https://doi.org/10.2352/ISSN.2470-1173.2020.7.ISS-329

From the radiance reflected by a scene to a digital picture : a compact model based simulator for image sensor design

Alix de Gouvello, Laurent Soulier, Antoine Dupret
CEA, LIST - Nano-INNOV, Bât 862 - PC 172, F91191 Gif-sur-Yvette Cedex, France

Abstract

In order to explore the design space of a new, potentially unconventional, sensor or to optimize sensor characteristics for a given computer vision application, an image acquisition process simulator has been designed. Its aim is to be simple and modular, yet complete and accurate enough to match the physical phenomena involved. The approach has been described in this paper to highlight the different steps of the acquisition process and to explain the implementation choices and the hypotheses that were made. The simulator has been tested on images of point sources, on simulated test patterns and on real high definition pictures and has proven realistic.

Context and approach

The design of a new sensor can benefit from simulation to test different sets of parameters without really manufacturing the sensor, all the more when unconventional pixel organizations are explored, to save time and money. Indeed, sensors are complex systems with a large set of parameters that can and should be co-optimized.

Moreover, the development of robotics and embedded systems design involve a lot of computer vision algorithms to become less computationally and energetically expensive, more and more real-time and to drive their latency down, while keeping their performance and accuracy high. Optimizing the image aquisition process - through simulation -, so that no unuseful information is aquired and sent to the processing units for example, helps meeting those embedded systems requirements.

High fidelity simulators of this kind of systems do exist but, to the best of our knowledge, they often focus only on the optical part of the process and rely on ray-tracing or multi-physical simulations, which are computationally intensive and may not be needed for the precision required in the modeling we want to achieve. Besides, their use usually requires a high level of expertise. Simpler simulators exist, like ISET that delegates the most complex optical simulation to third party tools (Zemax, Code V) [1]. Designed by Farrell et al., it simulates the optics as well as the photoconversion, like we do. It is a good compromise but is still not lightweight enough to allow the generation of long video sequences. The main differences are : 1) ISET is more versatile in terms of input data (it can work with 3D input) and optical transfer functions (though we can quite trivially add the shift-invariant ones), 2) it modelizes the display, 3) it uses Zemax or Code V to account for distortion whereas we use empirical, yet, image analysis wise, accurate enough, distortion models, 4) our simulator offers more possibilities in the image plane to define unconventional pixel organizations and acquisition modes, 5) our simulator is lighter, thus faster.

In order to make design space exploration easier and faster, we propose a pragmatic model of the image acquisition (see Figure 3) : from a high quality source picture, we modelize the propagation of the energy through an optical system and its integration and conversion on photosites located on the focal plane of the system, taking into account various significant physical phenomena and noise sources. The most challenging part is to simulate the optics in a functional high level fashion to test numerous algorithms on various image sensors architectures. Once in the sensor plane, the simulation is more direct. The main characteristic of this simulator is to be modular (Figure 1) in order to allow the user to perform many tests.

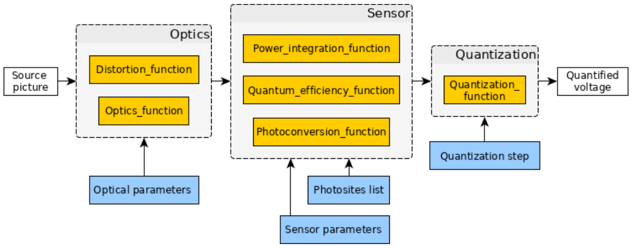


Figure 1. *A modular architecture (blue : data, yellow : function)*

Optics

Any third party optical model can be used, provided it can issue a power map in the focal plane, but we propose, by default, to use a simple one : the light is diffracted by a diaphragm, it is focused by a lens on its focal plane and distorted according to an empirical model. For a global overview of the system, see Figure 3 and [2]. Notations are given on Figure 2.

