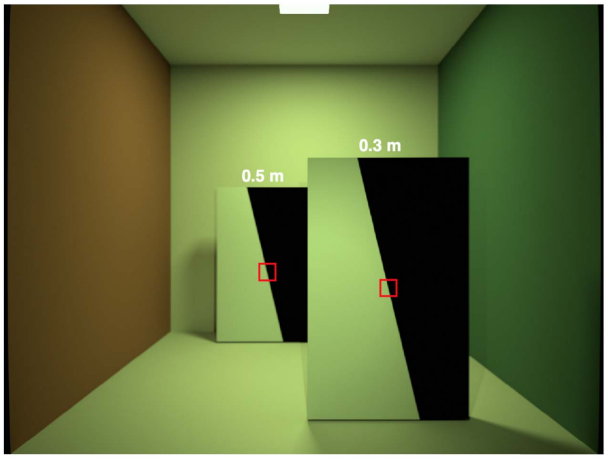


Fig. 6. LSF was estimated from a simulated image, the contained two slanted edges. These were positioned at two different distances (0.3 and 0.5 m) and field heights (0.7 and 0.8 mm) that were matched between the simulation and the Google Pixel 4a camera measurement. The small red boxes show the regions of interest used to estimate the LSF and MTF. See text for details.

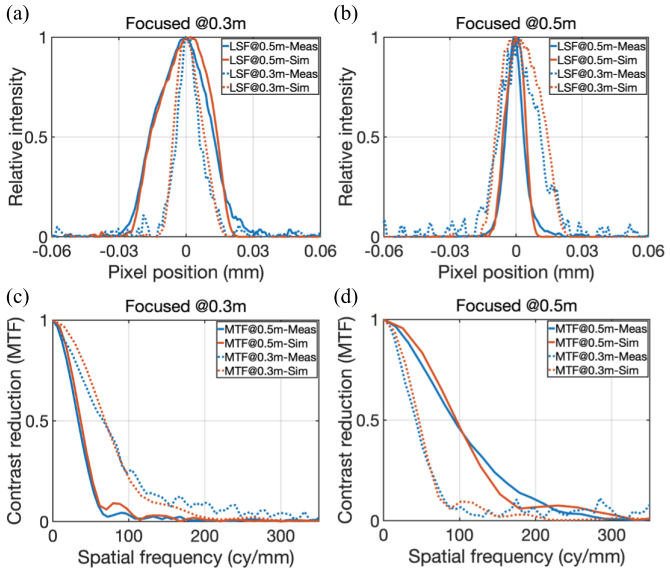


Fig. 7. Comparison of the spatial blur and depth of field. (a) and (b) Measured (blue) and simulated (red) LSFs are compared. (a) LSF when the focal distance is 0.3 m. (b) Focal distance is 0.5 m. In each case, the narrow LSF is for the line at the focal distance, and the broader LSF is for the line that is too near or too far. (c) and (d) LSF data are transformed into the corresponding MTF curves where the broader MTF is for the line that is in focus. The measured and simulated lines were at slightly different field heights (0.7 and 0.8 mm), but this difference did not have a substantial impact on the measurements.

D. Sensor Noise

In this section, we describe our evaluation of the sensor noise model. The model includes photon noise and several classes of temporal (reset, dark current, and read) and fixed pattern (DSNU and PRNU) noise. These sources contribute differently to the total noise, and some of these sources depend on the sensor irradiance level.

To assess the accuracy of the sensor noise model, we compared simulated and measured data from many 10×10 image regions with a wide range of digital levels. Specifically, we selected 500 uniform patches using criteria described in Section IV. The regions are shown overlaid on a measured image in Fig. 8(a). The sensor noise, measured as the standard deviation of the green pixels, increases with the mean level [Fig. 8(b)]. The relationship between the standard deviation and the mean is similar whether estimated using the measured or simulated data. Hence, we conclude that the sensor noise model is a good approximation to the measured data.

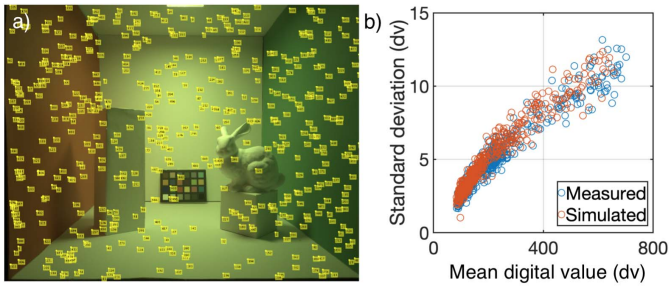


Fig. 8. Sensor noise estimated in measurement and simulation. (a) We randomly tested many small image regions. We accepted a region for a noise estimate if it contained no dead pixels, and the values were consistent with a uniform spectral irradiance. The yellow boxes show the regions that were accepted. See text for details. (b) Comparing the simulated and predicted noise in the uniform regions, we find a good match.

