Received 6 January 2023; revised 10 May 2023; accepted 25 June 2023. Date of publication 27 June 2023; date of current version 17 July 2023.
The review of this article was arranged by Editor S. Krishnamoorthy.

Digital Object Identifier 10.1109/JEDS.2023.3290106

Photon Counting Histogram Expectation Maximization Algorithm for Characterization of Deep Sub-Electron Read Noise Sensors

AARON J. HENDRICKSON[1] AND DAVID P. HAEFNER[2]

1 U.S. Navy, NAWCAD DAiTA Group, Patuxent River, MD 20670, USA
2 U.S. Army, C51SR Center, Fort Belvoir, VA 22060, USA

CORRESPONDING AUTHOR: A. J. HENDRICKSON (e-mail: ajh4184@gmail.com)

ABSTRACT We develop a novel algorithm for characterizing Deep Sub-Electron Read Noise (DSERN) image sensors. This algorithm is able to simultaneously compute maximum likelihood estimates of quanta exposure, conversion gain, bias, and read noise of DSERN pixels from a single sample of data with less uncertainty than the traditional photon transfer method. Methods for estimating the starting point of the algorithm are also provided to allow for automated analysis. Demonstration through Monte Carlo numerical experiments are carried out to show the effectiveness of the proposed technique. In support of the reproducible research effort, all of the simulation and analysis tools developed are available on the MathWorks file exchange (Hendrickson and Haefner, 2022).

INDEX TERMS Clustering algorithms, conversion gain, DSERN, expectation maximization, Gaussian mixture, PCH, PCH-EM, photon counting, photon transfer, quanta exposure, read noise.

I. INTRODUCTION

Advances in image sensor technology have resulted in pixels with sufficiently low read noise; enabling the ability to discern the presence of individual electrons without the need for avalanche gain or electron multiplication [2], [3], [4], [5]. Sensors with this property, aptly named Deep Sub-Electron Read Noise (DSERN) sensors, open the door to new applications for the CMOS sensor architecture in ultra low-light imaging environments. With the first DSERN sensors now commercially available, methodologies for characterizing these devices has become an emerging topic of interest [6], [7].

To date, three methods for characterizing DSERN sensors have been developed including the Photon Transfer (PT) method [8], [9], [10], Photon Counting Histogram (PCH) method [6], and a third method based on constrained Maximum Likelihood Estimation (MLE) [7]. The work by Fossum & Starkey [6]–and subsequently Nakamoto & Hotaka [7]–show promise that the PCH and constrained MLE methods out-perform the traditional PT method in accuracy and precision when applied to DSERN pixels.

While promising, both methods present challenges related to numerical stability, computational cost, and/or autonomy. In this correspondence we present a new method in the form of the PCH Expectation Maximization (PCH-EM) algorithm, which incorporates attributes of both the PCH and constrained MLE methods to enable a fully automated characterization technique for maximum likelihood estimation of quanta exposure, conversion gain, bias, and read noise of DSERN pixels.

Prior to presenting the PCH-EM algorithm we must, however, first develop a statistical model for the digital signal produced by DSERN pixels.

II. THE PHOTON COUNTING DISTRIBUTION MODEL

We begin by deriving the Photon Counting Distribution (PCD) as a model of the probability density for data generated by DSERN pixels. The number of free-electrons generated in a DSERN pixel when exposed to a constant rate of impinging photons can be modeled by the Poisson random variable $K \sim \text{Poisson}(H)$. Here, $H = H_\gamma + H_d$ denotes the *quanta exposure* describing the expected number

of free-electrons generated in the pixel per integration time, which is further decomposed into H_γ (the expected number of photoelectrons generated by interacting photons) and H_d (the expected number of free-electrons generated by thermal contributions, i.e., dark current). The act of sensing the electron signal introduces a continuous read noise component $R \sim \mathcal{N}(0, \sigma_R^2)$, where σ_R is the input-referred analog read noise in $(e-)$. The pixel output signal in Digital Numbers (DN) can thus be represented by the random variable

$$X = \lceil (K + R)/g + \mu \rfloor, \tag{1}$$

where g is the conversion gain in $(e-/\text{DN})$, μ is the bias (DC offset) in units of (DN), and $\lceil \cdot \rfloor$ denotes rounding to the nearest integer.

To derive the PCD, we will model the act of quantization (rounding) as a simple additive noise process so that $X|K = k \sim \mathcal{N}(\mu + k/g, \sigma^2)$, where $\sigma = (\sigma_R^2/g^2 + \sigma_Q^2)^{1/2}$ is the combined read and quantization noise in (DN). To avoid confusion, we note that when characterizing an image sensor, the quantity σ is commonly referred to as the read noise with σg being its corresponding value in electron units. We may now obtain the PCD by integrating the join density $f_{XK}(x, k) = \mathsf{P}(K = k) f_{X|K}(x|k)$ w.r.t. k yielding

$$f_X(x) = \sum_{k=0}^{\infty} \frac{e^{-H} H^k}{k!} \phi\Big(x; \mu + k/g, \sigma^2\Big), \tag{2}$$

where $\phi(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-(x-\mu)^2/2\sigma^2)$ is the Gaussian probability density with mean μ and variance σ^2. For notational purposes we will use $X \sim \text{PCD}(H, g, \mu, \sigma^2)$ to denote a PCD random variable with parameter $\theta = (H, g, \mu, \sigma^2)$ as described by (2).

From (2) we see that the PCD is an infinite mixture of Gaussian components with g controlling the spacing between each component, σ as the width of each component, μ the location of the zeroth component, and H the relative heights of the components. Figure 1 plots the PCD for various H and σ^2 with $\mu = 0$ and $g = 1$ fixed. From the figure we can see how the parameter H controls the overall envelope of the distribution, while σ determines whether the individual peaks may be resolved. As shown in the right column of Figure 1, as σ increases, the contrast of the individual peaks is reduced. Typically, the term DSERN is assigned to pixels where σ is small enough such that the peaks are clearly resolved [3].

