An Event-Based Vision Sensor Simulation Framework for Space Domain Awareness Applications

B Exam

Rachel Oliver

Cornell University

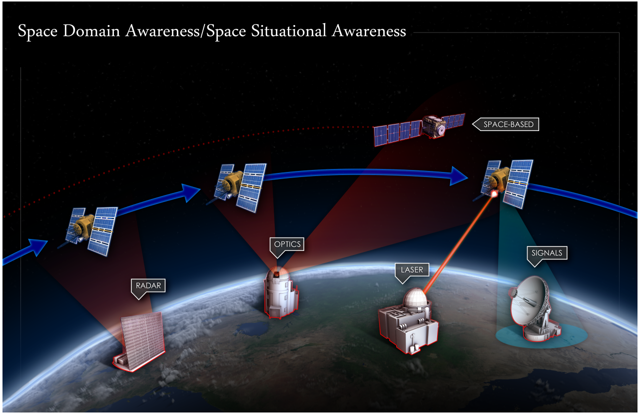
SIOS LABORATORY

25 July 2024

Outline

Outline

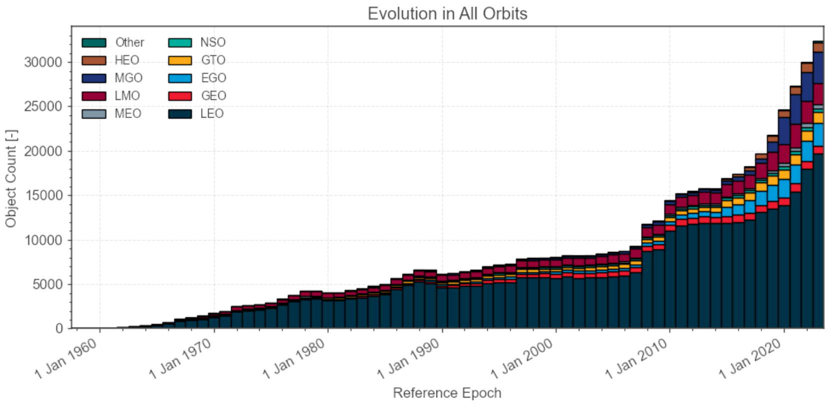
Space Domain Awareness (SDA)



Weisbarth, 2020, `https://media.defense.gov/2020/Mar/15/2002264814/-1/-1/0/190111-F-HF064-002.PNG`

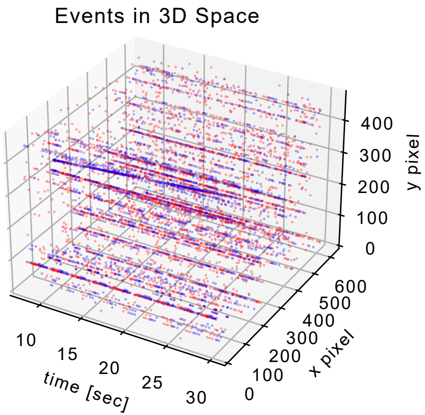
MDA, 2008, `http://www.mda.mil/mdalink/pdf/too164.pdf`

Growing Complexity



ESA's Space Environment Report Office 2023

Event-Based Vision Sensors (EVS)



- Asynchronous data collection
- Records time series events, e
  - $e = \{t, x, y, \rho\}$
  - recording timestamp, t
  - pixel location, x and y
  - change polarity, ρ
- Blue = ON events (positive events)
- Red = OFF events (negative events)
- Other Names = Neuromorphic Camera, Neuromorphic Sensor, Silicon Retina, Event-Based Camera, Event-Based Sensor, or Dynamic Vision Sensor

EVS Advantages in SDA

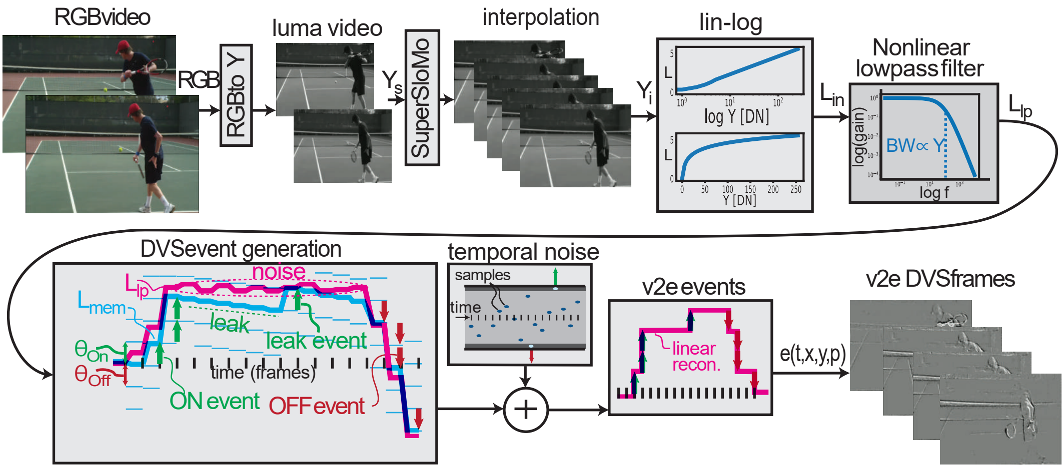
- Temporal Sensitivity
- High Dynamic Range
- Space-Based
    - Low SWaP for Space-Based
    - Maximize Information for Downlink
    - Enable Onboard Computation
- Ground-Based
    - Low Cost for Augmenting SDA Operations