E. Inter-Reflections

Next, we compare the simulated and measured digital values for an image that includes substantial inter-reflections. We acquired and simulated images of the Cornell box that contained an MCC at the rear wall. Fig. 9(a)–(c) shows the images that were measured when the MCC was positioned closer to the left wall (red), right wall (green), or halfway in between. The graphs [Fig. 9(d)–(f)] below each image plot the digital values in a horizontal line passing through the achromatic series of the MCC in that image. The solid lines plot the digital values from the measured image, and the points plot the digital values from the simulated image, with one free scalar adjustment. This adjustment was necessary, because the angle of the simulated MCC is perfectly parallel to the rear wall, but the measured MCC was tilted slightly. Hence, the amount of light scattered from the achromatic series was larger in the measured image than in the simulated image. We, therefore, scaled all of the simulated values in the horizontal line passing through the achromatic series of the MCC by a single positive constant that brought the two sets of data into reasonable agreement.

There are several features worth noting when comparing the simulated and measured digital values. First, notice that the ratio between the green:red channel ratio is close to one (1.08) at the left (red) wall and becomes significantly higher near the right (green) wall (1.28). This change is due to the light reflected onto the achromatic patches from the nearby walls. Second, the variance in the simulated and acquired data is quite similar at all levels. Finally, there is a difference in the black line regions between the simulated and measured data. This arises because the black lines in the simulated MCC were set to zero reflectance, but the black lines in the real MCC are slightly more reflective. We left this imperfection in the simulation to show that it is possible to see such small mismatches. Note also that the lowest digital value contains the black level offset (64).

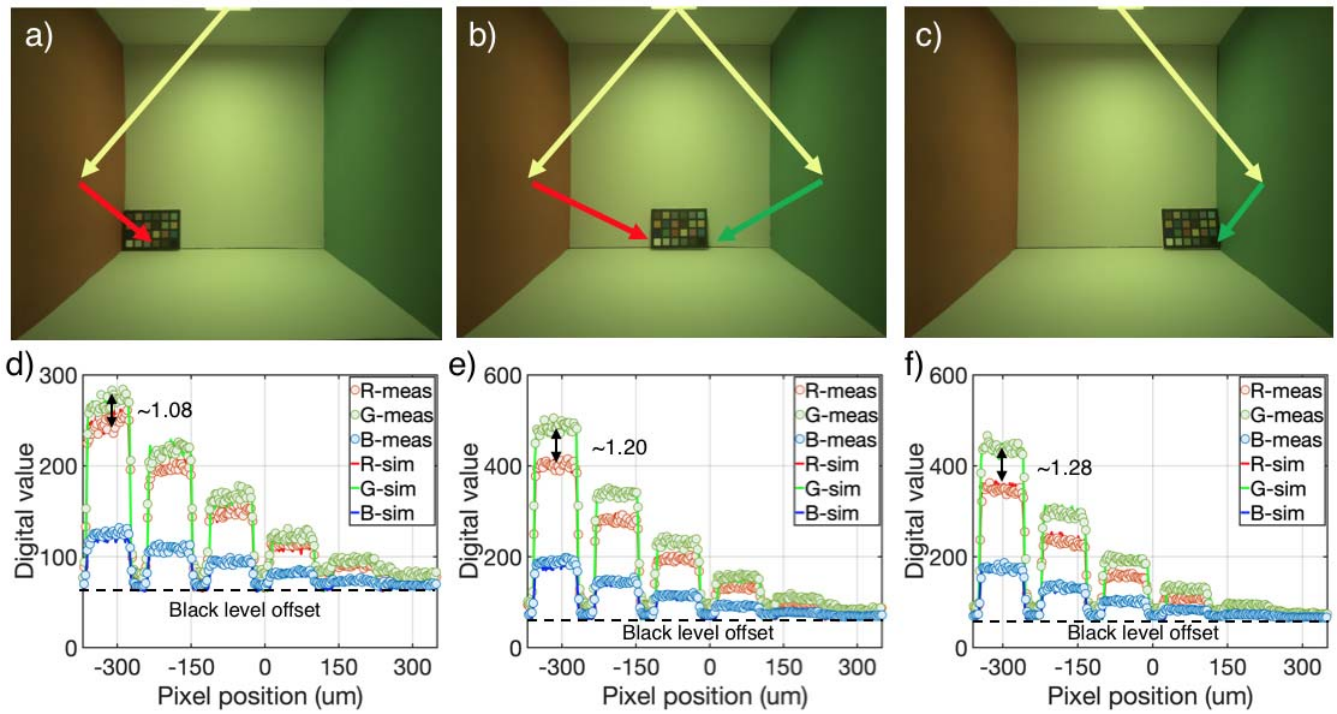


Fig. 9. Comparisons of simulated and measured digital values in the complex scene including inter-reflections. (a)–(c) Acquired images of an MCC at three different positions along the rear wall. The arrows show the likely paths for inter-reflections from the light source off the nearby walls onto the MCC. The white rectangle highlights the position of the achromatic patches on the MCC. (d)–(f) Digital values from the Google Pixel 4a images (dashed lines) are compared with the simulated values (solid lines).