III. THE PCH-EM ALGORITHM

Looking back at (2), we can see that the PCD can be interpreted as the marginal density resulting from marginalizing K out of the joint model (X, K):

$$f_{XK}(x, k) = \frac{e^{-H} H^k}{k!} \phi\Big(x; \mu + k/g, \sigma^2\Big). \tag{3}$$

This reflects real-world data in the sense that we are only able to observe the pixel output X, while the number of free electrons K is unknown. For this reason K is a hidden (latent)

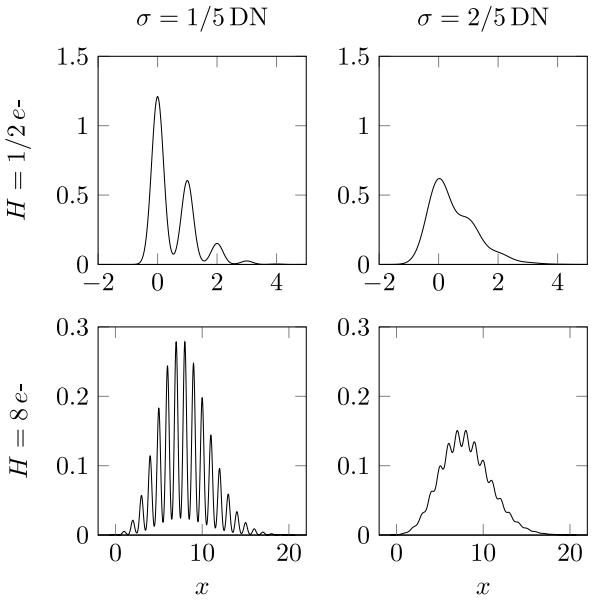

FIGURE 1. Plots of the PCD for various H and σ^2 with $\mu = 0$ and $g = 1$ fixed.

variable in the PCD model. To see what impact hidden variables has on our approach for estimating θ, Appendix A provides a derivation of the maximum likelihood estimator for θ in the hypothetical scenario where K can be directly observed. As one can see in the Appendix, the ability to directly observe K rather unexpectedly results in a simple, closed-form estimator for θ.

In the realistic scenario where K is hidden, maximum likelihood estimation of θ becomes much less straightforward. To see why, we first denote $\mathbf{x} = (x_1, \ldots, x_N)$, with $x_n \sim \text{PCD}(H, g, \mu, \sigma^2)$, as a sample of N observed values from a DSERN pixel. From this sample we may construct a likelihood function

$$L(\theta|\mathbf{x}) := f_X(\mathbf{x}|\theta) = \sum_{\mathcal{K}} f_{XK}(\mathbf{x}, \mathbf{k}|\theta), \tag{4}$$

where we have used the shorthand notation $\mathbf{k} = (k_1, \ldots, k_N)$, $\sum_{\mathcal{K}} = \sum_{k_1=0}^{\infty}, \ldots \sum_{k_N=0}^{\infty}$, and $f_{XK}(\mathbf{x}, \mathbf{k}|\theta) = \prod_{n=1}^{N} f_{XK}(x_n, k_n|\theta)$. Denoting $\ell(\theta|\mathbf{x}) = \log f_X(\mathbf{x}|\theta)$ as the log-likelihood function our goal is to solve the density estimation problem

$$\tilde{\theta} = \arg\max_{\theta} \ell(\theta|\mathbf{x}). \tag{5}$$

Deriving the log-likelihood we find

$$\ell(\theta|\mathbf{x}) = \sum_{n=1}^{N} \log \sum_{k=0}^{\infty} \frac{e^{-H} H^k}{k!} \phi\Big(x_n; \mu + k/g, \sigma^2\Big), \tag{6}$$

which is problematic for deriving closed-form maximum likelihood estimators due to the series inside the logarithm. In situations like this, we can directly maximize $\ell(\theta|\mathbf{x})$ through numerical methods such as gradient descent; however, this may be undesirable as one must calculate derivatives of the

likelihood function and carefully control step size to ensure convergence.

A lesser-known, yet widely accepted, method for maximum likelihood estimation is that of the Expectation Maximization (EM) algorithm [11]. Instead of maximizing the log-likelihood function directly, the EM algorithm maximizes a related (often simpler) function to produce a sequence of estimators that converge to those of the maximum likelihood estimators. The key to the success of this method is that EM models the estimation problem as one containing hidden variables. Throughout the remainder of this section, we derive a custom-built EM algorithm for characterizing DSERN pixels, which takes into account the specific structure of the PCD in (2).

A. DERIVATION OF THE PCH-EM ALGORITHM

A.1. SUPPORTING THEORY

The key insight provided by the PCH-EM algorithm is that the log-likelihood can be written in an alternative form only made possible by knowing that the PCD model contains hidden variables. This alternative form then allows us to derive update equations that improve an initial estimate of θ in such a way as to guarantee an increase in the log-likelihood.

To derive this alternative form of the log-likelihood we first call on the definition of conditional density to write

$$p_{K|X}(\mathbf{k}|\mathbf{x},\theta) = \frac{f_{XK}(\mathbf{x},\mathbf{k}|\theta)}{f_X(\mathbf{x}|\theta)}, \tag{7}$$

which upon taking the logarithm and rearranging gives

$$\ell(\theta|\mathbf{x}) = \ell(\theta|\mathbf{x},\mathbf{k}) - \log p_{K|X}(\mathbf{k}|\mathbf{x},\theta), \tag{8}$$

where $\ell(\theta|\mathbf{x},\mathbf{k}) = \log f_{XK}(\mathbf{x},\mathbf{k}|\theta)$. Multiplying both sides of this relation by $p_{K|X}(\mathbf{k}|\mathbf{x},\theta')$ and summing over $\mathcal{K}$ we obtain the following alternative representation of the log-likelihood function that holds for any θ' in the PCD parameter space [12]

$$\ell(\theta|\mathbf{x}) = \mathsf{E}_{\theta'}(\ell(\theta|\mathbf{x},\mathbf{K})) - \mathsf{E}_{\theta'}\big(\log p_{K|X}(\mathbf{K}|\mathbf{x},\theta)\big). \tag{9}$$

Denoting the *expected complete-data log-likelihood* as the first term in (9), namely,

$$Q\Big(\theta|\theta^{(t)}\Big) = \mathsf{E}_{\theta^{(t)}}(\ell(\theta|\mathbf{x},\mathbf{K})), \tag{10}$$

the PCH-EM algorithm takes an initial estimate for the parameter $\theta^{(0)} = (H^{(0)}, g^{(0)}, \mu^{(0)}, \sigma^{2(0)})$ and iterates between two steps:

1) The expectation (E) step to compute $Q(\theta|\theta^{(t)})$ and
2) The Maximization (M) step, which maximizes Q to update the estimate via

$$\theta^{(t+1)} = \arg\max_{\theta} Q\Big(\theta|\theta^{(t)}\Big). \tag{11}$$

Updating the parameter estimate in this way it is guaranteed in each iteration that [13]

$$\ell\Big(\theta^{(t+1)}|\mathbf{x}\Big) \geq \ell\Big(\theta^{(t)}|\mathbf{x}\Big) \tag{12}$$

so that a local maxima of the log-likelihood is always achieved. With the supporting theory we now present the details of the E- and M-step in the PCH-EM algorithm.