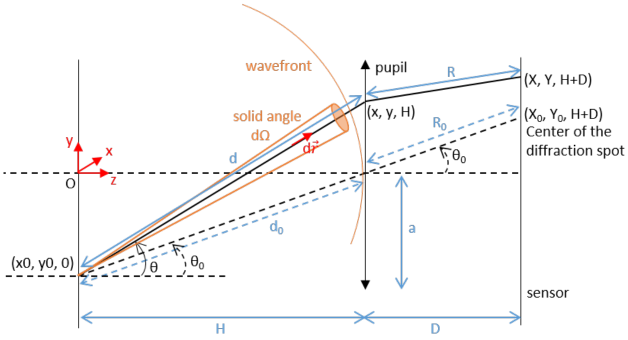


Figure 2. *The notations used*

The entry signal of the system is an image considered up to an unknown multiplying constant as the square root of a radiance

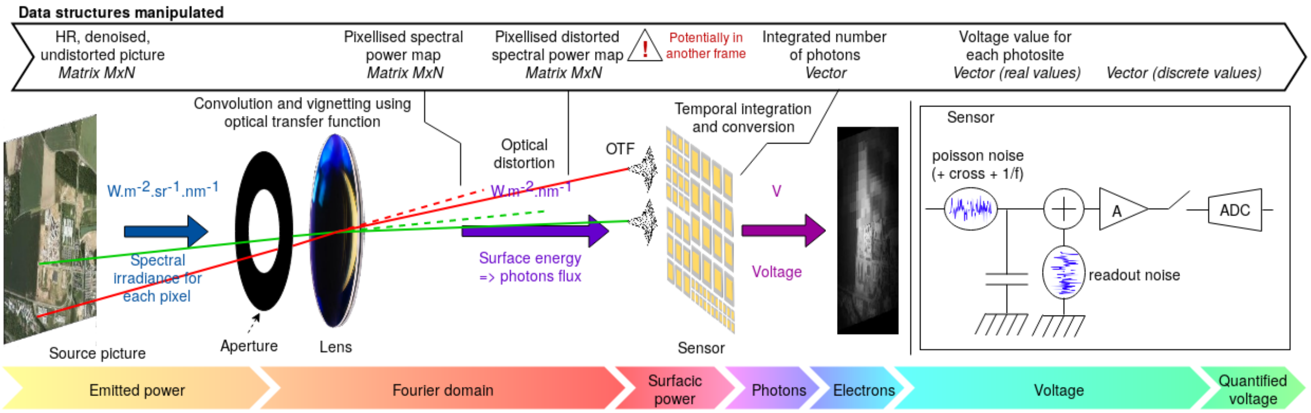

Data structures manipulated
HR, denoised, undistorted picture
Matrix MxN
Pixellised spectral power map
Matrix MxN
Pixellised distorted spectral power map
Matrix MxN
!
Potentially in another frame
Integrated number of photons
Vector
Voltage value for each photosite
Vector (real values)
Vector (discrete values)
Convolution and vignetting using optical transfer function
W.m-2.sr-1.nm-1
Spectral irradiance for each pixel
Source picture
Aperture
Lens
Optical distortion
OTF
W.m-2.nm-1
Surface energy => photons flux
Temporal integration and conversion
Sensor
V
Voltage
Sensor
poisson noise (+ cross + 1/f)
A
ADC
readout noise
Emitted power
Fourier domain
Surfacic power
Photons
Electrons
Voltage
Quantified voltage

Figure 3. *Model overview [2]*

matrix (in $W^2.m^{-2}.sr^{-1}.\lambda^{-1}$). For each pixel and each wavelength, we need the optical system PSF. Even if in our examples, we usually take into account only 3 wavelengths, it is possible to model any arbitrary number of wavelengths. We use a simple model where each pixel of the source picture is seen as a point source emitting a spherical wave whose equation is :

$$A(x,y) = A(x_0,y_0)e^{-i(\omega t - \vec{k}.\vec{r})} \tag{1}$$

A is the amplitude of the wave (in $W^{1/2}.m^{-1}.sr^{-1/2}.\lambda^{-1/2}$), ω is its pulse, $\vec{k}$ its wave vector and $\vec{r} = {}^t(x-x_0, y-y_0, z)$ since $z_0 = 0$. We consider the wave propagation to the pupil, immediately followed by a lens, where we apply the Huygens model as in [2]. This allow us to derive a model that takes into account vignetting and diffraction.

As an additional simplification, we are looking for a simple analytical expression. This requires the Gauss conditions : the source should be far from the pupil and the angles between light rays and the optical axis should be small, therefore the pupil should be small compared to other distances :

$$a \ll H \quad \text{and} \quad a \ll D \tag{2}$$

Those conditions may not be verified in a real system but have not brought on noticeable errors so far in our simulations. Besides, focusing occurs on a curved surface whereas we suppose we work in a focal plane, this approximation is once again only valid for small angles, near the optical axis.

