

Event-based sensor model for space domain awareness

Rachel Oliver

Cornell University, 124 Hoy Rd, Ithaca, NY 14850

Dmitry Savransky

Cornell University, 124 Hoy Rd, Ithaca, NY 14850

ABSTRACT

We developed a physics-based end-to-end model for event-based sensing of resident space objects (RSOs). Such a model is a key component to advise design decisions for dedicated Space Domain Awareness (SDA) event-based sensors. Additionally, it produces consistent simulated output for algorithm development. The event-based sensors' pixels' asynchronous operations provide higher temporal resolution, improved dynamic range, and discretized output over traditional charged-coupled device (CCD) frames. These qualities appeal to the Space Domain Awareness community as they present opportunities to improve optical observational periods, enable optical analysis of low earth orbit (LEO) satellites, and reduce data generation to levels feasible for space-based SDA networks. With these lofty goals in mind, we demonstrate a physics-based end-to-end model of an event-based sensor. This proof of concept tool includes satellite dynamics, atmospheric turbulence, noise and current leaks in the photoreceptor and comparator circuits, the refractory period of the reset circuit, and the recording process of the arbiter. The initial demonstration of event output is reassembled into a traditional focal plane image over the simulation time frame. The satellite's track is distinguishable from the background shot noise. Next steps will include model validation with truth event-based data and improvement of computational efficiency.

1. INTRODUCTION

Event-based sensors have been of interest to the SDA community in recent years with proof-of-concept demonstrations and characterization of commercial-off-the-shelf (COTS) hardware to assess their capability for low light sensing [6, 15, 23]. What makes these sensors unique is their asynchronous pixels that only record data when an individual pixel's photocurrent passes a defined threshold ratio of the current photocurrent with a previously stored current value. The sensors are biologically inspired and are often compared to a hardware implementation of a neuron that produces a spiked network output. Since the sensors detect logarithmic changes in intensity, they produce minimal data when observing a static scene and have a high dynamic range. These qualities produce high temporal sensitivity for changes that occur within the scene. Subsequently, the sensor also benefits from lower energy consumption and latency [11]. These qualities particularly interest the SDA community by improving optical analyses of low earth orbit (LEO) satellites and minimizing data from extended observation periods to changes in intensity. The sensors also improve the feasibility of small satellite SDA constellations constrained by minimal power and link budgets. Their benefits in size, weight, power, and reduced data generation lend promise to overcoming these constraints.

Implementation of event-based sensors into the SDA architecture still requires research and development. First, the current COTS hardware is not designed for low light SDA sensing. The limiting magnitude of the current hardware given a low-background-noise bias is at best $9.8m_v$ in a staring configuration, with the next generation predicted to be at $10.9m_v$ [15]. During telescope slewing motion, the additional events from the background variation degrade the limiting magnitude performance [15, 23]. Sensitivity studies have shown that event-based cameras are limited in operation to between 1 and 1.8 visual magnitudes brighter than a CMOS camera when viewing the same scene. The second reason event-based sensing is not ready for implementation is that current use of the disaggregated output from the sensors does not take advantage of data minimization. Analyses presented in [6, 15, 23] describe results in the traditional frame versus time

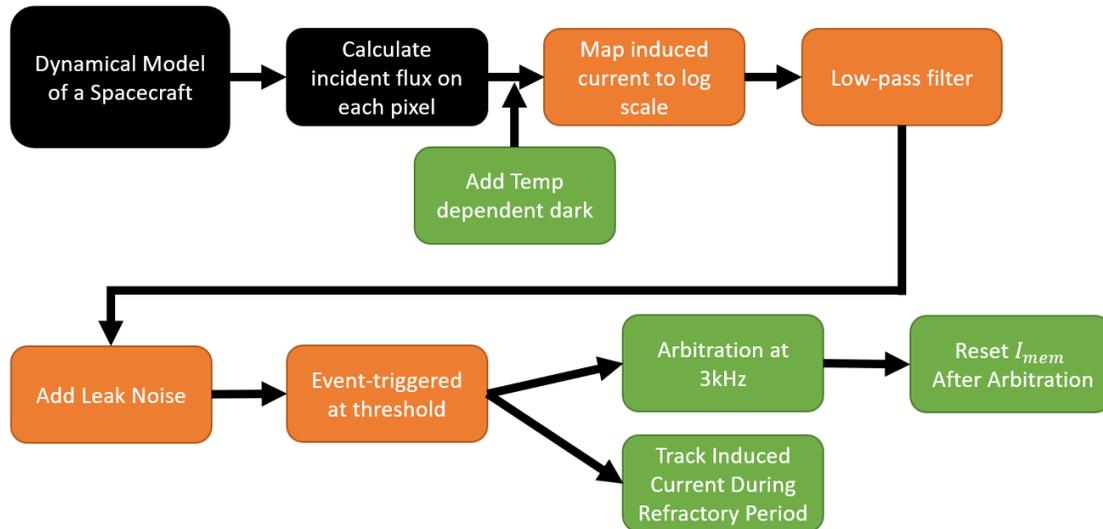


Fig. 1: The model’s existing code, such as the satellite dynamics propagation, is colored in black on the flow chart. The orange indicates inclusion in [8]’s software that takes traditional video frames and estimates a dynamic vision center output [8]. The green are additions added to the processing flow to capture noise that is particularly dominant in low light imaging. These account for the possible impacts of the event-based sensor’s arbiter.

format. While this is convenient for comparison against conventional imaging methods, analysis techniques of the time series data need attention. Improvement of pattern recognition and noise rejection with machine learning techniques presents an avenue towards autonomous on board computing that could utilize the collected data prior to down-linking.