A.2. E-STEP

The expectation step entails deriving $Q(\theta|\theta^{(t)})$. Substituting appropriate values into (10) yields

$$Q\Big(\theta|\theta^{(t)}\Big) = \sum_{\mathcal{K}} p_{K|X}\Big(\mathbf{k}|\mathbf{x},\theta^{(t)}\Big)\log f_{XK}(\mathbf{x},\mathbf{k}|\theta), \tag{13}$$

which upon further expanding gives

$$Q\Big(\theta|\theta^{(t)}\Big) = \prod_{m=1}^{N}\sum_{k_m=0}^{\infty} p_{K|X}\Big(k_m|x_m,\theta^{(t)}\Big) \times \sum_{n=1}^{N}\log f_{XK}(x_n,k_n|\theta). \tag{14}$$

Bringing the sum w.r.t. n to the outside results in a lot of simplification. After interchanging the sums and substituting the expression for $f_{XK}(x_n,k|\theta)$ we obtain

$$Q\Big(\theta|\theta^{(t)}\Big) = \sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}\log\left(\frac{e^{-H}H^k}{k!}\phi\Big(x_n;\mu+k/g,\sigma^2\Big)\right), \tag{15}$$

where $\gamma_{nk}^{(t)} = p_{K|X}(k|x_n,\theta^{(t)})$ and

$$p_{K|X}(k|x_n,\theta) = \frac{\frac{e^{-H}H^k}{k!}\phi\big(x_n;\mu+k/g,\sigma^2\big)}{\sum_{m=0}^{\infty}\frac{e^{-H}H^m}{m!}\phi\big(x_n;\mu+m/g,\sigma^2\big)}. \tag{16}$$

Comparing the final expression in (15) to the complete data log-likelihood of equation (39) in the Appendix shows many similarities. The major difference between these expressions is that the $\mathbb{1}_{k_n=k}$ has been substituted for $\gamma_{nk}^{(t)}$. In the case where K was known, the indicator function $\mathbb{1}_{k_n=k}$ can be interpreted as a degenerate probability distribution centered on the known value for each observed k_n. In comparison, in the PCH-EM algorithm we do not know K, so this degenerate distribution is replaced with $\gamma_{nk}^{(t)}$, representing the probability $K_n = k$ given the observed data $X_n = x_n$ and the current estimate of the parameter $\theta^{(t)}$. For this reason, the $\gamma_{nk}^{(t)}$ are commonly referred to as *membership probabilities* as they assign the probability of x_n belonging to each Gaussian component of the PCD.

A.3. M-STEP

Maximization of $Q(\theta|\theta^{(t)})$ is a very similar process to maximizing the complete log-likelihood in (39). To maximize this function, we solve for the critical point $\nabla_\theta Q = 0$. Solving this system of equations is fairly straightforward yet tedious. A proof can be found in Appendix B. Upon solving the system, we obtain the collection of update equations

$$H^{(t+1)} = A^{(t)} \tag{17a}$$

$$g^{(t+1)} = \frac{B^{(t)} - H^{2(t+1)}}{C^{(t)} - \bar{x}H^{(t+1)}} \tag{17b}$$

$$\mu^{(t+1)} = \bar{x} - \frac{H^{(t+1)}}{g^{(t+1)}} \tag{17c}$$

$$\sigma^{2(t+1)} = \hat{x} - \frac{B^{(t)} - H^{2(t+1)}}{g^{2(t+1)}}, \tag{17d}$$

where $\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$ is the sample mean, $\hat{x} = \frac{1}{N}\sum_{n=1}^{N}(x_n - \bar{x})^2$ is the sample variance, and

$$A^{(t)} = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}k \tag{18a}$$

$$B^{(t)} = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}k^2 \tag{18b}$$

$$C^{(t)} = \frac{1}{N}\sum_{n=1}^{N}x_n\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}k. \tag{18c}$$

Again, comparing these update equations to the closed-form maximum likelihood estimates in (41) shows many similarities as is expected. To check and see if these update equations make sense, notice that $A^{(t)}$, $B^{(t)}$, and $C^{(t)}$ have the form of Monte Carlo estimators for expected values w.r.t. $X \sim \mathrm{PCD}(\theta)$, e.g.,

$$A^{(t)} \sim \mathsf{E}_\theta\left(\sum_{k=0}^{\infty} p_{K|X}\left(k|X,\theta^{(t)}\right)k\right) \tag{19}$$

as $N \to \infty$. Assuming we also have a good starting point so that $\theta^{(t)} \to \theta$ then produces the asymptotic approximations for large N and large iteration number t

$$A^{(t)} \sim \mathsf{E}K = H \tag{20a}$$

$$B^{(t)} \sim \mathsf{E}K^2 = H^2 + H \tag{20b}$$

$$C^{(t)} \sim \mathsf{E}(KX) = \frac{1}{g}\left(H^2 + (1+\mu g)H\right). \tag{20c}$$

Likewise, the sample moments act as Monte Carlo estimators for the exact moments as $N \to \infty$ giving

$$\bar{x} \sim \mathsf{E}X = \mu + \frac{H}{g} \tag{21a}$$

$$\hat{x} \sim \mathsf{Var}X = \sigma^2 + \frac{H}{g^2}. \tag{21b}$$

Substituting these asymptotic approximations into the right hand sides of (17) show that the updates are asymptotic to the parameters they estimate, e.g., $H^{(t+1)} \sim H$, $g^{(t+1)} \sim g$, and so on.

One additional benefit of the PCH-EM algorithm is that it also provides a means for estimating (demarginalizing) the hidden variable K via the membership probabilities. Since $\gamma_{nk}^{(t)}$ represents a probability distribution for K_n given $X_n = x_n$, we may estimate the hidden values k_n via

$$\tilde{k}_n = \arg\max_k \gamma_{nk}^{(t)} \tag{22}$$

and thus predict the number of free-electrons that generated each observation x_n.