Under the conditions mentioned above, each point of the pupil can be considered as a secondary source of a spherical wave whose amplitude is obtained from energy conservation considerations. Then, the power received by a point (X,Y) in the sensor plane is the continuous sum of the contributions of each point (x_0,y_0) of the source plane :

$$P(X,Y,\lambda) = \iint_{x_0,y_0} \left(\frac{A_0(x_0,y_0,\lambda)p_0}{H\lambda R_0}\right)^2 \frac{\cos^3(\theta_0)}{2\pi} \left|FT\{T(x,y,\lambda)\}\left(-\frac{x_0}{\lambda d_0}-\frac{X}{\lambda R_0}, -\frac{y_0}{\lambda d_0}-\frac{Y}{\lambda R_0}\right)\right|^2 dx_0\,dy_0 \tag{3}$$

With p_0 the size of a source pixel, $T(x,y,\lambda)$ the dimensionless transparency function of the optical system. Typically, we chose :

$$T(x,y,\lambda) = \begin{cases} 1 & \text{if } x^2+y^2 < a \\ 0 & \text{else} \end{cases} \tag{4}$$

P is the spectral power density in $W.m^{-2}.\lambda^{-1}$, A_0 is in $W^{1/2}.m^{-1}.sr^{-1/2}.\lambda^{-1/2}$ and the Fourier transform of T is homogeneous to an area.

It should also be noted that we postulate that the whole source picture is located in the same source plane, thus we do not account for defocus.

Discretization

Because the source energy plane is already a picture, it is discrete ; thus we choose to sample the energy plane accordingly. It also implies a discretization of the optical system's optical transfer function, so the simulator does not enable to define apertures' radius a smaller then a given threshold. Let (M,N) be the dimensions of the source picture, the condition is then :

$$a \gg \frac{H\lambda}{p_0 \min(M,N)} \tag{5}$$

This condition arises from limiting the discretization error when approximating Equation (3) with a finite sum. It can be an issue when the simulated aperture's radius is close to the previous value, it can cause the aperture to be approximated by too few pixels and energy conservation to be violated. To overcome this limitation, we contemplate oversampling the output but the best solution is still to use as large source images as possible.

Optical distortion

To make our simulator more realistic, we introduced distortion empirically, assuming, as a simplification, that distortion and optical transfer function can be decoupled. We explored two distortion models : the Brown - Conrady model [4] and Scaramuzza's FishEye model [5].

The Brown - Conrady distortion model

It aims at modeling both barrel and pincushion distortion through the use of n radial distortion coefficients $k_1,\ldots,k_n$ and

n tangential distortion coefficients $r_1, ..., r_n$. We implement a second order distortion model. The normalized distorted coordinates (x_d^n, y_d^n) of a point of the image plane corresponding to the normalized undistorted coordinates (x_u^n, y_u^n) are given by Equation (6). The normalized coordinates are the special case of homogeneous coordinates where the third coordinate z equals 1.

$$\begin{cases} x_d^n - x_u^n = x_u^n(k_1 r_u^2 + k_2 r_u^4) + 2r_1 x_u^n y_u^n + r_2(r_u^2 + 2(x_u^n)^2) \\ y_d^n - y_u^n = y_u^n(k_1 r_u^2 + k_2 r_u^4) + 2r_2 x_u^n y_u^n + r_1(r_u^2 + 2(y_u^n)^2) \end{cases} \quad (6)$$

With $r_u^2 = (x_u^n)^2 + (y_u^n)^2$.

Then, to convert the real pixel coordinates $\mathbf{X}^p = (x^p, y^p, z^p)$ of a given point to its normalized coordinates $\mathbf{X}^n = (x^n, y^n, z^n = 1)$, and vice-versa, a conversion matrix K is needed, it is determined through calibration too and $\mathbf{X}^p = K\mathbf{X}^n$ with :

$$K = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}$$

f_x and f_y being the focal length expressed in pixels in the x and y directions respectively, and (c_x, c_y) is the optical center in pixels.