http://greg-cohen.com/project/astrosite/ *The Astrosite: A Mobile Neuromorphic Space Domain Awareness Observatory* 2019

Implementation of EVS in SDA

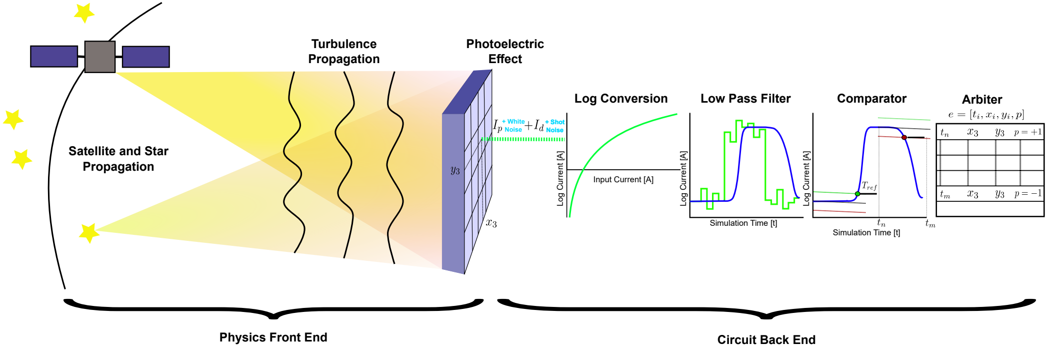
Challenges to Implement EVS for SDA	My Contributions
Existing data has limited edge cases for algorithm development	Photon-Level Event Generation Simulation
Difficult to collect space-based data	
Current synthetic event generators take shortcuts	
Leveraging data sparsity requires frame-less processing techniques	Event Data Point Source Tracking Algorithm
Point sources generate events on single pixels which resemble noise	

State of the Art EVS Simulation

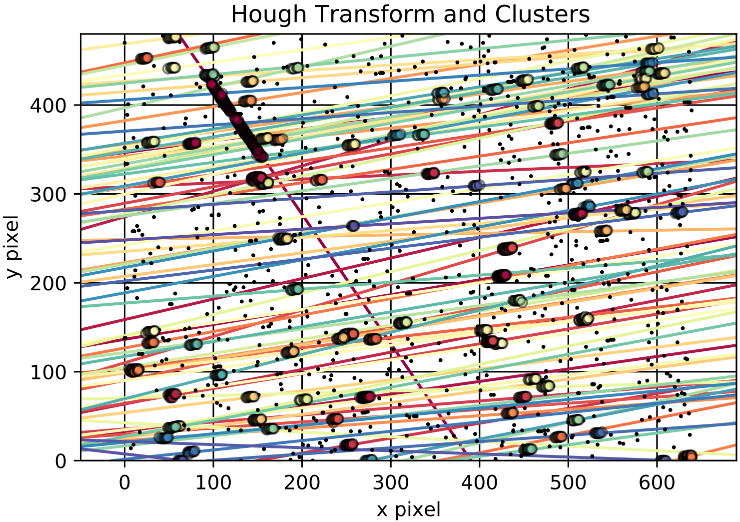


State of the Art Event Generation Simulation, Delbruck, Hu, and He 2021, `https://sites.google.com/view/video2events/home`

Prior Contributions Overview



Prior Contributions Overview

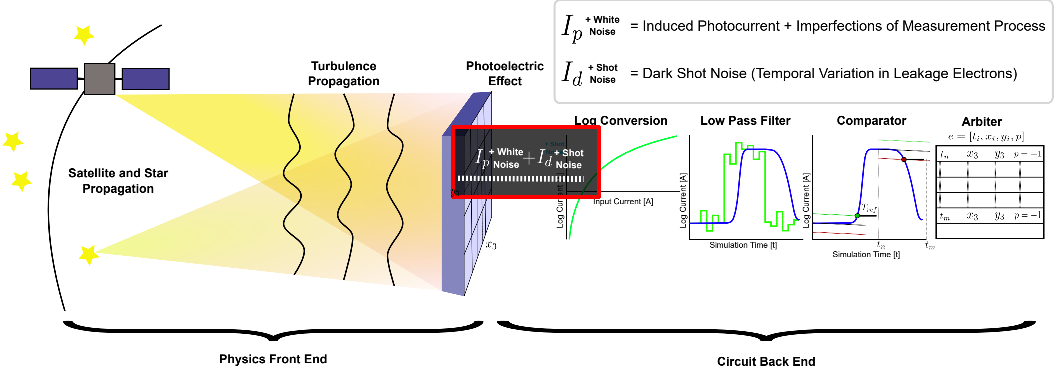


New Contributions

- Photon-Level Event Generation Simulation
  - Dark Shot Noise Generation
  - Circuit Noise Generation
- Event Data Point Source Tracking Algorithm
  - Tracking Methodology
  - Online Grouping Methods
  - Group Classification Methods