A.4. FINAL ALGORITHM

With the derivation of the E- and M-step now complete, the PCH-EM algorithm works by supplying a starting point $\theta^{(0)} = (H^{(0)}, g^{(0)}, \mu^{(0)}, \sigma^{2(0)})$, then:

1) E: Compute $\gamma_{nk}^{(t)}$ by substituting $\theta^{(t)}$ into (16).
2) M: Update $\theta^{(t)} \mapsto \theta^{(t+1)}$ with (17).
3) Repeat until $Q(\theta^{(t+1)}|\theta^{(t)}) - Q(\theta^{(t)}|\theta^{(t-1)}) \le \epsilon$.

IV. STARTING POINT ESTIMATION

As is the case with numerical optimization methods, one of the challenges when working with the PCH-EM algorithm is the need for a starting point. While the PCH-EM update equations guarantee an increase in log-likelihood at each iteration, the effectiveness of the algorithm in achieving a global maximum, as compared to a local maximum, is sensitive to the initial starting point. Here we present a method for extracting the starting point directly from the original sample by relying on the properties of the PCD Fourier transform.

A. PROPERTIES OF THE PCD FOURIER TRANSFORM

We begin by considering the magnitude of the PCD Fourier transform $\mathcal{F}\{f_X\}(\omega) \coloneqq \mathsf{E}\exp(-2\pi i\omega X)$

$$|\mathcal{F}\{f_X\}(\omega)| = \exp\left(H(\cos(2\pi\omega/g) - 1) - 2\pi^2\sigma^2\omega^2\right). \tag{23}$$

The magnitude function in (23) is asymptotic to a Gaussian curve near integer multiples of g in the sense that as $\omega \to ng$ $(n = 0, 1, 2, \ldots)$

$$|\mathcal{F}\{f_X\}(\omega)| \sim a_n \exp\left(-\tau(\omega - b_n)^2\right), \tag{24}$$

with $\tau = 2\pi^2(\sigma^2 + H/g^2)$,

$$a_n = \exp\left(-2\pi^2\left(Hn^2 - \frac{(Hn)^2}{g^2\left(H/g^2 + \sigma^2\right)}\right)\right), \tag{25}$$

and

$$b_n = \frac{Hn}{g\left(H/g^2 + \sigma^2\right)}. \tag{26}$$

From this observation we expect to find local maxima (peaks) in the magnitude function at $\omega \approx b_n$. Figure 2 depicts a graph of $|\mathcal{F}\{f_X\}(\omega)|$ showing the two most dominant peaks at $\omega = 0$ and $\omega \approx b_1$ along with the approximate position of the secondary peak $(\omega, |\mathcal{F}\{f_X\}(\omega)|) = (b_1, a_1)$.

The existence of the secondary peak at $\omega \approx b_1$ depends on the values of quanta exposure and read noise. To see why, we evaluate $\partial_\omega|\mathcal{F}\{f_X\}(\omega)| = 0$, which given $\omega > 0$ simplifies to

$$\mathrm{sinc}(2\pi\omega/g) + \frac{(\sigma g)^2}{H} = 0, \tag{27}$$

where $\mathrm{sinc}\, x = \sin x/x$. In order for a secondary peak to exist, this equation must have at least two solutions on $\omega > 0$, which occurs only when

$$\frac{(\sigma g)^2}{H} < |\min_{x>0}\mathrm{sinc}\, x| = 0.217\ldots \tag{28}$$

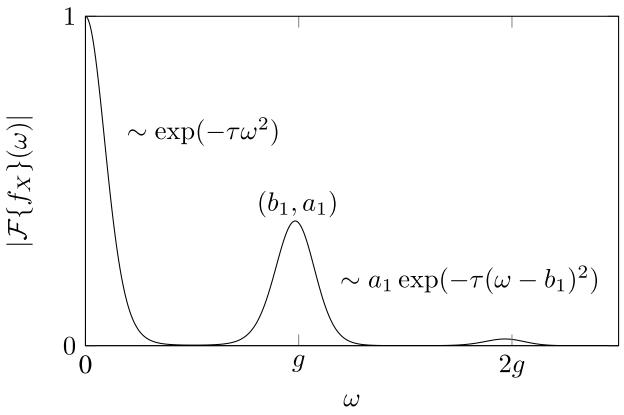

FIGURE 2. Graph of $|\mathcal{F}\{f_X\}(\omega)|$ versus ω showing the two most dominant peaks at $\omega = 0$ and $\omega \approx b_1$ along with the approximate position of the secondary peak (b_1, a_1).

In other words, the magnitude function at $\omega \approx b_1$ is only a peak if H is about five times larger than $(\sigma g)^2$. When this is not the case, we cannot reliably extract the starting point from the observed data. From the properties outlined here we can extract the starting point as follows.

B. STARING POINT EXTRACTION PROCEDURE

Given a set of N observations $\mathbf{x} = (x_1, \ldots, x_N)$ with $x_n \sim \text{PCD}(H, g, \mu, \sigma^2)$, we begin the process of extracting the starting point $\theta^{(0)}$ by creating a density normalized PCH via

$$\tilde{f}_X(n) = \frac{1}{N}\sum_{k=1}^{N} \mathbb{1}_{x_k=n}, \tag{29}$$

where $n \in \{\min(\mathbf{x}), \min(\mathbf{x})+1, \ldots, \max(\mathbf{x})\}$. The number of bins in the density normalized PCH is denoted $N_b = \max(\mathbf{x}) - \min(\mathbf{x}) + 1$. Next, we evaluate the Discrete Fourier transform (DFT)

$$\mathcal{F}\left\{\tilde{f}_X\right\}(\omega_n) = \sum_k \tilde{f}_X(k)\exp(-2\pi i(k-1)\omega_n), \tag{30}$$

where $\omega_n = (n-1)/N_b$.