An end to end, physics-based model will help with both of the aforementioned issues. It enables efficient hardware exploration to inform investigations to improve low light sensing such as choosing a different detector material or incorporating active cooling. Theoretical improvements predicted by the simulation can be weighed against cost before investment. A physics-based model will also provide consistent simulated data sets for algorithm development. There are previous modeling efforts described in literature. One effort conducted by [8] converts standard video footage into a simulated event-based output. Their simulator provides a processing framework for the model described in this paper, but it is not tailored to the SDA analysis. The model presented in this paper on the other hand follows the processing flow in Fig. 1. Starting with inputs of a two line element set, satellite material and physical properties, observer and sensor properties, and atmospheric conditions, the model estimates incident power on the focal plane. Additions to the existing simulation tool’s processing include temperature dependent leaks and dark current to capture low light sensing impacts. Finally, the arbitration process is included to capture delays between events and readout and events missed while an pixel’s event waits for recording. These steps that tailor the model for SDA and component analysis will be described in greater detail in Section 2.

2. METHODOLOGY

2.1 Dynamics

The simulation starts with by determining if a satellite of interest is in the field of view (FOV) of the chosen sensor and telescope combination. Two line element sets (TLEs) are loaded from Celestrak.org through the Skyfield framework [12, 19]. The simplified general perturbations propagator (SGP4) informs the satellite position during the chosen simulation time. TLEs should be as close as possible to the simulation time chosen because of the error accumulation in this propagation method. Historical TLEs are available in the Celestrak archives. Determining if the satellite is in the FOV is equally dependent on the observer’s properties. The

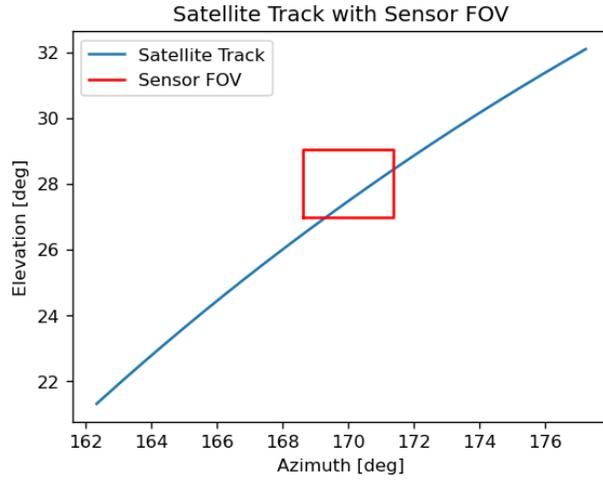


Fig. 2: Output of satellite selection and propagation includes identification of when the RSO is in the FOV, distance from the center of the FOV to assist in the creation of a point source, and a plot to assure the user that their chosen combination of satellite, observer properties, and time will yield an observation.

user inputs a viewing location, pointing of the observer, and either an angular definition of the true FOV or properties to calculate the true FOV. At this time, the simulation only takes inputs for staring sensors. These parameters define azimuth and elevation bounds of the sensor FOV which are then compared to the satellite's azimuth and elevation at each simulation time step. After all of the user defined parameters are chosen, the model outputs the propagation graphically as depicted in Fig. 2. Subsequent calculations only begin once the satellite is in the FOV. Additional condition checks prior to calculations ensure the satellite is not in eclipse.

2.2 Energy Propagation

Once the simulation confirms the satellite is in the field of view of the sensor, the next step is to determine the incident energy on the focal plane area. This is completed through Fourier propagation of the electromagnetic field created by the satellite. Assuming the satellite is a point source, an estimate of the power per subtended angle, intensity, is needed to define the source field [9]. With one additional assumption that radiance is constant over the satellite surface, the power of the electric field at the point source, Φ , is estimated as a function of the projected area with

$$\Phi \approx \pi L A_{send} [W]. \quad (1)$$

L is the radiance and A_{send} is the projected area. The constant radiance is approximated by the function

$$L \approx \frac{\bar{\rho} \varepsilon_{external}}{\pi} \left[\frac{W}{m^2 sr} \right] \quad (2)$$

of the radiant energy, $\varepsilon_{external}$ from the sun and the reflectivity, ρ , of the surface of the satellite.

After the power directed towards the hemisphere of the observer is estimated, a source plane is constructed by assuming the point source is Gaussian and ensuring it is shifted off axis of the center of the focal plane. Therefore, the point source field, U is defined by

$$U_{pt}(r_1) = \Phi e^{-i \frac{k}{2\Delta z} r_1^2} e^{i \frac{k}{2\Delta z} r_c^2} e^{-i \frac{k}{\Delta z} r_c \cdot r_1} \times \left(\frac{D}{\lambda \Delta z} \right)^2 \text{sinc} \left[\frac{D(x_1 - x_c)}{\lambda \Delta z} \right] \text{sinc} \left[\frac{D(y_1 - y_c)}{\lambda \Delta z} \right]. \quad (3)$$

Φ , the power, is the scaling factor the field. The phase is captured in polar coordinates, so r_1 and r_c are array of radii from the center and satellite grid points respectively. The radii are magnitudes without direction, so the Cartesian mappings, x_1 , y_1 , x_c , and y_c , and the sinc functions capture the point source dimensions on the source plane and the spatial shift of the point source. The other constants in Equation 3 include

the wavelength, λ , wave-number, k , propagation distance, Δz , and the aperture diameter of the observation plane, D .