Outline

Event-based Sensor Model Updates

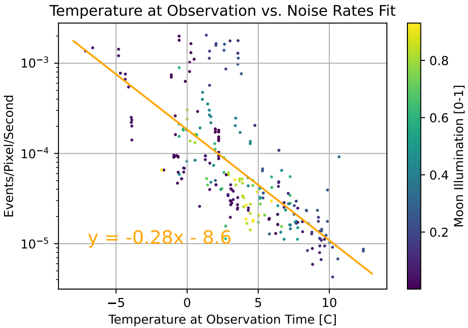


Dark Shot Noise State of the Art Methodology

- Sets noise generation from chosen event rate

$$r = ((F - 1) \times (I_{pd}/I_{max}) + 1) \times R_n$$
$$p = r \times \delta t$$
$$u < p : event_{OFF}$$
$$u > (1 - p) : event_{ON}$$

- Modeling at electron level provides greater insight into EVS noise behavior

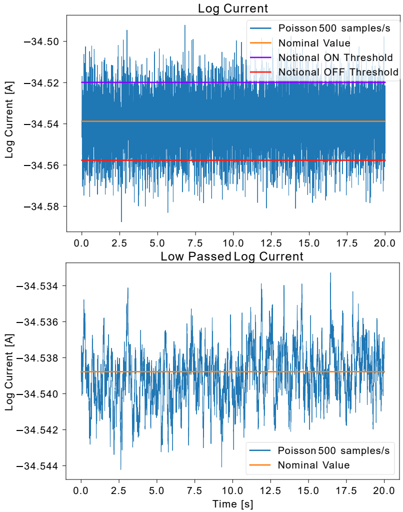


Dark Shot Noise Proposed Methodology

- Model the shot noise on dark current as Poisson distribution at the electron level

$$P(X = k) = \frac{\lambda_e^k e^{-\lambda_e}}{k!}$$

- Issue with the low pass filter
  - Simulation rate can be up to microsecond rate
  - High frequency Poisson pull does not make it through the low-pass filter
- Proposed solution rolling summation of Poisson

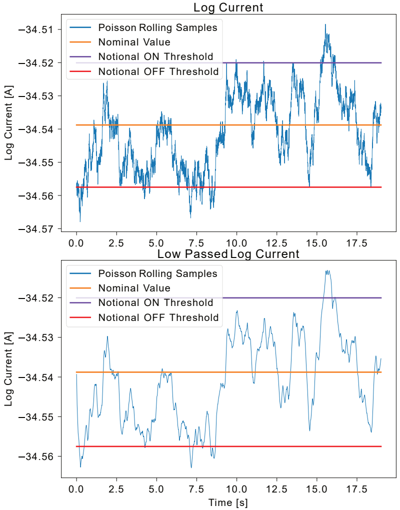


Rolling Poisson Summation

$$\underbrace{\begin{bmatrix} P_l(n_1,m_1) \\ P_l(n_2,m_1) \\ \vdots \\ P_l(n_n,m_1) \end{bmatrix} \begin{bmatrix} P_i(n_1,m_1) \\ P_i(n_2,m_1) \\ \vdots \\ P_i(n_n,m_1) \end{bmatrix}}_{\text{L steps in one second}} \underbrace{\begin{bmatrix} P_1(n_1,m_1) & P_1(n_1,m_2) & \dots & P_1(n_1,m_m) \\ P_1(n_2,m_1) & P_1(n_2,m_2) & \dots & P_1(n_2,m_m) \\ \vdots & & \ddots & \\ P_1(n_n,m_1) & P_1(n_n,m_2) & \dots & P_1(n_n,m_m) \end{bmatrix}}_{\text{M pixels in the x direction}} \Bigg\} \text{N pixels in the y direction}$$