F. Validation of Chromatic Channel and Relative Illumination

Finally, we performed one more assessment of the accuracy of the sensor QE by comparing simulated and measured R, G, and B digital values of the MCC. This validation was performed with a miniature version of the MCC in the center of a Gretag light booth. The booth has three different lights nominally labeled "illuminant A," "Cool White Fluorescent (CWF)," and "day." For each light, we captured images using the Google Pixel 4a camera in eight different positions, for a total of 576 color patch comparisons (3 illuminants $\times$ 8 positions $\times$ 24 color patches). Moving the camera while keeping the position of the MCC fixed places the MCC at different field heights and preserves the same illumination on the MCC. To correctly simulate these sensor responses, we must have an accurate model of both the color channels and the relative illumination.

We simulated RGB values in several steps. First, we multiplied the spectral reflectance of each of the 24 MCC color patches with the spectral power of the illumination and the spectral QEs of the Google Pixel 4a camera. Second, we corrected for relative illumination by dividing the measured RGB values by the measured relative illumination (see Fig. 5). Third, we used the conversion gain (see Appendix III) to calculate the sensor RGB digital values. Finally, we added the digital black level (64) to the simulated values.

Fig. 10 compares 1728 measured and simulated mean RGB values (576 patches with three values) at several steps in the calculation. We first compare values without accounting for relative illumination and using the nominal sensor QE functions [Fig. 3(a)]. Next, we compare the measured and simulated RGB values after correcting for sensor gain and crosstalk [see Fig. 3(b)]. Finally, we make the same comparison also accounting for the effect of relative illumination [Fig. 10(c)]. Each additional step improves the fit between simulated and measured data. The histogram panel [Fig. 10(d)] illustrates the distribution of digital values in the dataset.

V. Discussion

We assessed simulation accuracy using a series of quantitative measurements: the end-to-end image systems simulation matches the camera data in many ways. Simulations of the optical blur and depth of field agreed with the measurements to within a few percent (Figs. 7 and 10). Simulations of the spatial pattern of sensor responses within a complex scene (Fig. 9) and the sensor noise (Fig. 8) are also in close agreement with the measurements. The quantitative agreement between the simulated and measured camera image data is a strong validation of the end-to-end simulation methodology.

The validation is based on a 3-D scene that includes shadows and inter-reflections. The light scattered from the colored walls is a substantial factor in illuminating the sides of the cubes, and the cubes cast significant shadows on the walls. All of these features are congruent in the simulated and measured camera images.

Modeling commercial camera optics is a long-standing challenge in image systems simulation. Goossens *et al.* [32] developed the ray-transfer method that enables us to model a Google Pixel 4a lens even though the specific design is unknown. We found that key properties determined by the optics—relative illumination, optical blur, and depth of field—can be matched even though the specific lens components are

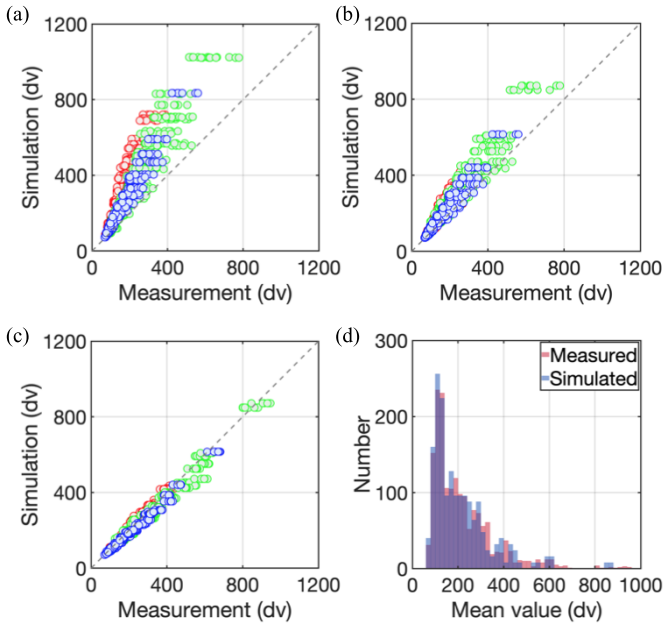


Fig. 10. Validation of estimated color filter QE. The scatter plots compare the mean simulated and measured value for the 24 patches under three different illuminants. The color of each point is a measurement in the R, G, or B channels. (a)–(c) Fit using three different simulations. (a) Without lens vignetting, channel gain, or crosstalk correction, the fits are quite poor. (b) Correcting for lens vignetting alone improves the fit. (c) Correcting for lens vignetting, channel gain, and crosstalk places all the points near the identity line. (d) Histogram shows the distribution of measured and simulated digital values; most of the comparisons are based on digital values less than 500.

not specified. These results show that the ray-transfer function obtained from a Zemax black model for the Google Pixel 4a camera can be substituted for a lens model in PBRT.

Prior work established the accuracy of linear plus noise sensor models for planar scenes with simple lighting [8], [54]. Image system simulation validation based on 3-D scenes with fluorescent materials and sensors was assessed qualitatively in [22]. The tests here substantially extend those validations by comparing the R, G, and B digital values in simulated and measured camera images of the MCC under multiple light sources and different camera positions.

The mean pixel values in the measured camera images of the MCC are very close to the predicted values [Figs. 9 and 10(c)]. This agreement spans measurements in different spatial positions and different illuminants, confirming the general accuracy of the sensor model. The standard deviation of the points is similar, though somewhat larger than the expected variation based on sensor and photon noise alone [Fig. 8(b)]. To explain this, additional variance may require more precise accounting for factors, such as rendering noise.