The location of the secondary peak in the DFT magnitude $|\mathcal{F}\{\tilde{f}_X\}(\omega_n)|$ yields a single point, which encodes estimates of a_1 and b_1, namely, $(\omega_{\text{peak}}, |\mathcal{F}\{\tilde{f}_X\}(\omega_{\text{peak}})|) = (\tilde{b}_1, \tilde{a}_1)$. Likewise, an estimate for τ is obtained via $\tilde{\tau} = 2\pi^2\hat{x}$, where $\hat{x} = \frac{1}{N-1}\sum_{n=1}^{N}(x_n - \bar{x})^2$. Equating $\tilde{a}_1$, $\tilde{b}_1$, and $\tilde{\tau}$ with their exact expressions yields a system of equations that can be inverted to obtain initial estimates of H, g, and σ^2:

$$\tilde{H} = \frac{\tilde{b}_1^2\tilde{\tau} - \log\tilde{a}_1}{2\pi^2} \tag{31a}$$

$$\tilde{g} = 2\pi^2\frac{\tilde{H}}{\tilde{b}_1\tilde{\tau}} \tag{31b}$$

$$\tilde{\sigma}^2 = \frac{\tilde{\tau}}{2\pi^2} - \frac{\tilde{H}}{\tilde{g}^2}. \tag{31c}$$

TABLE 1. Simulation parameters.

Quantity	Symbol	Value
sample size	N	1000 $(-)$
dark current	i_d	0.12 $(e\text{-}/s)$
integration time	t	0.1 (s)
dark quanta exposure	$H_d = i_d \times t$	0.012 $(e\text{-})$
photon quanta exposure	H_γ	6.858 $(e\text{-})$
total quanta exposure	$H = H_\gamma + H_d$	6.87 $(e\text{-})$
bias	μ_R	1.82 $(e\text{-})$
read noise	σ_R	0.26 $(e\text{-})$
conversion gain	g	0.0433 $(e\text{-}/\text{DN})$
quantization noise	σ_Q	0.2887 (DN)
bias (sensor units)	$\mu = \mu_R/g$	42 (DN)
read noise (sensor units)	$\sigma = ((\sigma_R/g)^2 + \sigma_Q^2)^{1/2}$	6.0069 (DN)

The final starting points $H^{(0)}$, $g^{(0)}$, and $\sigma^{2(0)}$ are then computed by fitting the full model (23) to $|\mathcal{F}\{\tilde{f}_X\}(\omega_n)|$ using nonlinear least squares with $\tilde{H}$, $\tilde{g}$, and $\tilde{\sigma}^2$ as starting values.

As for the starting value of μ, we obtain an initial estimate via

$$\tilde{\mu} = \bar{x} - \frac{H^{(0)}}{g^{(0)}}. \tag{32}$$

If the estimate $\tilde{\mu}$ differs from the exact value of μ by approximately $1/(2g)$, e.g., $\tilde{\mu} \approx \mu \pm 1/(2g)$, supplying this estimate to the PCH-EM algorithm will result in slow convergence. This can be mostly circumvented by first constructing the autocorrelation function

$$R(t) = \sum_k \tilde{f}_X(k) f_X\left(k - t|\tilde{\mu}, H^{(0)}, g^{(0)}, \sigma^{2(0)}\right) \tag{33}$$

and then extracting a correction factor

$$\text{correction} = \arg\max_t R(t) \tag{34}$$

to refine the initial estimate via

$$\mu^{(0)} = \tilde{\mu} + \text{correction}. \tag{35}$$

V. IMPLEMENTATION AND EXAMPLES

To demonstrate the utility of the derived results, the starting point algorithm as well as the PCH-EM algorithm were implemented in MATLAB.[1] This code can be freely downloaded from the MathWorks file exchange [1]. Using this code we simulated the TPG jot presented in [6] with the parameters in Table 1. A sample of $N = 1000$ observations for the simulated jot were generated according to (1), namely, $X = \lceil (K+R)/g \rfloor$, where $K \sim \text{Poisson}(H)$ and $R \sim \mathcal{N}(\mu_R, \sigma_R)$.

The first step in the process is to generate the starting point. The algorithm for generating the starting point requires no user input and only accepts the raw sensor data. Figure 3 presents the result of the starting point algorithm showing the simulated PCH DFT magnitude compared to the magnitude of the exact PCD Fourier transform and the estimated fit using our starting point algorithm.

1. For a comprehensive comparative analysis of PCH-EM, PT, and constrained MLE methods see [14].

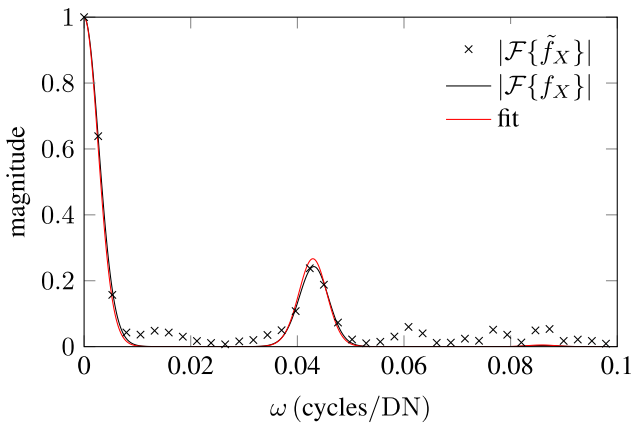

FIGURE 3. Magnitude of PCH DFT ($|\mathcal{F}\{\tilde{f}_X\}|$) compared against the exact PCD Fourier transform magnitude ($|\mathcal{F}\{f_X\}|$) and fitted magnitude.

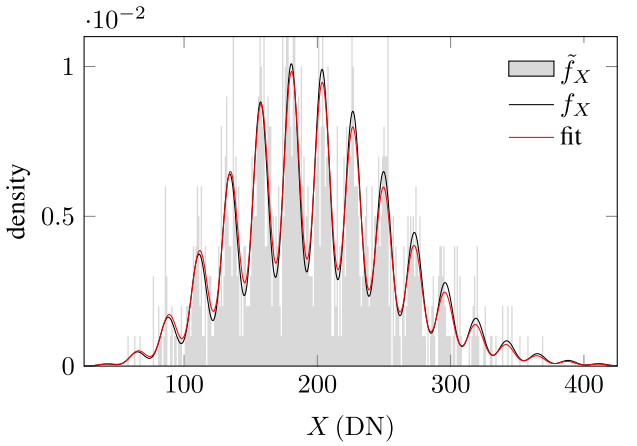

FIGURE 4. Simulated PCH data ($\tilde{f}_X$) versus the exact PCD (f_X) and fitted PCD generated by the PCH-EM algorithm.