To propagate the energy on the satellite source plane to the optical frame, a multi-layer angular spectrum propagation is utilized. The multiple layers allow use of Fresnel diffraction and the inclusion of a multi-layered phase-screen to model atmospheric turbulence. The split-step beam propagation method with “n” propagation steps has “n-1” phase screens incorporated into the propagation by a Fourier product with the formula

$$\begin{aligned}
 U(r_n) = & \mathcal{Q} \left[\frac{m_{n-1} - 1}{m_{n-1} \Delta z_{n-1}}, r_n \right] \\
 & \times \prod_{i=1}^{n-1} \left\{ \mathcal{T}[z_i, z_{i+1}] \mathcal{F}^{-1} \left[f_i, \frac{r_{i+1}}{m_i} \right] \mathcal{Q}_2 \left[-\frac{\Delta z_i}{m_i}, f_i \right] \mathcal{F} \left[r_i, f_i \right] \frac{1}{m_i} \right\} \\
 & \times \left\{ \mathcal{Q} \left[\frac{1 - m_1}{\Delta z_1} \right] \mathcal{T}[z_1, z_2] U(r_1) \right\}.
 \end{aligned} \tag{4}$$

The observation plane field is found through this equation by multiplying the source field by a complex phase screen, \mathcal{T} , and a quadratic phase factor, \mathcal{Q} . For each layer of propagation the Fourier transform is taken, another quadratic phase factor is applied to focus the spherical wave onto the current plane, the inverse Fourier transform is taken, and finally the phase modification from the turbulence is multiplied element wise onto the output. The final step is to collimate the field, removing the spherical-wave phase by imposing a final quadratic factor [22, 5]. Fig. 3 shows an example of mapping of an on-axis point source to the intensity on a focal plane.

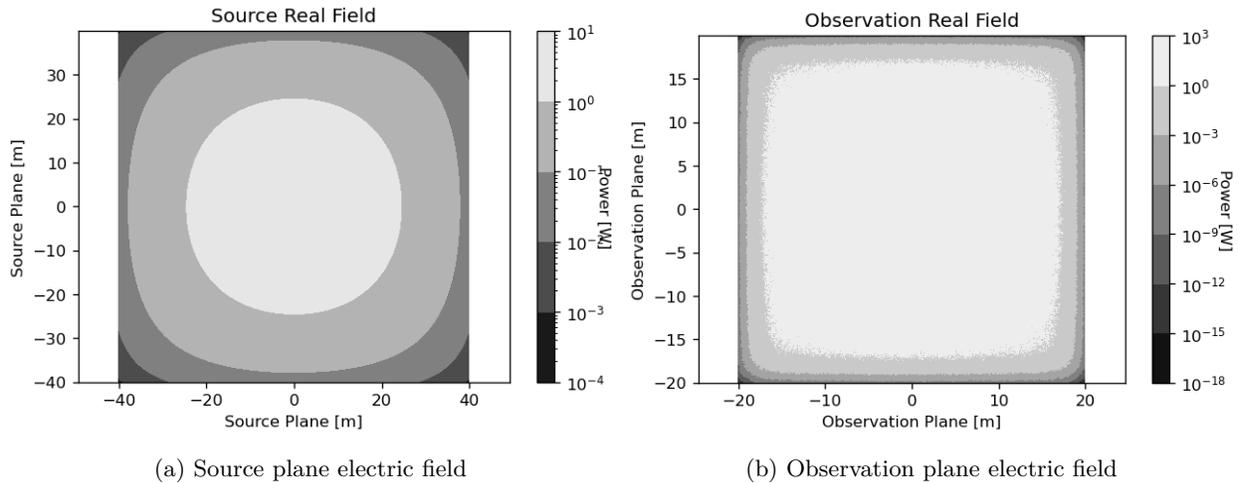


Fig. 3: Plotted power in Watts from the source plane through a turbulent phase screen propagation. Sampling dimensions for the source and observation plane must follow bounds to prevent Fourier aliasing. The dimensions provided in these plots are not representative of the electric field in the FOV of a staring sensor. Instead, these less extreme dimensions are chosen for demonstration of the phase screen impact.

Creating a relevant set of phase screens is done using Kolmogorov’s statistical theory of turbulence. Randomly sampled coefficients of a set of Zernike modes describe the Karhunen-Loéve functions [20, 5]. The definition of the modes, as outlined in [16], organizes the orthogonal modes by radial degree, n , and azimuthal frequency, m . Those with higher frequencies have even and odd modes defined with cosine and sine functions respectively as

$$\begin{aligned}
 Z_{evenj} &= \sqrt{n+1} R_n^m(r) \sqrt{2} \cos(m\theta) \\
 Z_{oddj} &= \sqrt{n+1} R_n^m(r) \sqrt{2} \sin(m\theta) \\
 Z_j &= \sqrt{n+1} R_n^0(r), m = 0
 \end{aligned} \tag{5}$$

where R_n^m is determined by

$$R_n^m(r) = \sum_{s=0}^{(n-m)/2} \frac{(-1)^s (n-s)!}{s! [(n+m)/2 - s]! [(n-m)/2 - s]!} r^{n-2s}. \quad (6)$$

These modes are functions of the polar coordinates, radius, r , and phase, θ , of the propagation plane and, therefore, must be computed for each cell for each phase screen generated unless the spacing is constant [20].