FishEye undistortion model

It was developped by Scaramuzza et al. [5], it links an undistorted viewing direction expressed in normalized coordinates $(x_u^n, y_u^n, z_u^n = 1)$ to the ”unstretched” distorted pixel coordinates of an image point on the sensor plane $(\tilde{x}_d^p, \tilde{y}_d^p, \tilde{z}_d^p)$.

$$\begin{pmatrix} x_u^n \\ y_u^n \\ 1 \end{pmatrix} = \frac{1}{f(\tilde{r})} \begin{pmatrix} \tilde{x}_d^p \\ \tilde{y}_d^p \\ f(\tilde{r}) \end{pmatrix} \quad (7)$$

With $\tilde{r} = \sqrt{\tilde{x}_d^p + \tilde{y}_d^p}$, $f(\tilde{r}) = a_0 + a_2\tilde{r}^2 + a_3\tilde{r}^3 + a_4\tilde{r}^4$ and a_0, a_2, a_3, a_4 the coefficients of the Scaramuzza model. Then, to obtain the real distorted pixel coordinates $\mathbf{X}_d^p = (x_d^p, y_d^p, z_d^p)$, we need a distortion matrix S and a center of distortion vector C.

$$\begin{pmatrix} x_d^p \\ y_d^p \end{pmatrix} = \underbrace{\begin{pmatrix} c & d \\ e & 1 \end{pmatrix}}_{S} \begin{pmatrix} \tilde{x}_d^p \\ \tilde{y}_d^p \end{pmatrix} + \underbrace{\begin{pmatrix} c_x \\ c_y \end{pmatrix}}_{C} \quad (8)$$

Implementation

To implement both distortion models, the method was :

1. To distort the foothold of the picture (its border points) using the direct distortion model to get the distorted foothold. This step was direct for the Brown-Conrady model ; it was more complex for the FishEye model, we based our implementation on OCamCalib Toolbox [5, 6, 7] .
2. For each pixel of the rectangular bounding box of the distorted foothold, to calculate its source point in the undistorted picture using the inverse distortion model and to interpolate its value. Unlike the previous step, the FishEye undistortion model can be used directly whereas to inverse the Brown-Conrady model, the work of Heikkilä [8], for instance, can be used. In the following example, we used a custom inversion algorithm.

For a given camera, to get its (un)distortion parameters, we used MATLAB® Calibrator App.

Validation

To validate this approach, we tried to reproduce a picture taken by a Mobius ActionCam camera, which introduces significant optical distortion (see Figure 4), using the FishEye model. Here are the optical parameters we used in our simulation :

Source pixel size	11.6 μm
Square root of radiance	$\sim$ 1 uSI
Distance between scene and aperture	0.56 m
Focal length	22 mm
Distortion polynom coefficients	1.57e+03,-2.75e-04, 3.76e-08,-3.64e-11

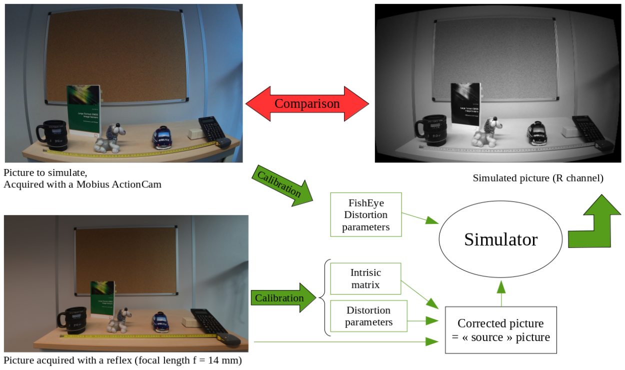

Comparison
Picture to simulate,
Acquired with a Mobius ActionCam
Simulated picture (R channel)
Calibration
FishEye Distortion parameters
Simulator
Calibration
Intrisic matrix
Distortion parameters
Corrected picture = « source » picture
Picture acquired with a reflex (focal length f = 14 mm)

Figure 4. *Optical distortion testing on Mobius ActionCam using FishEye distortion model. The difference between the real and the simulated pictures results from a misalignment of the cameras.*

Quantitative assessment of the precision was not possible because the position of both cameras during acquisition was not monitored accurately enough, we realized afterward that the optical axes and centers were not properly aligned. Thus, the geometric error is largely contaminated by this misalignment.