Accurate sensor modeling is necessary for designing novel sensor architectures. Soft prototyping using realistic scene data can shorten development time by enabling designers to select imaging components and avoid mistakes. Image systems simulation also has value in developing subsequent signal processing and machine learning algorithms. Considering, for example, that the machine learning developers for driving applications are using synthetic camera images to augment datasets and create specific training scenarios (e.g., [55], [56], [57]). Each of those vendors emphasizes different types of validations for their simulations. A contribution of this work is the focus on properties of the optics and sensor that are beyond what we have found in the literature.

Correctly modeling the optics and sensor properties, including pixel size and color filter arrays, is important for many applications. For example, neural networks trained on physically based simulations of camera images can detect cars in real camera image data almost as well as neural networks trained on the real camera image data [3]. Furthermore, neural networks trained on camera images generated by physically based simulations performed better than neural networks trained on camera images generated by raster-based graphics or by ray-traced graphics rendering methods that did not include the correct optics and sensor modeling [3], [4].

The results of our study support the use of physically based end-to-end image systems simulations to create large datasets that are automatically and accurately labeled for training neural networks. The results also support using the simulations to assess how hardware changes will impact system performance. These analyses can be helpful in fields apart from consumer photography, such as medical diagnosis, driving, and robotics.

VI. Future Work

The results that we report in this article give us confidence in our ability to accurately simulate the camera imaging certain complex, natural scenes. Some of the limitations in this work are a good target for future work.

The range of scenes that can be simulated can be expanded. Computer graphics models of fog (participating media), complex materials (translucent, retro-reflective), and complex area lights remain an active research area. The validation performed here is for a scene under clear conditions with relatively simple materials and lights.

There are also opportunities to extend the optics and sensor models. ISETCam and ISET3d include the ability to simulate microlens arrays. We did not have this information, and the missing information produced a difference between the simulated and measured relative illumination. We are investigating methods for accounting for the microlens array when this information is not provided. As related, we are simulating sensors with multiple photosites beneath each microlens to measure the light field. A particularly simple form of these sensors, the dual pixel architecture, is already in wide use for setting focus. A useful extension of the analysis here is to simulate dual pixel and light field sensors. These simulations will help us understand how much information can reliably be extracted from these devices.

We are also exploring other types of imaging systems, such as time-of-flight and gated-CMOS cameras. Our simulation of such devices has been enabled by the ability to use ray tracing to calculate the path length of each ray from the sensor into the scene, including the effect of optical elements and filters that are in the light path. Further work that accounts for the participating media on the light path (fog and smoke), and the bidirectional scattering distribution functions of the materials at the relevant wavelengths, will be helpful in the design of such systems and in synthesizing realistic datasets. Such datasets can be used to extract information about the scene using different algorithms, or simply to create labeled training data for machine learning applications.

TABLE I
SONY IMX363 SENSOR SPECIFICATION

Properties	Parameters	Values (units)
Geometric	Pixel Size	[1.4, 1.4] (μm)
	Fill Factor	100 (%)
Electronics	Well Capacity	6000 (e^-)
	Voltage Swing	0.4591 (volts)
	Conversion Gain	0.1707(dv/e^-)
	Analog Gain	1
	Black Level Offset	64 (dv)
	Quantization Method	10 (bit)
Noise Sources @Analog gain=1	DSNU	0.038 (mV)
	PRNU	0.54 (%)
	Dark Voltage	0.02 (mV/sec)
	Read Noise	0.226 (mV)

For generating large-scale dataset for machine learning purposes, more work is needed on improving the ability to compose natural and realistic virtual 3-D scenes. Specifically, smart algorithm for scene auto assembly and using more physically faithful material reflectance data will be beneficial.

An important goal for this project is to build trust in these simulations. We work toward this goal by the following: 1) assessing the accuracy of our simulations and 2) making our image systems simulation software open source and freely available. The code, which relies on three GitHub repositories (ISETcam [5], ISET3d [6], and ISETCornellbox [7]) also explains how to perform this validation. The parameters of the calculations and methods are documented in the source code. By making these soft-prototyping tools available, we hope to build trust and advance the design of imaging sensors for future imaging applications.

APPENDIX I
RENDER CONFIGURATION

The optical images of the Cornell Box scene were rendered using PBRT-V3 [13] with the RTF modifications. The images were rendered at the same spatial sampling resolution as the Sony sensor (3024 × 4032). The rendering parameters were set to six bounces and 3072 rays per pixel. These parameters were chosen to reduce the rendering noise to a very small level. Rendered on a CPU with 40 cores, the rendering time was 15 h. Experiments with PBRT-V4 [58] and a 3080 Nvidia GPU suggest that the rendering time for the same scene will be ≈1.5 h, a tenfold speedup.

APPENDIX II
SENSOR PARAMETERS

Sensor parameters used for simulation are in Table I. The pixel size, fill factor, well capacity, voltage swing, conversion gain, black level offset, and quantization are provided by the manufacturer. We also estimated conversion gain (see Appendix III).