TABLE 2. Estimated parameters for simulated jot.

Quantity	Estimate	Exact	Error
H (e-)	6.7394	6.87	1.90%
g (e-/DN)	0.0435	0.0433	0.46%
μ (DN)	42.869	42	2.07%
σ (DN)	6.2585	6.0069	4.19%

The estimated starting point, along with the raw data $\mathbf{x} = (x_1, \ldots, x_{1000})$, were then fed into the PCH-EM algorithm. The algorithm converged in just two iterations, which took approximately 0.042 seconds to complete running on a Intel Core i7 processor. Figure 4 plots the simulated PCH data along with the exact PCD (according to Table 1) and fitted PCD generated by the PCH-EM algorithm parameter estimate. Table 2 also shows the estimated parameters along with their exact values and percent error. From these results we can see that we were successfully able to estimate all four parameters from just 1000 observations. In particular, we were able to estimate the conversion gain with less than 1% error. From these estimates we were also able to estimate the read noise σg yielding the value 6.2585 $e-$, which

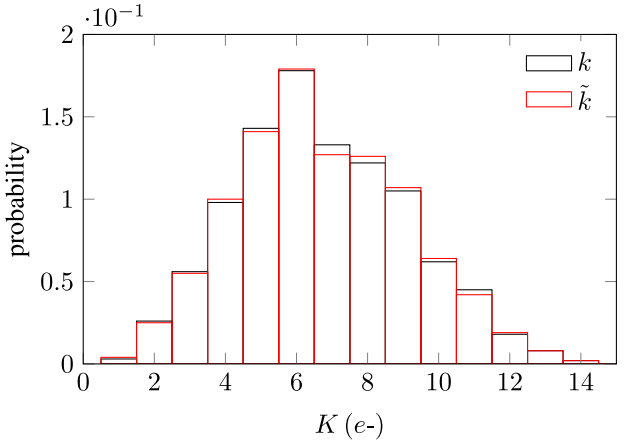

FIGURE 5. Histogram of estimates for the hidden variable K compared to the histogram of their exact values.

TABLE 3. Comparison of starting values and final parameter estimates to exact value for $N = 1000$.

Quantity	Exact	Start	Final
H (e-)	6.87	6.8419 ± 0.3380	6.8515 ± 0.3629
g (e-/DN)	0.0433	0.0433 ± 0.0002	0.0433 ± 0.0002
μ (DN)	42	42.4309 ± 8.1255	42.4042 ± 8.0173
σ^2 (DN^2)	36.083	35.8996 ± 2.1886	36.0569 ± 2.1298

results in 4.67% error. For comparison, we generated a second dark sample of $M = 1000$ observations according to the PCD model $Y = \lceil (K + R)/g \rfloor$, where $K \sim \text{Poisson}(H_d)$ and $R \sim \mathcal{N}(\mu_R, \sigma_R^2)$. We then calculated the conversion gain using the two-sample PT gain estimator described in [10]

$$\tilde{g} = \frac{\bar{x} - \bar{y}}{\hat{x} - \hat{y}}, \tag{36}$$

where $\bar{x}$ is the sample mean of the illuminated data, $\hat{x}$ is the sample variance of the illuminated data, and likewise for the dark data. This estimator gave an estimate of the conversion gain equal to $\tilde{g} = 0.0489$ resulting in a substantially larger error of 12.8%.

After the PCH-EM algorithm was complete, we took the resulting membership probabilities and used them to estimate the hidden variable K corresponding to each x_n via

$$\tilde{k}_n = \arg\max_k \gamma_{nk}^{(t)}. \tag{37}$$

These estimates are plotted as a histogram in Figure 5 along with the exact values of k_n, which were hidden from the algorithm. As we observe from the figure, the PCH-EM algorithm was able to demarginalize the hidden variable K as indicated by the fact that the histogram of $\tilde{k}_n$ is in close agreement with the histogram of the exact values. For this dataset, 93.7% of the estimates $\tilde{k}_n$ agreed with the exact values k_n.

To evaluate the algorithm's behavior for the chosen parameters, we repeated this experiment 10, 000 times at $N = 1000$ and again at $N = 5000$, observing the distribution of starting and final estimates. Tables 3-4 compare the exact parameter

TABLE 4. Comparison of starting values and final parameter estimates to exact value for $N = 5000$.

Quantity	Exact	Start	Final
H (e-)	6.87	6.8622 ± 0.1545	6.8700 ± 0.0461
g (e-/DN)	0.0433	0.0433 ± 0.0001	0.0433 ± 0.0001
μ (DN)	42	42.0050 ± 0.7301	42.0073 ± 0.6998
σ^2 (DN^2)	36.083	36.0539 ± 0.9936	36.0687 ± 0.9389

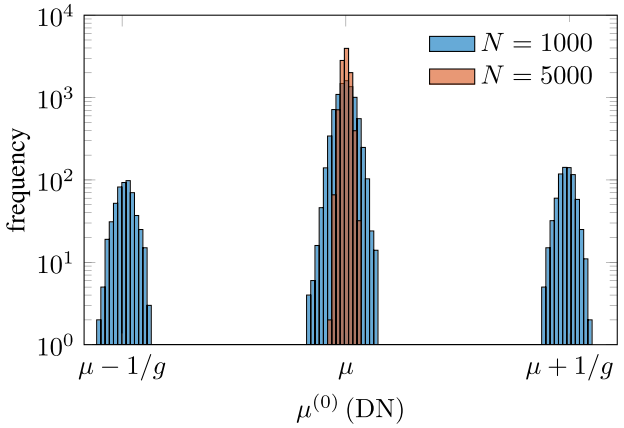

FIGURE 6. Histograms of $\mu^{(0)}$ for 10, 000 Monte Carlo experiments using $N = 1000$ and $N = 5000$.

values to the estimated starting point and final parameter estimates ($\pm$ one standard deviation) for $N = 1000$ and $N = 5000$, respectively. From Table 3, we see all parameters are approximately unbiased; however, we note a couple of anomalies, in particular, high variance in the starting values and final estimates of μ and increased variance in the final estimates of H compared to the starting values. These anomalies disappear in the corresponding data for $N = 5000$.