Photoconversion and noises modelization

After the optical module, the result is a ”spectral power density map”, that is a matrix representing the energy distribution in the focal plane. We then integrate this power in time (chosen integration duration) and space, converting it into a number of photons received by each photosite of the sensor that is simulated. The photosites can be defined as a classical matrix of pixels or as a list of pixels for each of which its size and position are given. This second option makes it possible to explore more exotic architectures such as the ones cited in [9]. Currently, only the integration mode of a sensor can be simulated, however it could be interesting to model spiking and other modes too, in the future.

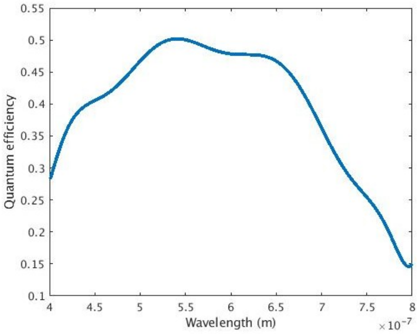


Figure 5. A typical polynomial approximation of Si quantum efficiency, derived from tabulated values found in [9]

After having converted the energy into photons, they are themselves converted into electrons through the quantum efficiency function. The quantum efficiency function simulated is a polynomial interpolation of tabulated values (see Figure 5). We could replace it by a simpler constant piecewise function or on the contrary by a more complex function. We can also add spectral filtering (so as to simulate bayerisation for instance) ; practically, it is just another filter whose effect mixes with quantum efficiency, and they can be implemented as one. The filters implemented can be seen on Figure 6.

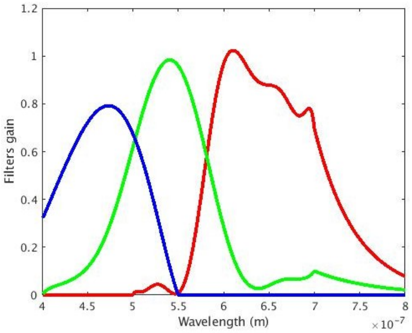


Figure 6. Bayerisation RGB filters

On Figure 7, we represent a caricatural example : we take Lena gray level picture as a mono-wavelength radiance map respectively at 450, 550 and 620 nm and we mix the 3 channels into one single bayerized RGB picture.

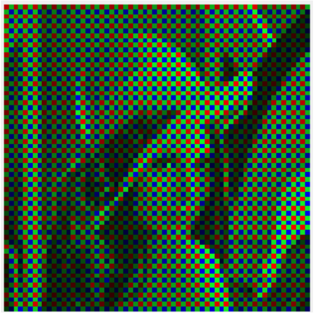


Figure 7. Bayerization example

The last step is to convert the electrons into a voltage value that can be saturated by a given threshold V_{sat} and quantized on N bits, V_{sat} and N being fixed by the user. Quantization is done on a linear scale but it would be interesting to make it variable, to account for high dynamic range sensors for example. Quantization and bayerisation are optional as is the consideration of the various noises described in the following section.

Noises

Several types of noises can be accounted for, see [10], for instance, so far we have implemented (see Figure 8) :

- *Photo-noise* : the number of photons that can actually be detected by the photosensitive elements follows a Poisson's law of parameter the theoretical number of photons that get to the photosensitive element's area. It can be approximated by a Gaussian law when this number is greater than a few tens, which is almost always true. Doing this approximation is computationally much more efficient than calculating the real Poisson's law.

- *kTC* : for 3T pixels, we consider kTC noise which is a zero average Gaussian noise whose standard deviation is kTC, with k the Boltzmann constant, T the temperature and C the capacity of the photosite.

- *Crosstalk* : We only consider electronic crosstalk that is due to the diffusion of electrons in the silicium before they are detected. We do not account for optical cross-talk, we suppose the integration time we want to simulate are long enough for it to be negligible. After trying some more complex models, it was finally chosen to model it as a simple convolution by a constant matrix whose values we found in [11] but can be changed according to the sensor that is simulated.

- *Readout noise* : it is a zero average Gaussian noise with parametrized standard deviation. This noise is predominant for low light levels.