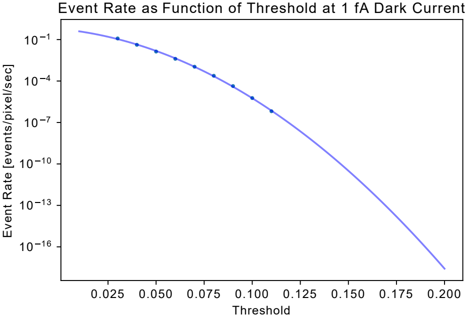
Rolling Poisson Performance

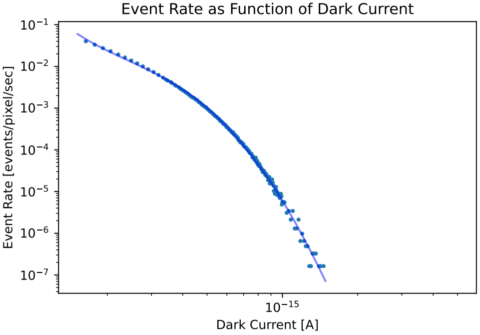
- Desire to Maintain Variation from Nominal
- Maximum Variation Loss
  - Traditional $= 92.3\%$
  - Rolling $= 5.4\%$
- Standard Deviation Loss
  - Traditional $= 92.2\%$
  - Rolling $= 7.5\%$
- Both generate slightly leptokurtic distributions
- Rolling poisson maintains deviations capable of passing threshold



Distribution Analysis

Dark Shot Noise Sensitivity

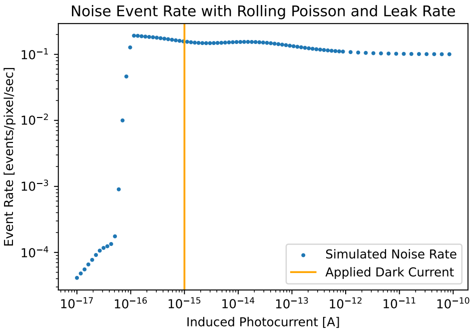




Circuit Noise State of the Art Methodology

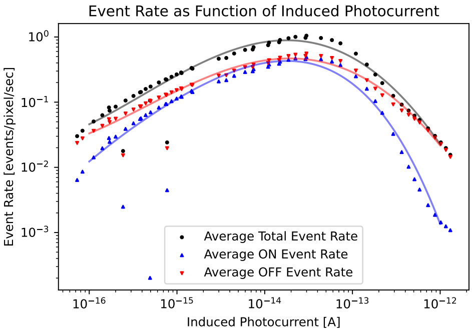
- Produces consistent 0.1 Hz rate of noise events
- All events induced are ON events
- Dark shot noise dominates at lower induced current levels

$$\delta_{leak} = \Delta t R_{leak} \theta_{ON}$$



Problems With State of the Art Circuit Noise Model

- Data from integrating sphere deviates from consistent noise rate
- High frequency noise in circuit at low photocurrent levels dominates over leak noise



New Circuit Noise Model

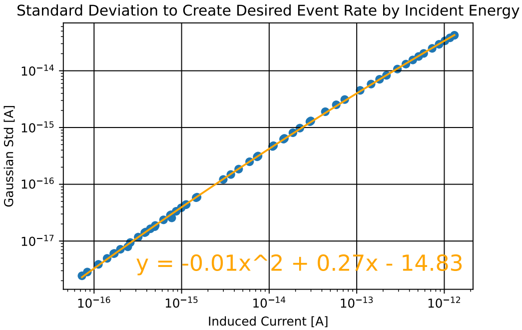
- Treat noise rate as area under a Gaussian distribution

$$\arg\min\left((P_{event} - erfc(z))^2\right)$$

- Find standard deviation for empirical data based on negative threshold

$$\sigma = \frac{|x-\mu|}{z}$$

- Quadratic fit to the standard deviation

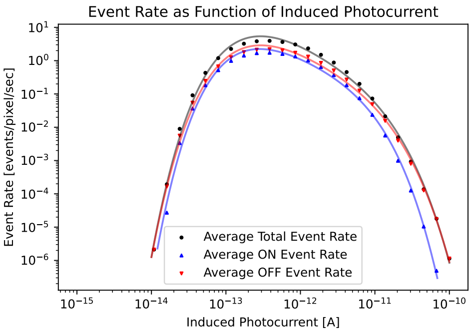


Simulated Noise Rate

- Offset of ON and OFF events
- Standard deviation drop off of events at higher induced photocurrents
- Low-pass filter drop off of events at lower induced photocurrents
- Low end drop off steepness tied to frequency cutoff model