We estimated the DSNU, PRNU, dark voltage, and read noise from laboratory measurements. Specifically, we used an integrating sphere to produce images of uniform intensity. We then acquired a set of images for a range of integration times. We fit the measurements from a collection of different exposure times to estimate these quantities (see methods in [8]).

APPENDIX III
ESTIMATING CONVERSION GAIN

We estimate the conversion gain by analyzing the pixel values in an image of a bright scene, where we expect shot noise to be dominant. In the ideal case, the number of excited electrons, which is the signal $\tilde{S}$, follows a Poisson distribution [59]:

$$\tilde{S} \sim \text{Poisson}(\mu). \tag{4}$$

The mean and variance are equal. In practice, the observed signal also depends on the PRNU and DSNU. The measured DSNU is very small, and so, we do not include it in the calculations. The PRNU $\tilde{P}$ follows a normal distribution:

$$\tilde{P} \sim 1 + \mathcal{N}(0, \sigma_{\text{prnu}}). \tag{5}$$

The observed signal $\tilde{O}$ will be the product of these two independent random variables

$$\tilde{O} = \tilde{S}\tilde{P}. \tag{6}$$

The expected value of the number of observed electrons can be expressed in terms of the mean and variance of these two random variables

$$E(\tilde{O}) = E[\tilde{S}\tilde{P}] = E(\tilde{S})E(\tilde{P}) = \mu. \tag{7}$$

The variance of the observed signal $V(\tilde{O})$ is

$$\begin{aligned} V(\tilde{O}) &= V[\tilde{S}\tilde{P}] \\ &= [E(\tilde{S})]^2 V(\tilde{P}) + [E(\tilde{P})]^2, V(\tilde{S}) + V(\tilde{S})V(\tilde{P}) \\ &= \mu^2\sigma^2_{\text{prnu}} + \mu + \mu\sigma^2_{\text{prnu}}. \end{aligned} \tag{8}$$

Since we can only measure digital values, the conversion gain α is defined as converting electrons directly to digital value that has a unit of (dv/e$^-$)

$$\text{DV} = \alpha\tilde{O}. \tag{9}$$

The relationships of expectation and variance between digital value and observed signal are

$$\begin{aligned} E(\text{DV}) &= \alpha E(\tilde{O}) \\ V(\text{DV}) &= \alpha^2 V(\tilde{O}). \end{aligned} \tag{10}$$

Substituting in (7) and (8), we have

$$\begin{aligned} V(\text{DV}) &= \alpha^2 V(\tilde{O}) \\ &= \alpha^2(\mu^2\sigma^2_{\text{prnu}} + \mu + \mu\sigma^2_{\text{prnu}}) \\ &= \alpha^2\left(\frac{[E(\text{DV})]^2\sigma^2_{\text{prnu}}}{\alpha^2} + \frac{E(\text{DV})}{\alpha} + \frac{E(\text{DV})}{\alpha}\sigma^2_{\text{prnu}}\right) \\ &= [E(\text{DV})]^2\sigma^2_{\text{prnu}} + \alpha E(\text{DV})\left(1 + \sigma^2_{\text{prnu}}\right). \end{aligned} \tag{11}$$

Finally, by rearranging terms, we have a solution for α in terms of the measured digital values from a uniform scene and the measured PRNU

$$\alpha = \frac{V(\text{DV}) - [E(\text{DV})]^2\sigma^2_{\text{prnu}}}{E(\text{DV})\left(1 + \sigma^2_{\text{prnu}}\right)}. \tag{12}$$

To estimate the mean and variance of the digital values of a bright uniform scene, we used 1500 pixel values taken from 15 images acquired using an integrating sphere (see Appendix II). We took 10 × 10 crops near the center of each

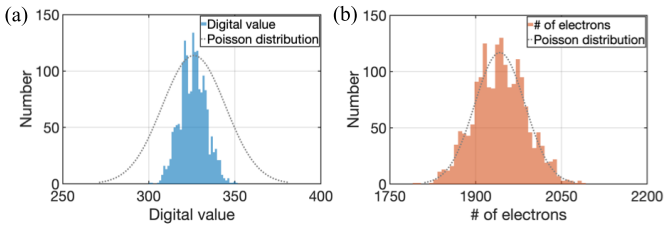


Fig. 11. Estimating conversion gain from digital values. (a) Histogram of digital values. The mean is equal to 320.7, and the variance is equal to 57.8, which differs substantially from the Poisson distribution. (b) After scaling by the inverse of the conversion gain, we can obtain an estimate of the number of electrons. In this case, the distribution mean and variance are more nearly equal as expected in (6) and more closely match the expected Poisson distribution.

image; this minimized the influence of lens vignetting. The estimated conversion gain 0.1677 dv/e$^-$ is close to the value provided by the manufacturer (0.1707 dv/e$^-$), differing by only 1.8%. Fig. 11 shows the histogram of the digital values [Fig. 11(a)] and estimated number of electrons [Fig. 11(b)]. The distribution of digital values is very different from Poisson. After applying the estimated conversion gain to estimate the number of electrons, the distribution is close to Poisson, as expected.