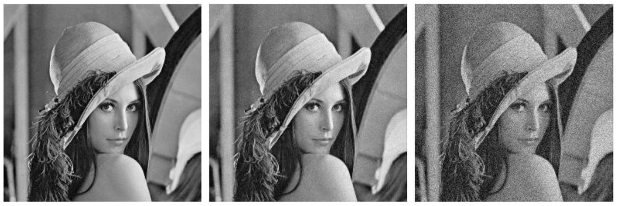


Figure 9. Variable readout noise. Readout = 0, 50, 200 e-. Maximum number of photons received by one pixel = 6000. Full well = 7000 e-.

- *Fixed Pattern Noise* : pixel-wise and column-wise Gaussian noise whose standard deviation we chose to set to a given percentage of the saturation voltage and whose mean we chose to consider as null. It originates from technological dispersion of the reading circuitry. For one sensor, we define a random matrix of the same size as the final picture, its coefficients follow a normal centered law whose standard deviation is 2% of the maximum voltage value. For columnar noise, we only need a random vector whose coefficients follow the same law and whose size is the number of columns in the final image matrix.

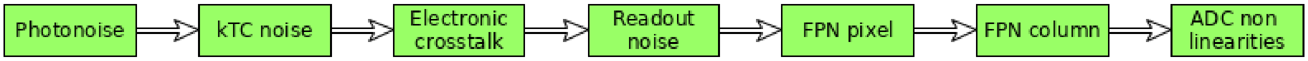


Figure 8. *Overview of the noises along the image sensor electronic chain*

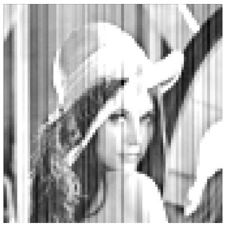


Figure 10. *Columnar Fixed Pattern Noise generated by the simulator*

- *ADC non ideality* : here we give the example of the impact of Differential Non-Linearity (DNL). For each quantized interval, for each column, we define a random value who follows a Gaussian centered law whose standard deviation is 0.5 the value of the least significant bit. This value is added to the pixel value before quantization.

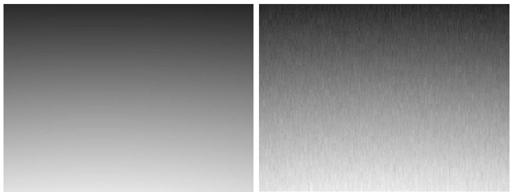


Figure 11. *Non realistic DNL noise example generated by the simulator. Left : without noise, right : with noise*

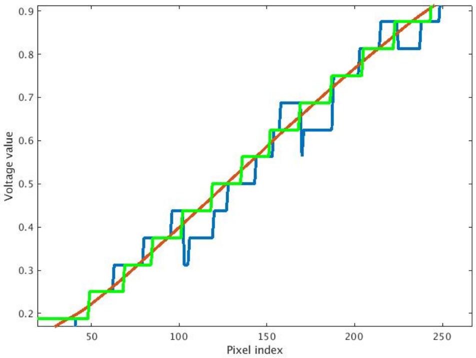


Figure 12. *Non realistic DNL noise example : one column of the previous image. Red : analog value, green : quantized value on 4 bits, blue : quantized value on 4 bits with DNL*

Examples of parameters values

	Foveal simulation	Distortion simulation
Integration time	1/30 s	1/60 s
Sensor pixel size	a few μm	2.2 μm
Readout noise	1 e-	1 e-
Capacity	$\sim 10^{-13}$ F	4.8 x 10^{-13} F
Voltage range	3 V	3 V
Number of bits	8	8

Applications and prospects

This simulator has been used to optimize a sensor's matrix of pixels size for given applications, for instance for an optical flow application. Let ε be the precision we need to ensure, M the dimension of the matrix and p_s the size of the pixels and let us say that the computation complexity of our application is linear in M. Thus, we want to minimize M - if we want to maintain the same foothold for the acquired picture, then we will have to increase p_s - under the constraint that the precision is greater then ε. The simulator is especially designed for this kind of application.

It would also be useful to inquire into unconventional architectures, such as foveal sensors, see Figure 13. The blind spot at the center of the picture is due to a design constraint we imposed on pixel minimum size.