To create a phase screen describing the statistics with these modes, they are combined with the independent Karhunen-Loève coefficients through a matrix summation

$$\Theta_{atm}(r, \theta) = \sum_j a_j Z_j(r, \theta). \quad (7)$$

The combination with the Zernike modes comprises a summation over each mode once the coefficients, a_j , are statistically determined [20]. The coefficients are calculated by computing the covariance of pairs of Zernike polynomials, Z_j and $Z_{j'}$, excluding the first mode through

$$C_{j,j'} = E[a_j, a_{j'}] = \frac{K_{zz'} \delta_z \Gamma[(n+n'-5/3)/2] (D/r_o)^{5/3}}{\Gamma[(n-n'+17/3)/2] \Gamma[(n'-n+17/3)/2] \Gamma[(n+n'+23/3)/2]}. \quad (8)$$

The $K_{zz'}$ coefficient is defined by

$$K_{zz'} = \frac{\Gamma(14/3) [(24/5) \Gamma(6/5)]^{5/6} [\Gamma(11/6)]^2}{2\pi^2} \times (-1)^{(n+n'-2m)/2} \sqrt{(n+1)(n'+1)} \quad (9)$$

and the Kronecker delta is

$$\delta_z = (m = m') \wedge (\overline{\text{parity}(j, j')} \vee (m = 0)). \quad (10)$$

The values of n and m maintain their values from the initial Zernike mode definitions, while D and r_o are the aperture of the receiving plane and Fried's parameter respectively. Fried's parameter ties the statistics of the Karhunen-Loève coefficients to the severity of the atmospheric turbulence [10].

After defining the covariance matrix and reordering the polynomials to ensure a block diagonal construction, the covariance matrix is now Hermitian and, therefore, there exists a unitary matrix, U , such that $U \cdot C \cdot U^T$ is diagonal. The Karhunen-Loève coefficients are defined as a product of the unitary matrix, gained through a decomposition of the covariance matrix, with a column vector of Gaussian random variables, \vec{n} with zero mean in

$$a_j = U^T \cdot \vec{n}. \quad (11)$$

The variance of the vector is given by a diagonal matrix output of the decomposition for models with more than one mode requiring a single value decomposition [20]. 5 Zernike modes are depicted without their orthonormal complement along with a resulting phase screen in Fig. 4. Higher order phase screens provide more complex variance in phase that captures beam intensity modification, but the predominant modes that capture distortion on the focal plane are tip and tilt, modes 2 and 3 respectively [3]. Therefore, a low order model will suffice for the estimation done in this simulation and save computational time.

With the assumption of frozen flow, as described in [18], the computationally expensive phase screens can be reused for multiple times during the simulation. This is because the time scale of the satellite passing through the image frame is short relative to the time scale of the variation in atmosphere turbulence. Individual layers of atmospheric phase modulation remain internally stagnant, but translate relative to an observer. A shorter revisit time can be set in this simulation to force regeneration phase screens. Ideally, if the dynamics of the drifting turbulence is known, then the phase screen can be used for longer periods of time. This is done by translating the screen with regard to the observer's coordinate system. The one caveat is that the phase screen is finite. Therefore, while translating a phase screen over time, careful consideration must be taken to determine phase screen coverage of the propagation. This model does not currently take this possibility into account, but it may be considered in the future to increase the simulation accuracy.

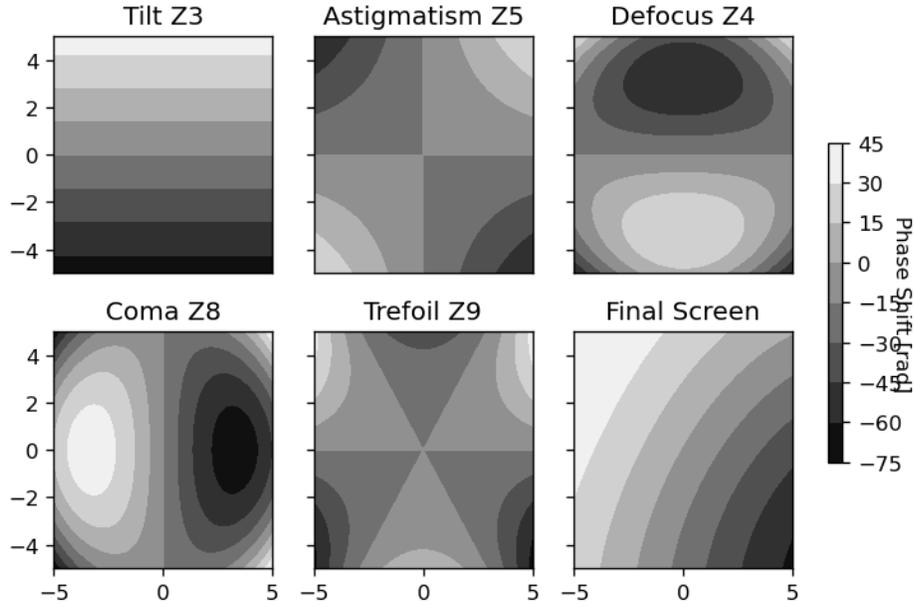


Fig. 4: From the upper left to bottom right the Zernike phase distributions of increasing modes are shown. The Noll number associated with the mode is listed in each graph's title. Complementary modes with the opposite sign are not plotted. The resulting phase screen, on the lower right, is a combination of these modes and their complements scaled by the Karhunen-Loève coefficients that capture the statistics of atmospheric turbulence.