Appendix IV
Uniform Region Identification

To estimate sensor noise, we chose regions in the image with uniform response levels. We randomly sampled a large number of 10 × 10 square regions in both the simulated and measured images. For each region, we extracted the 50 green channel responses and selected regions based on an analysis of the digital values.

We rejected any region from measured images that contained an obviously "dead" pixel or significant image nonuniformity. To accomplish this, we first converted the median value of the green pixel responses to electrons, using the conversion gain (see Appendix III). In the absence of sensor noise, a uniform region would have a variance equal to the mean, as predicted by the Poisson distribution of photon noise. We rejected any region that had a pixel value of more than three standard deviations from the mean. We performed the same analysis on the simulated data, though in that case, we attribute the rejection to rendering noise.

Second, we defined a criterion for nonuniformity. We held out 20% of green values and fit the remaining values with the second-order spatial polynomial. We then compared the root mean squared error (RMSE) of the held out data to the polynomial fit and to a constant value set to the mean of the green pixel values. We rejected a region as nonuniform if the RMSE of the fit to the constant was larger (2% greater) than the RMSE of the polynomial fit.

Acknowledgment

The authors would like to thank Krithin Kripakaran for constructing the Cornell Box. They would like to thank Max Furth and Eric Tang for building a model of the Cornell Box in Cinema 4-D. They would like to thank Zhenyi Liu, Henryk Blasinski, and David Cardinal for their contributions in developing and supporting the ISET3d code and also for many helpful discussions. They would also like to thank Gordon Wan, Jamyuen Ko, Guangxun Liao, Bonnie Tseng, and Ricardo Motta for their advice, support, and encouragement for this project.

References

[1] Y. Kang, H. Yin, and C. Berger, "Test your self-driving algorithm: An overview of publicly available driving datasets and virtual testing environments," *IEEE Trans. Intell. Vehicles*, vol. 4, no. 2, pp. 171–185, Jun. 2019.
[2] A. Elmquist and D. Negrut, "Modeling cameras for autonomous vehicle and robot simulation: An overview," *IEEE Sensors J.*, vol. 21, no. 22, pp. 25547–25560, Nov. 2021.
[3] Z. Liu, T. Lian, J. Farrell, and B. A. Wandell, "Neural network generalization: The impact of camera parameters," *IEEE Access*, vol. 8, pp. 10443–10454, 2020.
[4] Z. Liu, J. Farrell, and B. A. Wandell, "ISETAuto: Detecting vehicles with depth and radiance information," *IEEE Access*, vol. 9, pp. 41799–41808, 2021.
[5] *ISETCAM*. Accessed: May 2, 2022. [Online]. Available: https://github.com/ISET/isetcam
[6] *ISET3D*. Accessed: May 2, 2022. [Online]. Available: https://github.com/ISET/iset3d
[7] *IsetCornellBox*. Accessed: May 2, 2022. [Online]. Available: https://github.com/ISET/isetcornellbox
[8] J. E. Farrell, P. B. Catrysse, and B. A. Wandell, "Digital camera simulation," *Appl. Opt.*, vol. 51, no. 4, p. A80, Feb. 2012.
[9] J. E. Farrell, F. Xiao, P. B. Catrysse, and B. A. Wandell, "A simulation tool for evaluating digital camera image quality," *Proc. SPIE*, vol. 5294, pp. 124–131, Dec. 2003.
[10] J. Farrell, M. Okincha, and M. Parmar, "Sensor calibration and simulation," in *Digital Photography IV*, vol. 6817. Bellingham, WA, USA: International Society for Optics and Photonics, Mar. 2008, Art. no. 68170R.
[11] J. Chen *et al.*, "Digital camera imaging system simulation," *IEEE Trans. Electron Devices*, vol. 56, no. 11, pp. 2496–2505, Nov. 2009.
[12] Z. Liu *et al.*, "A system for generating complex physically accurate sensor images for automotive applications," 2019, *arXiv:1902.04258*.
[13] M. Pharr, W. Jakob, and G. Humphreys, *Physically Based Rendering: From Theory to Implementation*. San Mateo, CA, USA: Morgan Kaufmann, Sep. 2016.
[14] C. M. Goral, K. E. Torrance, D. P. Greenberg, and B. Battaile, "Modeling the interaction of light between diffuse surfaces," *SIGGRAPH Comput. Graph.*, vol. 18, no. 3, pp. 213–222, Jan. 1984.
[15] M. Cohen, D. Greenberg, D. Immel, and P. Brock, "An efficient radiosity approach for realistic image synthesis," *IEEE Comput. Graph. Appl.*, vol. CGA-6, no. 3, pp. 26–35, Mar. 1986.
[16] G. W. Meyer, H. E. Rushmeier, M. F. Cohen, D. P. Greenberg, and K. E. Torrance, "An experimental evaluation of computer graphics imagery," *ACM Trans. Graph.*, vol. 5, no. 1, pp. 30–50, Jan. 1986.
[17] N. Sumanta Pattanaik, A. James Ferwerda, E. Kenneth Torrance, and D. Greenberg, "Validation of global illumination simulations through CCD camera measurements," in *Proc. Color Imag. Conf.*, 1997, pp. 53–250.
[18] T. Lian, J. Farrell, and B. Wandell, "Image systems simulation for 360 deg camera rigs," *Electron. Imag.*, vol. 2018, p. 353, Jan. 2018.
[19] H. Blasinski, T. Lian, and J. Farrell, "Underwater image systems simulation," in *Imaging and Applied Optics*. Washington, DC, USA: The Optical Society (OSA), Jun. 2017.
[20] H. Blasinski, J. Farrell, T. Lian, Z. Liu, and B. Wandell, "Optimizing image acquisition systems for autonomous driving," *Electron. Imag.*, vol. 2018, no. 5, p. 161, 2018.
[21] Z. Liu, T. Lian, J. Farrell, and B. Wandell, "Soft prototyping camera designs for car detection based on a convolutional neural network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV) Workshops*, Oct. 2019.
[22] Z. Lyu *et al.*, "Simulations of fluorescence imaging in the oral cavity," *Biomed. Opt. Exp.*, vol. 12, pp. 4276–4292, Jul. 2021.
[23] J. Farrell *et al.*, "Soft-prototyping imaging systems for oral cancer screening," *Electron. Imag.*, vol. 2020, no. 7, p. 212, 2020.
[24] T. Lian, K. J. MacKenzie, D. H. Brainard, N. P. Cottaris, and B. A. Wandell, "Ray tracing 3D spectral scenes through human optics models," *J. Vis.*, vol. 19, no. 12, p. 23, Oct. 2019.
[25] F. Huck and S. Wall, "Image quality prediction: An aid to the Viking Lander imaging investigation on Mars," *Appl. Opt.*, vol. 15, no. 7, pp. 1748–1766, 1976.
[26] J. R. Schott, S. D. Brown, R. V. Raqueño, H. N. Gross, and G. Robinson, "An advanced synthetic image generation model and its application to multi/hyperspectral algorithm development," *Can. J. Remote Sens.*, vol. 25, no. 2, pp. 99–111, Jun. 1999.