$$f_{3dB} = \frac{(I_{dark} + I_{pd})}{I_{dark}} \times f_{3dBmin}$$

$$f_{3dB} \leq f_{3dBmax}$$

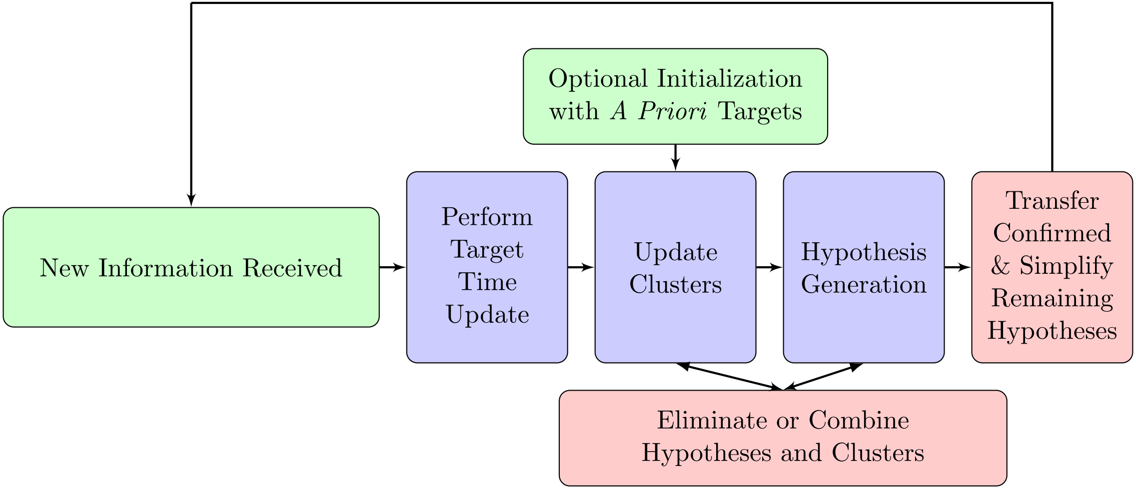


Model Improvement Summary

1. Dark shot noise as rolling Poisson
   - Maintains expected distribution characteristics
   - Carries enough inertia in current value to allow for noise generation with comparison operations
   - Two tuning parameters: Threshold and Dark Current
2. Tuned Gaussian for real signal circuit noise
   - Generates a peak in noise event rate
   - Generates ON and OFF events with OFF events favored
   - Tuning required with better estimate of dark current and thresholds

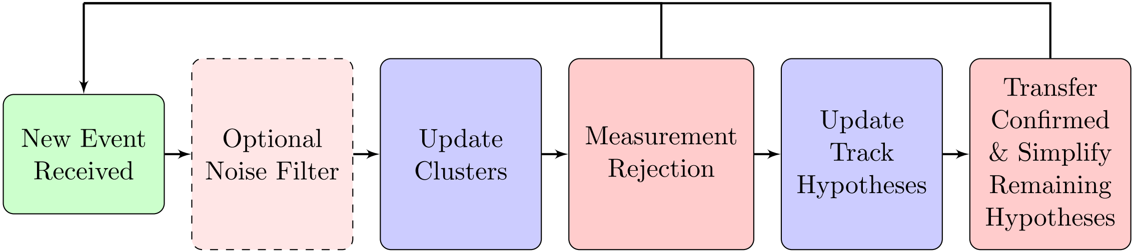
Outline

Classic Multiple Hypothesis Trackers



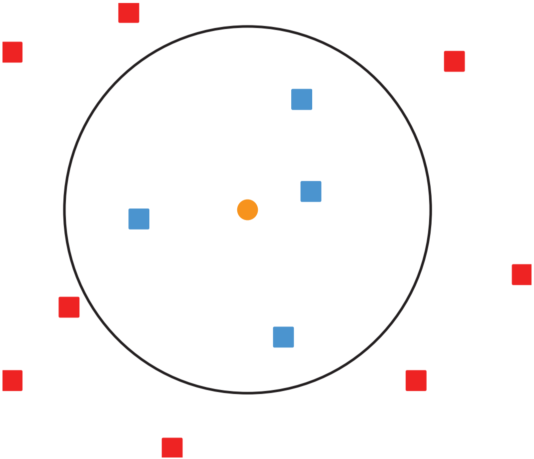
"An Algorithm for Tracking Multiple Targets," Reid 1979

Proposed Event-Based Tracking Algorithm



Proximity-Based Grouping

- Inspired by temporal clustering for labeling batch data
- Assign each event to a hypothesis group
- Select from most recent pixels from prior hypotheses or generate new hypothesis
- Original noise filter: only group high frequency events



RANSAC-Based Grouping

- Selects cuboid of events around newest event
- Linear regression fit & RANSAC with linear model
- RANSAC selects two data points at random and draws a line between them
- All proceeding events determine total inliers
- Select line with most inliers
- Not only groups, but provides linear slope and intercept

Grouper Results Comparison

- Optimality 1: Percent Events Properly Grouped

$$group_{opt} = \frac{r + n}{event_{tot}}$$

- Optimality 2: Percent Events Properly Grouped Assuming Noise Group Rejection

$$group_{opt2} = \frac{r + n + m}{event_{tot}}$$

- r = real events properly grouped
- n = noise events not grouped
- m = noise events in majority noise group

Grouper	**Opt 1**	**Opt 2**	**Real Duplicates**
Proximity-Based	91.0%	93.0%	697
Proximity-Based No Pruning	73.4%	98.9%	1587
RANSAC	94.5%	96.2%	33

Table: Grouping Methods Optimality & Real Duplicate Groups Compared

Pixel-Level Hypothesis Test Inspiration

Bayes Rule:

Probability of Detection A Given B

Probability of Detection A

$$P(A|B) = \frac{P(B|A)\ \ P(A)}{P(B)}$$

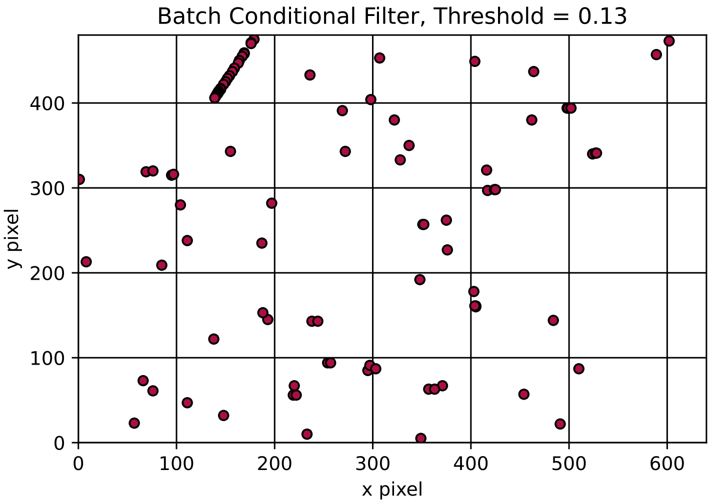
Conditional Probability of Attribute B Given Detection A

Probability of Attribute B

Event Profile	Star	Satellite
(1,0)	0.12	0.17
(1,0,0)	0.05	0.006
(1,1)	0.02	0.02
(1,1,0)	0.04	0.12
(1,1,0,0)	0.05	0.06
(1,1,0,0,0)	0.02	0.01
(1,1,1,0)	0.009	0.03
(1,1,1,0,0)	0.02	0.06
(1,1,1,0,0,0)	0.02	0.02
(1,1,1,0,0,0,0)	0.01	0.005

Table: Conditional probabilities of profiles of events given star or satellite detections. Positive Event = 1. Negative Event = 0.

Applying Bayes Rule to Event Profile



Bayesian Joint Conditional Probability of Pixels

Conditional Probability of Attributes Given Detection A

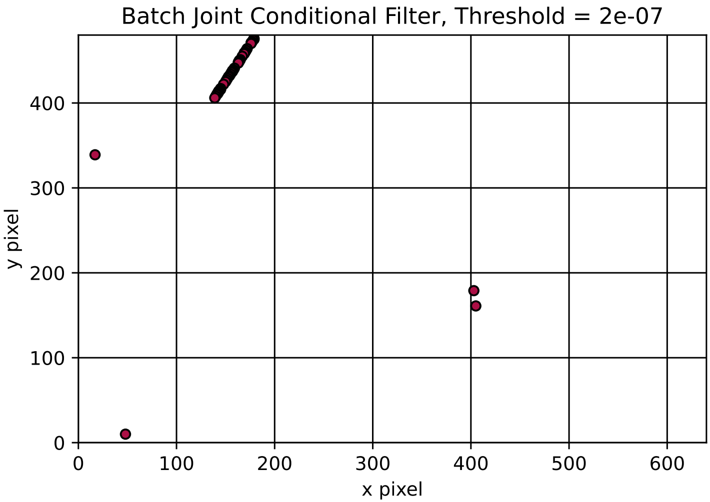
Probability of Detection A

$$P(A|B \wedge C) = \frac{P(B|A) \;\; P(C|A) \;\; P(A)}{P(B|A)P(C|A)P(A) + \;\; P(B|\sim A) \;\; P(C|\sim A) \;\; P(\sim A)}$$

Conditional Probability of Attributes Given Detections Other than A

Probability of Detections Other than A

Applying a Joint Conditional Probability Threshold



Pixel-Level Tracking Performance

	Predict Hot Pix	Pred Noise	Pred Star	Pred Sat
Actual Hot Pix	0	5242	13601	558
Actual Noise	0	123108	12128	17298
Actual Star	0	30609	351974	4929
Actual Sat	0	6191	7435	12392

Table: Pixel-wise Bayesian Classifier Online with Proximity Grouper Confusion Matrix

	Hot Pixel	Noise	Star	Sat
TPR	0	0.807086	0.908292	0.476286
TNR	1	0.90289	0.832465	0.959272