The source of the anomalies for the $N = 1000$ data comes from the starting values for μ. Figure 6 plots histograms of the 10, 000 starting values $\mu^{(0)}$ for the $N = 1000$ and $N = 5000$ runs. We can see that at $N = 1000$ the starting values are trimodal: one mode centered about the correct value of μ and the other two modes centered approximately around $\mu \pm 1/g$. We note the use of a logarithmic scale to enhance the appearance of these additional modes. When the starting values for μ were in one of these secondary modes the final estimates for H would also be negatively affected. However, we note that the estimates for g and σ^2 are quite robust and yield satisfactory results even when a starting point for μ ends up in one of these incorrect modes. These problems are much less likely when increasing the sample size to $N = 5000$. As one additional exercise, we also repeated the experiment another 10, 000 times for $N = 1000$ but this time we replaced the starting value for μ by the sample mean of $M = 1000$ dark observations $Y \sim \text{PCD}(H_d, g, \mu, \sigma^2)$. When we did this the multi-modal behavior completely disappeared from both the starting values and final estimates. This suggests that when working with small N, the algorithm is dramatically improved by estimating the starting value for μ from an independent dark sample.

VI. CONCLUSION

In this work we have developed the PCH-EM algorithm for estimating key performance parameters of DSERN pixels. A model for DSERN sensor data was derived in the form of the PCD, which was in turn was used to derive the PCH-EM algorithm from the principle of expectation maximization. A method for estimating the starting point for the PCH-EM algorithm was also discussed. These algorithms were implemented in MATLAB and Monte Carlo experiments validated the effectiveness of the algorithms in estimating key performance parameters of DSERN pixels. The specific parameters selected to demonstrate the methods came from [6], however we encourage the interested reader to download the provided source code and adjust the parameters to their specific device [1]. A useful feature of using the provided Monte Carlo method is the ability to conduct sensitivity analysis on potential experimental parameters, e.g., sample sizes.

Future work on the PCH-EM algorithm can be further extended to include two-samples: one captured under illumination and another under dark conditions. Doing so would allow us to separate out the effects of photon interactions and dark current by estimating H_γ and H_d separately, which may be of interest to the community. We also note from our simulation results that the introduction of a dark sample stabilized initial estimates of μ so we might expect a two-sample version of PCH-EM to be more stable compared to the current one-sample method when working with small sample sizes. Additionally, it is important to investigate the performance of the PCH-EM algorithm at higher read noise levels where peaks in the PCD are no longer discernible, identifying areas where different methods are valid.

APPENDIX A
PARAMETER ESTIMATION WITHOUT HIDDEN VARIABLES

Suppose K is not hidden so that we could directly observe the complete data $(\mathbf{x}, \mathbf{k}) = ((x_1, k_1), \ldots, (x_N, k_N))$. With this sample, a maximum likelihood estimator for the parameter θ is very easy to obtain. The likelihood function takes the form

$$L(\theta|\mathbf{x}, \mathbf{k}) = \prod_{n=1}^{N} \prod_{k=0}^{\infty} \left(\frac{e^{-H} H^k}{k!} \phi\left(x_n; \mu + k/g, \sigma^2\right) \right)^{\mathbb{1}_{k_n=k}}, \tag{38}$$

which upon taking the logarithm yields the corresponding log-likelihood function

$$\ell(\theta|\mathbf{x}, \mathbf{k}) = \sum_{n=1}^{N} \sum_{k=0}^{\infty} \mathbb{1}_{k_n=k} \log\left(\frac{e^{-H} H^k}{k!} \phi\left(x_n; \mu + k/g, \sigma^2\right) \right). \tag{39}$$

Here, $\mathbb{1}_A$ denotes the indicator function which is equal to one when A is true and zero otherwise. By the definition of

the indicator function, the log-likelihood then simplifies to

$$\ell(\theta|\mathbf{x},\mathbf{k}) = \sum_{n=1}^{N} \log\left(\frac{e^{-H}H^{k_n}}{k_n!}\phi\left(x_n;\,\mu + k_n/g,\,\sigma^2\right)\right). \quad (40)$$

The maximum likelihood estimate of θ then comes from solving for the critical point $\nabla_\theta \ell = 0$. Equating the appropriate derivatives to zero and solving the resulting system of equations we obtain

$$\tilde{H} = A \quad (41a)$$

$$\tilde{g} = \frac{B - \tilde{H}^2}{C - \bar{x}\tilde{H}} \quad (41b)$$

$$\tilde{\mu} = \bar{x} - \frac{\tilde{H}}{\tilde{g}} \quad (41c)$$

$$\tilde{\sigma}^2 = \hat{x} - \frac{B - \tilde{H}^2}{\tilde{g}^2}, \quad (41d)$$

where $\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$, $\hat{x} = \frac{1}{N}\sum_{n=1}^{N}(x_n - \bar{x})^2$,

$$A = \frac{1}{N}\sum_{n=1}^{N} k_n \quad (42a)$$

$$B = \frac{1}{N}\sum_{n=1}^{N} k_n^2 \quad (42b)$$

$$C = \frac{1}{N}\sum_{n=1}^{N} x_n k_n. \quad (42c)$$

So in the case where K is not hidden (it can be directly observed), closed-form maximum likelihood estimators for the PCD parameter are tractable.

APPENDIX B
DERIVATION OF PCH-EM UPDATE EQUATIONS

We begin with the expression for $Q(\theta|\theta^{(t)})$ in (15) to write

$$Q\left(\theta|\theta^{(t)}\right) = \sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}\left(-H + k\log H - \frac{1}{2}\log\sigma^2 - \frac{(x_n - \mu - k/g)^2}{2\sigma^2} + C\right), \quad (43)$$

where C is a constant independent of θ. The update equations are then derived by solving the system of equations $\nabla_\theta Q = 0$. Taking the derivative of Q w.r.t. H, equating with zero, and simplifying yields

$$\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}(H - k) = 0. \quad (44)$$

Because the $\gamma_{nk}^{(t)}$ represent probabilities w.r.t. the index k we have $\sum_{k=0}^{\infty}\gamma_{nk}^{(t)} = 1$ so that $\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)} = N$. Recalling the definition of $A^{(t)}$ then leads to the solution

$$H^{(t+1)} = A^{(t)}. \quad (45)$$

Next we evaluate $\partial Q/\partial\mu = 0$, which after some simplification gives

$$\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}(x_n - \mu - k/g) = 0. \quad (46)$$

Expanding and simplifying we obtain an expression for μ in terms of g, namely,

$$\mu^{(t+1)} = \bar{x} - \frac{H^{(t+1)}}{g^{(t+1)}}. \quad (47)$$

To find the update equation for g we repeat the process by evaluating and simplifying $\partial Q/\partial g = 0$ to find

$$\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}(x_n - \mu - k/g)k = 0. \quad (48)$$

Substituting $\mu = \bar{x} - H^{(t+1)}/g$ then gives us an equation with one unknown (unknown in g). Using the definitions of $A^{(t)}$ (which equals $H^{(t+1)}$), $B^{(t)}$, and $C^{(t)}$ we are able to write

$$C^{(t)} - \bar{x}H^{(t+1)} + \frac{1}{g}H^{2(t+1)} - \frac{1}{g}B^{(t)} = 0. \quad (49)$$

This equation is then easily solved for g yielding the update equation $g^{(t+1)}$ and subsequently $\mu^{(t+1)}$.