2.3 Photoreceptor

The photocurrent derived from the incident energy on the focal array is modeled as a traditional image sensor, then converted to a logarithmic scale to better capture the response of the silicon retina. The conversion of incident energy and electrical output is simply modeled by assuming a linear relationship through the parameter based term responsivity, R_λ , as defined by

$$R_\lambda = \eta \frac{q}{hf} \approx \eta \frac{\lambda}{1.23985} \left[\frac{A}{W} \right]. \quad (12)$$

The approximate responsivity value for the observing wavelength is related to the detector's quantum efficiency, η , and observing wavelength λ . Alternatively, it is calculated with the electron charge, q , the signal's frequency, f , and Plank's constant, h [4]. The induced photocurrent is then modeled as directly proportional to the received power by

$$I_p = R_\lambda \Phi. \quad (13)$$

The logarithmic response, key to the high dynamic range of the sensors, is completed with the piecewise function

$$I_{plog} = \begin{cases} I_p \frac{\ln(5)}{5} & I_p < \ln(5) \\ \ln(I_p) & I_p \geq \ln(5). \end{cases} \quad (14)$$

The piecewise mapping is linear at low levels of induced current because the log function becomes more sensitive at values closer to 0. In the simulation, not compensating for the sensitivity would trigger a large number of unphysical negative events when the true sensor's photocurrent gradually decays to zero as depicted in Fig. 5a [8].

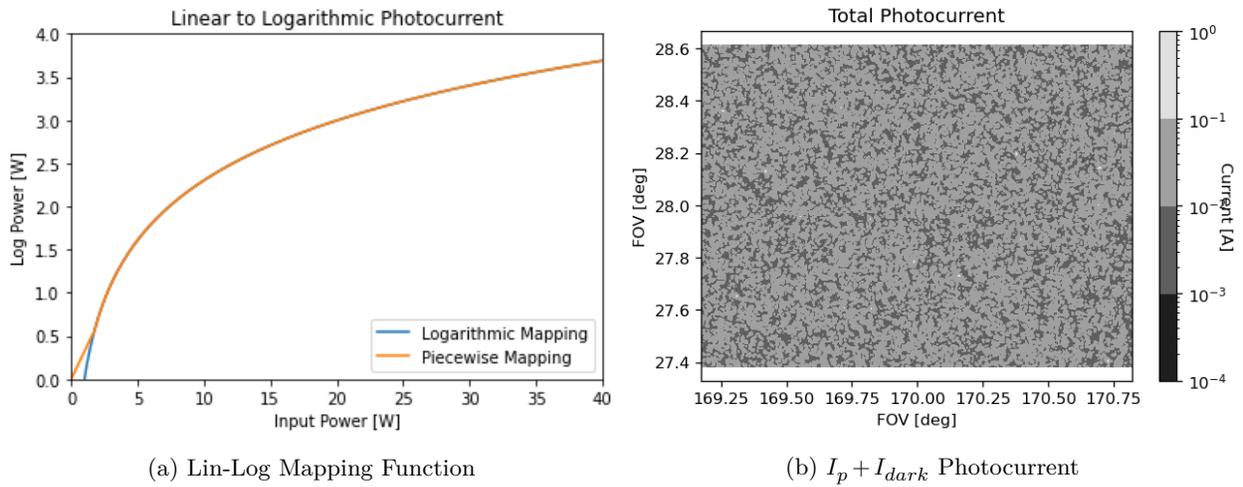


Fig. 5: The photoreceptor model takes into account the dynamic range of the event-based sensors and incorporates dark photocurrent. a) The piecewise function that linearizes the mapping at low intensities prevents triggering erroneous off events that would be nonphysical. b) The time slice of total photocurrent is a subsection of the observation field in 3. The current has undergone the logarithmic mapping and the dark current is incorporated. Note, at this scale it is easier to see the affects of the phase distortions during the propagation phase of the model.

The next consideration in the photoreceptor section of the circuit is the parasitic dark current. Dark current degrades the signal to background contrast which increases the event threshold and increases the shot noise during low light sensing [17]. Since these effects' impacts are prevalent in low light sensing scenarios, it is logical to include this phenomenon in an SDA model when it may normally be neglected. In other models the dark current may be modeled as a constant. However, since its effect is proportional to the diode's temperature and active cooling is a common method to improve the sensitivity of the detector, a more complex dark current determination is included.

$$I_{darklog} = \ln(I_{dark}) = constant - \frac{E_a A_{pix}}{kT^4} \quad (15)$$

provides the linear relationship to incorporate the reciprocal temperature into the dark current calculation. The constant and activation energy, E_a , is determined experimentally in [17] while k is Boltzmann's constant, A_{pix} is the area of the pixel, and T is the absolute temperature in Kelvin.