[27] W. T. Cathey, B. R. Frieden, W. T. Rhodes, and C. K. Rushforth, "Image gathering and processing for enhanced resolution," *JOSA A*, vol. 1, no. 3, pp. 241–250, 1984.
[28] F. O. Huck, C. L. Fales, N. Halyo, R. W. Samms, and K. Stacy, "Image gathering and processing: Information and fidelity," *JOSA A*, vol. 2, no. 10, pp. 1644–1666, 1985.
[29] J. M. Booth and J. B. Schroeder, "Design considerations for digital image processing systems," *Computer*, vol. 10, no. 8, pp. 15–20, Aug. 1977.
[30] C. Garnier, R. Collorec, J. Flifla, and F. Rousee, "General framework for infrared sensor modeling," in *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing IX*, vol. 3377. Bellingham, WA, USA: Society of Photographic Instrumentation Engineers, 1998, pp. 59–70.
[31] C. Garnier, R. Collorec, J. Flifla, C. Mouclier, and F. Rousee, "Infrared sensor modeling for realistic thermal image synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal*, vol. 6, Jun. 1999, pp. 3513–3516.
[32] T. Goossens, Z. Lyu, J. Ko, G. C. Wan, J. Farrell, and B. Wandell, "Ray-transfer functions for camera simulation of 3D scenes with hidden lens design," *Opt. Exp.*, vol. 30, no. 13, pp. 24031–24047, 2022.
[33] R. White, "Validation of Rochester Institute of Technology's (RIT's) digital image and remote sensing generation (DIRSIG) model, reflective region," Ph.D. dissertation, Rochester Inst. Technol., Rochester, NY, USA, 1996.
[34] E. D. Peterson, S. D. Brown, T. J. Hattenberger, and J. R. Schott, "Surface and buried landmine scene generation and validation using the digital imaging and remote sensing image generation model," in *Imaging Spectrometry X*, vol. 5546. Bellingham, WA, USA: Society of Photographic Instrumentation Engineers, Oct. 2004, pp. 312–323.
[35] *Verification and Validation Studies*. Accessed: Dec. 21, 2021. [Online]. Available: https://dirsig.cis.rit.edu/docs/new/validation.html
[36] *Dirsig = Digital Imaging and Remote Sensing Image Generation*. Accessed: May 5, 2022. [Online]. Available: http://dirsig.cis.rit.edu/
[37] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learn.*, 2017, pp. 1–16.
[38] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "AirSim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics*. Cham, Switzerland: Springer, 2018, pp. 621–635.
[39] T. Duy Son, A. Bhave, and H. Van der Auweraer, "Simulation-based testing framework for autonomous driving development," in *Proc. IEEE Int. Conf. Mechatronics (ICM)*, vol. 1, Mar. 2019, pp. 576–583.
[40] S. Hossain and D.-J. Lee, "Autonomous-driving vehicle learning environments using unity real-time engine and end-to-end CNN approach," *J. Korea Robot. Soc.*, vol. 14, no. 2, pp. 122–130, May 2019.
[41] S. Hossain, A. R. Fayjie, O. Doukhi, and D.-J. Lee, "CAIAS simulator: Self-driving vehicle simulator for AI research," in *Intelligent Computing & Optimization*. Cham, Switzerland: Springer, 2019, pp. 187–195.
[42] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3234–3243.
[43] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtualworlds as proxy for multi-object tracking analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4340–4349.
[44] A. Tsirikoglou, J. Kronander, M. Wrenninge, and J. Unger, "Procedural modeling and physically based rendering for synthetic data generation in automotive applications," 2017, *arXiv:1710.06270*.
[45] Y. Zhang *et al.*, "Physically-based rendering for indoor scene understanding using convolutional neural networks," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5287–5295.
[46] M. Grapinet, P. De Souza, J.-C. Smal, and J.-M. Blosseville, "Characterization and simulation of optical sensors," *Accident Anal. Prevention*, vol. 60, pp. 344–352, Nov. 2013.
[47] D. Gruyer, M. Grapinet, and P. D. Souza, "Modeling and validation of a new generic virtual optical sensor for ADAS prototyping," in *Proc. Intell. Vehicles Symp.*, Jun. 2012, pp. 969–974.
[48] *Specular Reflection and Transmission*. Accessed: Aug. 5, 2022. [Online]. Available: https://www.pbr-book.org/3ed-2018/Reflection_Models/Specular_Reflection_and_Transmission
[49] *The Stanford 3D Scanning Repository*. Accessed: Oct. 1, 2022. [Online]. Available: http://graphics.stanford.edu/data/3Dscanrep/
[50] *Opencamera*. Accessed: May 2, 2022. [Online]. Available: https://opencamera.org.uk/
[51] European Machine Vision Association. *Standard for Characterization of Image Sensors and Cameras*. Accessed: Jul. 1, 2022. [Online]. Available: https://www.emva.org/wp-content/uploads/EMVA1288-General-4.0ReleaseCandidate.pdf
[52] SVS-Vistek GmbH. *Relative Illumination and Microlenses*. Accessed: Jun. 1, 2022. [Online]. Available: https://www.svs-vistek.com/en/news/svs-news-article.php?p=shading-with-CMOS-cameras
[53] *Photography—Electronic Still Picture Imaging—Resolution and Spatial Frequency Responses*, Standard ISO 12233:2017, 2017.
[54] J. Chen *et al.*, "Digital camera imaging system simulation," *IEEE Trans. Electron Devices*, vol. 56, no. 11, pp. 2496–2505, Nov. 2009.
[55] C. Kamel, A. Reid, F. Plepp. (Dec. 2021). *Validating NVIDIA DRIVE SIM Camera Models*. Accessed: Dec. 23, 2021. [Online]. Available: https://developer.nvidia.com/blog/validating-drive-sim-camera-models/
[56] (Oct. 2021). *Anyverse*. Accessed: Dec. 23, 2021. [Online]. Available: https://anyverse.ai/
[57] *Simulation City: Introducing Waymo's Most Advanced Simulation System Yet for Autonomous Driving*. Accessed: Dec. 23, 2021. [Online]. Available: https://blog.waymo.com/2021/06/SimulationCity.html
[58] *Pbrt: Version 4*. Accessed: May 5, 2022. [Online]. Available: https://github.com/mmp/pbrt-v4
[59] J. R. Janesick, *Photon Transfer*. Bellingham, WA, USA: SPIE, 2007.