Table: Pixel-wise Bayesian Classifier Online with Proximity Grouper TPR and TNR

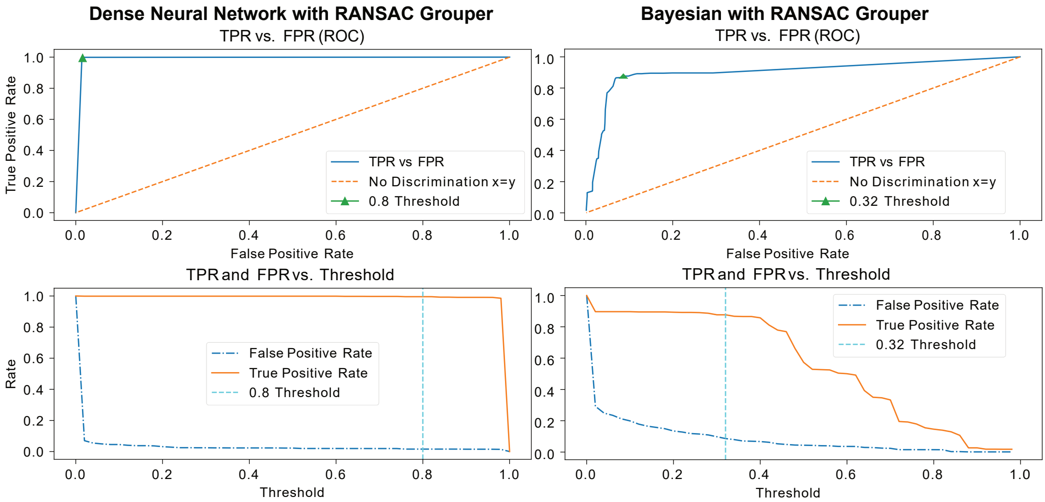
Pivot to Group-Level Classification

- Use grouped event information (time, x, y, polarity) as input features
- 70/30 Train/Validation data split
- Initially compared several FOSS-library Classifier Models and custom Convolutional and Dense Neural networks
- Chose Random Forest and Dense Neural Network models for tuning and integration with online groupers
- Tuned by training on RANSAC groupings, ROC analyses, and having multiple group-size models

Model	Star TPR	Sat TPR	Star TNR	Sat TNR
KNN	0.96173	0.93700	0.97499	0.98393
Gaussian NB	0.85800	0.87008	0.95149	0.93451
Random Forest	0.98108	0.98016	0.99305	0.99200
Extra Trees	0.98081	0.76108	0.91053	0.99193
Gradient Boosting	0.92277	0.96291	0.98410	0.96504
Dense Network	0.98889	0.99642	0.99808	0.99526
CNN	0.97562	0.98092	0.98339	0.98643

Table: ML Model TPR and TNR for Stars and Satellites based on offline Validation Data

Choosing Thresholds



Classifier Results

Satellite	RANSAC w/Star Filter		Proximity	
Model	**TPR**	**TNR**	**TPR**	**TNR**
Bayesian	0.7474	0.9970	0.3517	0.9539
Random Forest	0.9688	0.9997	0.9817	0.9924
Dense Network	0.9762	0.9995	0.9516	0.9988

Table: Final Classifier Performance Satellite TPR & TNR

Satellite	RANSAC w/Star Filter				Proximity			
Model	**TP**	**TN**	**FP**	**FN**	**TP**	**TN**	**FP**	**FN**
Bayesian	19445	557765	1682	6573	9151	533678	25769	16867
Random Forest	25207	559260	187	811	25543	555186	4261	475
Dense Network	25399	559164	283	619	24759	558723	724	1259

Table: Final Raw Classification Numbers

Final Online Classification

Bottom Line: This grouper and classifier combination can isolate satellite groups!

Tracker Development Summary

- Focus on filtering prior to state estimation with 2 steps
  1. Grouping
     - Proximity-based serves as a starting point
     - RANSAC improves grouping optimality metrics
  2. Classification
     - Pixel profile of events with Bayesian classifier serves as starting point
     - Multiple features, not just pixel polarity profile can contribute to classification
     - Dense Network and Random Forest with RANSAC improves overall classification and computational speed
- Proof of concept of online data rejection
- Next Step: Take filtered data to inform satellite state estimation and build a star tracker

Summary of Presented Results

1. Developed new modeling techniques for noise in synthetic data stream creation
    1. Utilizes the available electron rate information
    2. Dark shot noise as rolling Poisson
        - Inertia of rolling summation allows for variance to trigger events
    3. Tuned Gaussian for high frequency noise
        - Follows noise rate behavior on real signals
2. Developed non-frame-based data rejection techniques for online event-based tracking algorithms
    1. Proves pixel profile information and grouped attributes provide enough information for classification
    2. Develops two grouping techniques: Proximity-based & RANSAC-based
    3. Develops pixel-level and group-level classification techniques
    4. Combination of DNN and RANSAC grouper outperforms others in data rejection

Contributions Overview: Code

- Event-based Sensor Model for Space Domain Awareness, various modules (physics front end, noise generation, realistic event trigger timings, rate-limited arbiter)
- Event-based post processing clustering algorithm for data association and labeling
- Event-based Online RANSAC Grouper
- Event-based Online Proximity Grouper
- Dense Neural Network for Event Group Classification
- Trained Event Group Classification Models (DNN, Random Forest, Bayesian)