Lastly we evaluate $\partial Q/\partial\sigma^2 = 0$ and simplify to obtain

$$\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}\left(\sigma^2 - (x_n - \mu - k/g)^2\right) = 0. \quad (50)$$

Solving this equation for σ^2 gives

$$\sigma^{2(t+1)} = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}\left(x_n - \mu^{(t+1)} - k/g^{(t+1)}\right)^2. \quad (51)$$

Expanding the trinomial term and simplifying we find after much algebra

$$\sigma^{2(t+1)} = \frac{B^{(t)}}{g^{2(t+1)}} - 2\frac{C^{(t)}}{g^{(t+1)}} + \overline{x^2} + \mu^{2(t+1)} - 2\mu^{(t+1)}\frac{1}{N}\sum_{n=1}^{N}\sum_{k=0}^{\infty}\gamma_{nk}^{(t)}\left(x_n - k/g^{(t+1)}\right). \quad (52)$$

The remaining double sum is equal to $\mu^{(t+1)}$ (c.f. (46)). Replacing $\mu^{(t+1)}$ with the r.h.s. of (47) and performing some algebraic manipulations then yields the final result.

ACKNOWLEDGMENT

The authors would like thank Nico Schlömer for his `matlab2tikz` function, which was used to create the figures throughout this work [15].

REFERENCES

[1] A. Hendrickson and D. P. Haefner. "One-sample PCH-EM algorithm." MATLAB Central File Exchange. 2022. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/121343-one-sample-pch-em-algorithm
[2] E. R. Fossum, "Modeling the performance of single-bit and multi-bit quanta image sensors," *IEEE J. Electron Devices Soc.*, vol. 1, no. 9, pp. 166–174, Sep. 2013.
[3] J. Ma and E. R. Fossum, "Quanta image sensor jot with sub 0.3e- r.m.s. read noise and photon counting capability," *IEEE Electron Device Lett.*, vol. 36, no. 9, pp. 926–928, Sep. 2015.
[4] J. Ma, D. Starkey, A. Rao, K. Odame, and E. R. Fossum, "Characterization of quanta image sensor pump-gate jots with deep sub-electron read noise," *IEEE J. Electron Devices Soc.*, vol. 3, no. 6, pp. 472–480, Nov. 2015.
[5] E. R. Fossum, J. Ma, S. Masoodian, L. Anzagira, and R. Zizza, "The quanta image sensor: Every photon counts," *Sensors*, vol. 16, no. 8, p. 1260, 2016.
[6] D. A. Starkey and E. R. Fossum, "Determining conversion gain and read noise using a photon-counting histogram method for deep sub-electron read noise image sensors," *IEEE J. Electron Devices Soc.*, vol. 4, no. 3, pp. 129–135, May 2016.
[7] K. Nakamoto and H. Hotaka, "Efficient and accurate conversion-gain estimation of a photon-counting image sensor based on the maximum likelihood estimation," *Opt. Exp.*, vol. 30, no. 21, pp. 37493–37506, Oct. 2022.
[8] B. P. Beecken and E. R. Fossum, "Determination of the conversion gain and the accuracy of its measurement for detector elements and arrays," *Appl. Opt.*, vol. 35, no. 19, pp. 3471–3477, Jul. 1996.
[9] J. R. Janesick, *Photon Transfer: DN* $\rightarrow \lambda$. Bellingham, WA, USA: SPIE Press, 2007.
[10] A. Hendrickson, D. P. Haefner, and B. L. Preece, "On the optimal measurement of conversion gain in the presence of dark noise," *J. Opt. Soc. Amer. A*, vol. 39, no. 12, pp. 2169–2185, Dec. 2022.
[11] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc. B, Methodol.*, vol. 39, no. 1, pp. 1–22, 1977.
[12] C. Robert and G. Casella, *Introducing Monte Carlo Methods With R*, 1st ed. New York, NY, USA: Springer, 2010.
[13] G. Casella and R. Berger, *Statistical Inference* (Duxbury Advanced Series in Statistics and Decision Sciences), 2nd ed. Chicago, IL, USA: Thomson Learn., 2002.
[14] A. Hendrickson and D. P. Haefner, "A comparative study of methods to estimate conversion gain in sub-electron and multi-electron read noise regimes," in *Proc. SPIE Infrared Imag. Syst. Design, Anal., Model., Test.*, 2023, pp. 1–14.
[15] N. Schlömer. "Matlab2tikz: A script to convert MATLAB/Octave into TikZ figures for easy and consistent inclusion into LaTeX." May 8, 2021. [Online]. Available: https://github.com/matlab2tikz/matlab2tikz

AARON J. HENDRICKSON received the B.S. degree in imaging and photographic technology from the Rochester Institute of Technology, Rochester, NY, USA, in 2011, and the M.S. degree in applied and computational mathematics from Johns Hopkins University, Baltimore, MD, USA, in 2020. He is currently a Research Mathematician working with the U.S. Department of Defense, DAiTA Group, NAWCAD. His research focus is on developing theoretical foundations for image sensor characterization methods.

DAVID P. HAEFNER received the B.S. degree in physics from East Tennessee State University in 2004, the Ph.D. degree in optics from CREOL, University of Central Florida in 2010, and the first M.S. degree in electrical engineering and the second M.S. degree in mechanical engineering from the Catholic University of America in 2014 and 2015, respectively. Since 2010, he has been working with the U.S. Army C5ISR Center. His current research spans electro-optic imaging system measurement for performance predictions and new measurement development.