Zheng Lyu is pursuing the Ph.D. degree with the Department of Electrical Engineering, Stanford University, Stanford, CA, USA, advised by Prof. Brian A. Wandell and Dr. Joyce Farrell.

His research interests include end-to-end physically based image systems simulation for consumer photography and medical imaging.

Thomas Goossens received the Ph.D. degree from Katholieke Universiteit Leuven (KU Leuven), Leuven, Belgium.

He is currently a Postdoctoral Fellow with Stanford University, Stanford, CA, USA, where he is involved in the physics of patterned photonic structures and camera simulation. He is also with imec, Leuven, where he is involved in coordinating with scientists.

Dr. Goossens was a recipient of the Fellowship of the Belgian American Educational Foundation.

Brian A. Wandell is the first Isaac and Madeline Stein Family Professor. He joined the Stanford Psychology Faculty, Stanford, CA, USA, in 1979. He is also the Director of the Stanford's Center for Cognitive and Neurobiological Imaging, Stanford, and the Deputy Director of the Stanford's Neurosciences Institute, Stanford. His research interests include vision science, spanning topics from visual disorders, reading development in children, to digital imaging devices, and algorithms for both magnetic resonance imaging and digital imaging.

Dr. Wandell is a member, by courtesy, of Electrical Engineering, Ophthalmology, and the Graduate School of Education.

Joyce Farrell is a Senior Research Engineer and a Lecturer with the Department of Electrical Engineering, Stanford University, Stanford, CA, USA. She is also the Executive Director of the Stanford Center for Image Systems Engineering, Stanford. Dr. Farrell has more than 20 years of research and professional experience working at a variety of companies and institutions, including the NASA Ames Research Center, Mountain View, CA, New York University, New York, NY, USA, the Xerox Palo Alto Research Center, Palo Alto, CA, Hewlett Packard Laboratories, Palo Alto, and Shutterfly, Redwood City, CA.