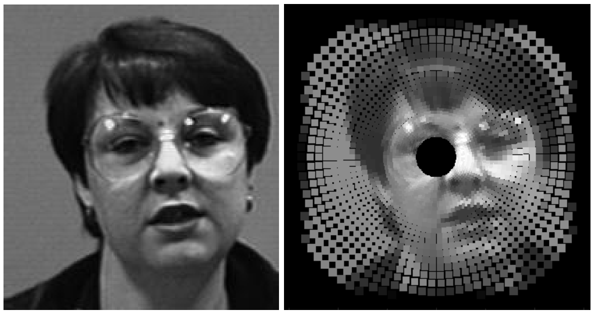


Figure 13. *Foveal sensor exploration*

A less advanced version of the simulator has been used to explore the design space of a multi-resolution sensor that has been designed and produced in our team [12].

All along the paper, we have seen the flexibility of the simulator, it makes it possible to add functions or enhance its user-friendliness for instance. Without using neither ray-tracing, nor multi-physical simulation, we get a good reliability for a moderate complexity. Indeed, complexity is $\mathcal{O}(\mathrm{N_{pix}}\log(\mathrm{N_{pix}})$ (With $\mathrm{N_{pix}}$ the number of pixels in the source picture) without taking into account distortion, it leads to generating images in less than a few seconds using MATLAB® (the largest source image we used was 4640x8256), even when we define unconventional pixels such as in the foveal geometry. For comparison, on the same picture (used in Figure 14), our simulator took 0.3 s for the whole process whereas ISET took around 20 s for the Optics and Sensor modules to run. When simulating distortion, the computation with our simulator took up to about 6 minutes.

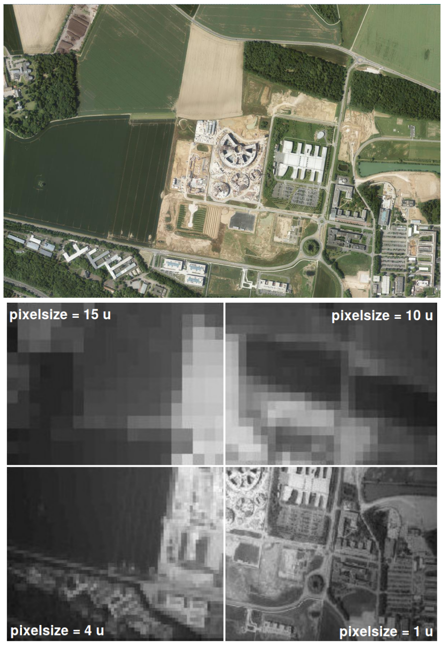


Figure 14. *4 zones multi-resolution sensor simulation [2]. [Source image : Géoportail]*

The scope of possibilities to use this tool is wide open, and could also include data augmentation for deep learning in classification and denoising. Indeed, for offline training of neural networks, it can be useful to have numerous images taken under different conditions or to be able to take into account artefacts resulting from image sensors limitations.
We are currently considering making our code open-source.

References

[1] J. Farrell, P. B. Catrysse and B. Wandell. Digital camera simulation. Applied Optics. Optical Society of America (2012).

[2] A. de Gouvello, L. Soulier, A. Dupret, Image acquisition process simulator, GRETSI. Lille, France (August 26-29, 2019).

[3] E. Aristidi. Diffraction. http://sites.unice.fr/site/aristidi/optique/.

[4] Duane C. Brown. Decentering distortion of lenses. Photogrammetric Engineering and Remote Sensing (1966).

[5] D. Scaramuzza, A. Martinelli, R. Siegwart. A Toolbox for Easy Calibrating Omnidirectional Cameras. Proceedings to IEEE International Conference on Intelligent Robots and Systems (IROS). Beijing, China (October 7-15, 2006).

[6] D. Scaramuzza, A. Martinelli and R. Siegwart. A Flexible Technique for Accurate Omnidirectional Camera Calibration and Structure from Motion. Proceedings of IEEE International Conference of Vision Systems (ICVS'06). New York (January 5-7, 2006).

[7] M. Rufli, D. Scaramuzza and R. Siegwart. Automatic Detection of Checkerboards on Blurred and Distorted Images. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008). Nice, France (September 2008).

[8] Janne Heikkilä. Geometric camera calibration using circular control points. IEEE Transactions on pattern analysis and machine intelligence (2000).

[9] Alireza Moini, Vision Chips, Kluwer Academic Publishers (1999).

[10] S. U. Ay. Large Format CMOS Image Sensors : Performance and Design. VDM, Verlag Dr. Müller (2008).