After adding the photocurrent and dark currents, a low-pass filter is applied to the input current values because a real event-based pixel has finite analog bandwidth limiting the response in low light conditions [8]. Since low light conditions are ubiquitous with SDA sensing, the low pass filter is necessary for the model. Kirchoff's laws are implemented through the difference method in the following process.

$$\begin{aligned} f_{3dB} &= ((I_{in} + 3)/5.616) \times f_{3dBmax} \\ \epsilon &= \frac{2\pi\Delta t f_{3db}}{2\pi\Delta t f_{3db} + 1} \\ I_p &\leftarrow (1 - \epsilon)I_{p-1} + \epsilon I_{in} \end{aligned} \quad (16)$$

The cutoff frequency, f_{3db} , changes with the strength of the current, so a constant low pass transfer cannot be applied. Instead it is estimated by equating the maximum cutoff frequency to the upper threshold of intensity values before saturation. The digital number computation, suggested by [8], is converted to a ratio of induced photocurrent with the assumption that the maximum cutoff frequency is approximately 3 kHz[13].

2.4 Comparator

The fundamental construct of event-based circuitry is the current comparison to a previously memorized current value. The comparison is easily represented as logarithmic intensity where the logarithmic ratio of the current values is the difference between the those values [8]. The comparison process is condensed into the logical process

$$\begin{aligned}
 \Delta I_{log} &= \ln \left(\frac{I_{new}}{I_{mem}} \right) = I_{1plog} - I_{memlog} \\
 \theta &= \begin{cases} \theta_{ON} & \Delta I_{log} \geq 0 \\ \theta_{OFF} & \Delta I_{log} < 0 \end{cases} \\
 Ne_{total} &= \text{floor} \left(\frac{\Delta I_{log}}{\theta} \right) \\
 Ne &= \begin{cases} 1 & Ne_{total} \geq 1 \\ 0 & |Ne_{total}| < 1 \\ -1 & Ne_{total} \leq -1 \end{cases} \\
 N_{loss} &= Ne_{total} - Ne \\
 I_{memlog} &= I_{memlog} + Ne \times \theta
 \end{aligned} \tag{17}$$

where I_{mem} is the memorized current from the last event that is stored in the differencing amplifier while I_{1p} is the output. The threshold values, θ , are unique to each pixel and different for positive and negative events. A randomly sampled Gaussian distribution centered on a nominal θ value of 0.2 and standard deviation of 0.03 defines the thresholds for a given simulation as derived in [8]. The number of possible events is the change in log intensity divided by the appropriate theta value. It is rounded towards a nominal 0 value because the number of events can only be whole numbers.

Unlike [8]'s simulator, which spreads the total number of events over evenly spaced time steps, only one event can be triggered per time step in this model because real sensors only record one event per pixel at a time. During a refractory period, discussed in further detail in Section 2.5, the circuit ignores changes in log intensity [8]. Therefore, the total number of possible events, Ne_{total} is reduced to a positive or negative unitary corresponding to the polarity, Ne . Finally, the memorized current value used in this process is updated according to the event's polarity. In order to capture the potential data loss, the number of possible events is recorded in the matrix N_{loss} .

There are three additional noise sources to consider that affect the comparison of the photocurrent to the memorized value. First, there is a current leak that passes from the differencing amplifier into the floating input node of the same amplifier which causes the differencing voltage and, therefore, the I_{memlog} to consistently decrease. Over time this effect will generate pseudo "on" events without an actual change in incident energy on the focal plane. Unlike the dark current, this junction leak is not sensitive to temperature because both the amplifier output voltage and photoreceptor transpediance are proportional to absolute temperature. Therefore, this leak can be modeled by

$$\begin{aligned}
 \delta_{leak} &= \frac{\Delta t R_{leak}}{\theta_{ON}} \\
 I_{memlog} &= I_{memlog} - \delta_{leak}
 \end{aligned} \tag{18}$$

as function of the leak rate, f_{leak} , the "on" threshold, θ_{ON} , and the simulation time step where the function leak rate is nominally 0.1 Hz [17].

The second noise source in the comparator circuit is a parasitic photocurrent caused by stray light that either has penetrated through the protective metal or scattered into the n-well. The current is treated as a parallel current to the junction leak, so it results in additional positive events. The noise is treated in the same way as the junction leak in Equation 18, but the computation of the leak rate is a function of the pixel illumination as empirically derived in [17]. Using the incident power provided by the propagation, pixel illumination, M_v is estimated through

$$\begin{aligned}
M_v &= \frac{\Phi\eta}{A} \\
R_{para} &= M_v \frac{f_{para}}{lum}.
\end{aligned} \tag{19}$$

The calculation also requires luminous efficacy, η , and the area of the pixel. Luminous efficacy is assumed to be $680lum/W$ for these calculations. The event rate, R_{para} , in Hz is calculated by a linear relationship found experimentally to link the pixel illumination to an event rate. The relationship is unique to each sensor [17]. Once the event rate is determined, the change to the memorized current is calculated using 18 to give the alternative rate.

The final noise source to consider is shot noise. Due to the integration time of an event-based pixel being roughly proportional to the intensity on that pixel, the integration is over a constant number of photons and has a total noise power constant with intensity. In particular, low light sensing produces higher rates of shot noise because the noise is spread over less bandwidth. Therefore its inclusion in the SDA model is important. The Poisson methodology,

$$\begin{aligned}
r &= ((F - 1) \times (I_p/I_{max}) + 1) \times R_n \\
p &= r \times \delta t \\
u < p &: event_{OFF} \\
u > (1 - p) &: event_{ON},
\end{aligned} \tag{20}$$

developed in [8], scales an observed rate noise, R_n , by a linear function with the slope F of the normalized log photocurrent used to determine r . It is assumed the maximum log photocurrent, I_{max} , is 5.54 corresponding to the maximum digital number, 255, of the silicon array converted to log scale. r is then multiplied by the time step, δt , to find the probability of an event, p . Now a uniformly distributed sample between 0 and 1, u , is compared to the Poisson probability distribution to determine if an event occurred. Generation of an event resets the memorized current just as a legitimate event and follows the θ determination and updates the final line of Equation 17. The [8] software limits the noise injection only to low light intensities because there is sufficient noise in the initial video input for higher intensities. This model however, requires noise injection throughout all intensities, so every pixel undergoes the shot noise determination at each time step even if a legitimate event is not recorded.