[11] Ryan D. Gow et al. A comprehensive tool for modeling CMOS image-sensor-noise performance. IEEE Transactions on Electron Devices (2007).

[12] L. Millet et al. A 5500-frames/s 85-GOPS/W 3-D stacked BSI vision chip based on parallel in-focal-plane acquisition and processing. IEEE Journal of Solid-State Circuits (2019).

Author Biography

Alix de Gouvello (alix.degouvello@cea.fr) received her MS degree in engineering from University Pierre and Marie Curie, France (2018) and she is an engineer from ENSTA ParisTech, France (2018). Since then she has been working as a PhD student in the Algorithm and Architecture Codesign Laboratory at CEA LIST, in Gif-sur-Yvette, France. Her work is focused on bio-inspired sensor design and image processing.

Laurent Soulier (laurent.soulier@cea.fr) is an engineer from the Ecole Polytechnique (Palaiseau, France), with a specialization in fluid mechanics and electronics, components and systems. He is also an engineer from Telecom Paristech (previously Ecole Nationale des Telecommunications), with a specialization in signal and image processing and a member of Corps des Mines. He is an expert in image processing and software engineering.

Antoine Dupret (antoine.dupret@cea.fr) graduated in electrical engineering from the Ecole Normale Suprieure de Cachan and Universit de Paris Sud 11, France (MS 91, PhD 95). He was assistant professor with the Universit de Paris 13, then full professor at ESIEE Engineering in 2009. He joined CEA in 2010 as senior expert in Image Sensors, and was the head of the Sensor Integration and Reliability Lab. He is author or co-author of over 100 papers in International Conferences and Journals and program chair of several international conferences.

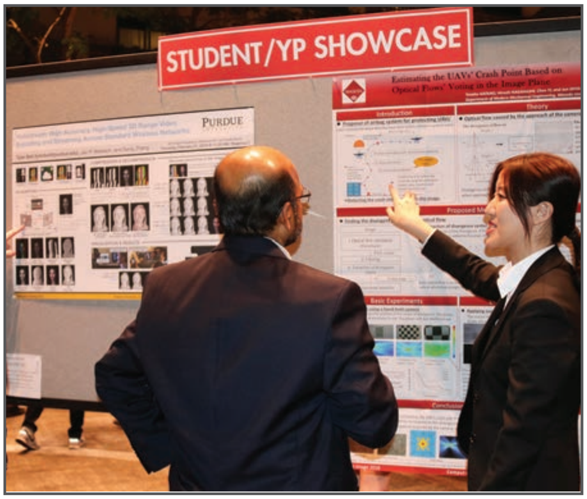
JOIN US AT THE NEXT EI!

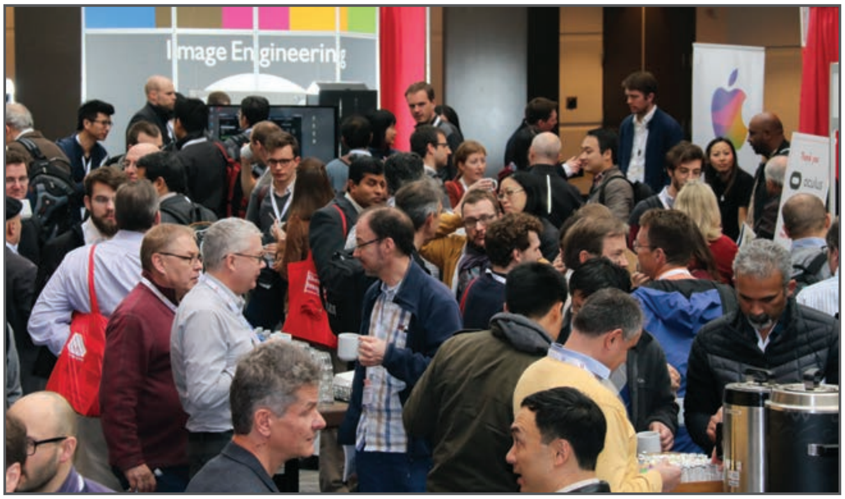
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





- AR is a fundamental and transformative interface and platform.
- 20+ years from my initial working demo to today's AR displays
- Optimistic that it won't take another 20 to reach ubiquitous consumer AR!

• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org