2.5 Reset Circuit and Arbitration Circuit

After an event occurs, the recording of the event is not instantaneous. Instead, a transistor is triggered and applies current to a reset circuit for a set period of time, the refractory period. During this time the triggered pixel does not compare its current with the newly memorized value [8]. Depending on the event-based sensor, the arbitration circuit either cycles through each row and acknowledges an event or lack of event in each column or addresses an entire column as a group that had at least one event in a column scan readout [2, 21]. Neither of these phenomenon are included in previous simulators or models despite having a potentially profound influence on the number of events recorded.

Not recording during the refractory period creates a time delay on triggered events. Additionally, if the current changes enough to elicit an event, but returns to baseline within the period, the information about the current during that time frame is lost by under-sampling. The refractory period is a user setting for COTS event-based sensors, but there are computational costs to weigh against the sampling benefits of a shorter refractory period. Capturing the reset circuit's refractory period will assist with choosing an appropriate setting for SDA applications. The model's refractory period process, summarized by

$$\begin{aligned}
T_{hold} &= T_{hold} + N_e \\
T_{holdtime} &= T_{holdtime} + \Delta t \\
T_{hold} &= \begin{cases} 0 & \Delta T_{holdtime} \geq T_{refr} \\ 1 & \Delta T_{holdtime} < 0, \end{cases}
\end{aligned} \tag{21}$$

maintains a matrix, T_{hold} , that tracks the pixels that have triggered events and another matrix, $T_{holdtime}$, that tracks the total time since the refractory period, T_{refr} , began for each triggered pixel. After each simulation step, T_{hold} is updated with the newly triggered events in N_e . Then each element of the $T_{holdtime}$ matrix corresponding to a in the T_{hold} matrix is compared against the defined refractory period time. Once the hold time is larger than the refractory period, the current comparison is allowed once again.

The arbitration process can also induce errors. The arbiter records data in an orderly fashion, but not necessarily in the order that events were actually triggered. Additionally, when saturated the arbiter will fail to record, skipping events, as described in [2]. Modeling the arbiter helps assess the prevalence of improperly ordered events, generates statistics on the magnitude of the error in the recording time, and increases understanding of the system limitations imposed by the arbitration circuit. The current model arbitration circuit follows the row then column sampling method. If an event exists at a queried index, the event data is recorded as a dictionary with the structure

$$e = \{t_a, x, y, \rho, t_t\} \quad (22)$$

where x and y are the row and column indices of the pixel, ρ is the polarity of the event, t_a is the simulation time the arbiter recorded, and t_t is the simulation time when the event was triggered.

3. RESULTS AND CONCLUSION

A sample of the model's output is integrated over the simulation's duration and mapped to the two-dimensional arrays in Fig. 6. The initial results show promise with a visible satellite track comprised of green "on" events through the lower right hand corner. The random scattered events are predominantly "off" events produced by shot noise. Therefore, the scattered noise is significantly reduced by only modifying the leak and shot noise rates in Fig. 6b. These parameters will vary between designs and individual event-based sensors. Parameter estimation will improve matching model performance to that of a true event-based sensor. With a functional proof of concept model, the next steps in the model improvement include validation of the model against RSO data collections and improvement of computational efficiency.

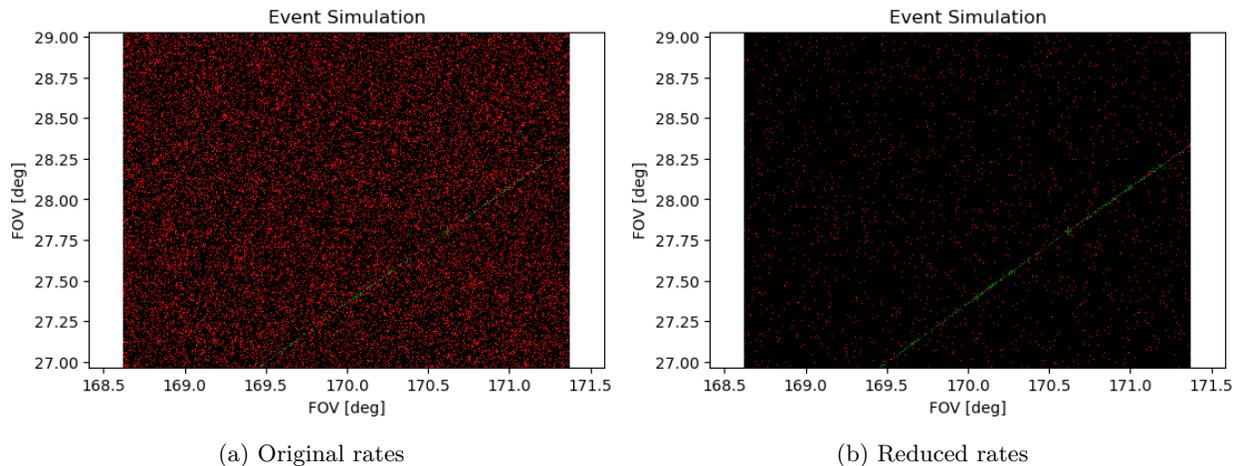


Fig. 6: The event-based output mapped to a 2D grid over a 7.7 second observation period. The green events correspond to positive threshold changes while the red corresponds to negative threshold changes. The satellite is traversing from the right hand side of the FOV to the left. The two simulations differ in the nominal leak and shot noise rates to reduce the noisy negative events. Since the true parameters are not perfectly known, parameter estimation will help narrow the performance to a particular sensor.

Validation of the model, which will likely be unique for each sensor with slightly different sensitivities, is well suited for parameter estimation. The performance of the model should converge towards the true

sensor's performance with a proper combination of settings. Key metrics to access the performance include the ratio of number of events to the expected number of events, the event-loss rate, and background noise rate. The first is a ratio of the simulation events to the collected events. The event-loss rate is a ratio between the recorded events and those lost due to sample recording rate or the refractory period. Finally, the background noise rate will examine pixels outside the satellite track in darker areas of the image to determine if the current leak model is accurately capturing the true leak events.

In addition to model validation, computational efficiency will be a focus of future model improvement. The current end-to-end model design is computationally heavy. The split-beam propagation and Zernike mode phase screens use large grids that require proportionally large allocations of memory. The grid sizes must be on the order of millions of spaces to cover the field of view and capture all the frequency information in the Fourier transforms. One way to reduce the memory load is to increase the time step size. A sensitivity analysis will be conducted to quantify the model confidence with lowering fidelity. Anticipated impacts of increasing the step size include discretization of the propagation's power delivered to the sensor, worsening of the induced current sampling frequency, and reduction of accuracy in the delay quantification associated with the arbiter. This analysis will provide the statistical foundation for making the trade-off between the model accuracy and speed.

4. REFERENCES

- [1] *1st ESA NEO and Debris Detection Conference*, 2019.
- [2] T. Brewer. A comparative evaluation of the fast optical pulse response of event-based cameras. Master's thesis, Air Force Institute of Technology, 2021.
- [3] L. Burger, I. Litvin, and A. Forbes. Simulating atmospheric turbulence using a phase-only spatial light modulator. *South African Journal of Science*, 104:129–134, March 2008.
- [4] K. Busch and M. Busch. *Multielement Detection Systems for Spectrochemical Analysis*. Wiley-Interscience, 1990.
- [5] S. Chesnokov, V. Kandidov, V. Shmalhausen, and V. Shuvalov. Numerical/optical simulation of laser beam propagation through atmospheric turbulence. Technical Report N68171-94-C-9147, United States Army, London, Dec 1995.
- [6] G. Cohen, S. Afshar, A. Schaik, A. Wabnitz, T. Bessell, M. Rutten, and B. Morreale. Event-based sensing for space situational awareness. [14].
- [7] Conference on Computer Vision and Pattern Recognition Workshops. *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021.
- [8] T. Delbruck, Y. Hu, and Z. He. V2e: from video frames to realistic dvs event camera streams. [7].
- [9] H. Evans, J. Lange, and J. Schmitz. *The Phenomenology of Intelligence-focused Remote Sensing*. Riverside Research, 2014.
- [10] D. L. Fried. Statistics of a geometric representation of wavefront distortion. *Journal of the Optical Society of America*, 55(11):1427–1435, November 1965.
- [11] G. Gallego, T. Delbruck, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. Davison, J. Conradt, K. Daniilidis, and D. Scaramuzza. Event-based vision: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
- [12] T. S. Kelso. Celestrak, 2021.
- [13] P. Lichtsteiner, C. Posch, and T. Delbruck. A 128x128 120 db 15 μ s latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):556–576, February 2008.
- [14] Maui Economic Development Board. *18th Advanced Maui Optical and Space Surveillance Technologies*, 2017.
- [15] P. MchMahon-Crabtree and D. Monet. Commercial-off-the-shelf event-based cameras for space surveillance applications. *Applied Optics*, 60(25):G144–G153, September 2021.
- [16] R. Noll. Zernike polynomials and atmospheric turbulence. *Journal of the Optical Society of America*, 66(3):207–211, March 1976.
- [17] Y. Nozaki and T. Delbruck. Temperature and parasitic photocurrent effects in dynamic vision sensors. *IEEE Transactions on Electron Devices*, 64(8):3239–3245, August 2017.
- [18] L. Poyneer, M. Dam, and J. Véan. Experimental verification of the frozen flow atmospheric turbulence

- assumption with use of astronomical adaptive optics telemetry. *Optical Society of America*, 26(4):833–846, April 2009.
- [19] B. Rhodes. Skyfield: High precision research-grade positions for planets and earth satellites generator. <https://github.com/skyfielders/python-skyfield>, 2020.
- [20] N. Roddier. Atmospheric wavefront simulation using zernike polynomials. *Optical Engineering*, 29(10):1174–1180, October 1990.
- [21] H. Ryu. Industrial dvs design; key features and applications. <https://docplayer.net/151636613-Industrial-dvs-design-key-features-and-applications.html>, 2019.
- [22] J. D. Schmidt. *Numerical Simulation of Optical Wave Propagation with examples in MATLAB*. SPIE, 2010.
- [23] M. Zolnowski, R. Reszelewski, D. Moeys, T. Delbruck, and K. Kaminski. Observational evaluation of event cameras performance in optical space surveillance. [1].