Contributions Overview: Papers

- Conference Papers:
  - Event-Based Sensor Multiple Hypothesis Tracker For Space Domain Awareness, Advanced Maui Optical and Space Surveillance Technologies Conference 2022
  - Event-based Sensor Model for Space Domain Awareness, Advanced Maui Optical and Space Surveillance Technologies Conference 2021
  - Co-author: Modeling and Decoding Event-Based Sensor Lightning Response, SPIE Optical Engineering + Applications 2023
  - Co-author: Demystifying Event-based Sensor Biasing to Optimize Signal to Noise for Space Domain Awareness, Advanced Maui Optical and Space Surveillance Technologies Conference 2023
- Journal Papers:
  - Verification of a Satellite Observing Event-Based Sensor Model, In Preparation 2024
  - Algorithmic Methods for Online Grouping and Classification of Point Source Event-Based Data, In Preparation 2024
  - Co-author: Event Camera Optimization and Photometric Reconstruction for Space Applications, IEEE, In Preparation 2024

Contributions Overview: Presentations & Proposals

- Presentations:
  - Conference Short Course: An Introduction to Event-Based Sensors for SDA: A Hands-On Tutorial, Advanced Maui Optical and Space Surveillance Technologies Conference 2022
  - Invited Presentation: Modeling and Tracking Algorithms for Space Domain Awareness with Event-Based Sensors, Event-Based Sensing Community of Interest, Air Force Research Laboratories, 2023
  - Invited Presentation: Modeling and Tracking Algorithms for Space Domain Awareness with Event-Based Sensors, Raytheon Event-Based Sensing, Processors, Algorithms, and Applications, 2023
- Proposals Written and Awarded:
  - EVS Research AFIT New Faculty Research: $10k
  - EVS Research with Air Force Research Laboratory: $37k
  - EVS Research with NPS: $50k
- NATO Research Task Group: SET-347 Applications of Neuromorphic Sensors, Invited Technical Team Member

Acknowledgements

Thank you for attending my B exam!

Special thank you to my committee members, colleagues, family, and friends for their support over the past 4 years.

Rachel Oliver

ro234@cornell.edu

References I

Delbruck, T., Y. Hu, and Z. He (2021). "V2E: from video frames to realistic DVS event camera streams". In: Conference on Computer Vision and Pattern Recognition Workshops.

Office, ESA Space Debris (2023). *ESA's Annual Space Environment Report*. https://www.esa.int/Space_Safety/Space_Debris/ESA_s_Space_Environment_Report_2023. Darmstadt, Germany.

Reid, D. (1979). "An Algorithm for Tracking Multiple Targets". In: *IEEE Transactions on Automatic Control* 24.6.

The Astrosite: A Mobile Neuromorphic Space Domain Awareness Observatory (2019). https://greg-cohen.com/project/astrosite/. Accessed: 2024-07-24.

Contributions Overview: Teaching & Advising

- Teaching Graduate Level Courses:
  - Instructor: Introductory Spaceflight Dynamics at Air Force Institute of Technology (Winter 2024)
  - Instructor: Intermediate Spaceflight Dynamics at Air Force Institute of Technology (Spring 2024)
  - Instructor: Spacecraft Design and Test at Air Force Institute of Technology (Summer 2024)
- Student Advising:
  - Committee Member for current PhD Candidate: EVS event frequency exploitation
  - Committee Member for PhD Student: EVS star tracking
  - Committee Member for MS Student (2023): Event-based Tracking of Target in a Threat Affect Scene
  - Committee Member for MS Student (2024): Disposal and Earth-Moon Escape from Cislunar Orbits using Non-Linear Programming Techniques

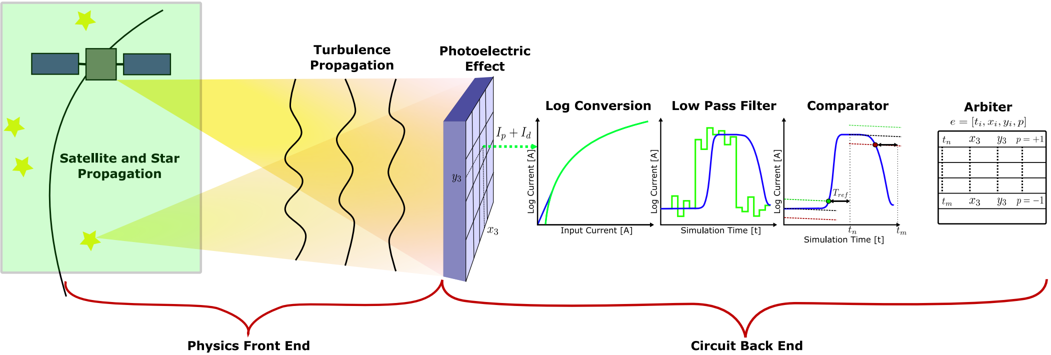
Courses Taken

- For Credit
  - AEP 5400 - Nonlinear and Quantum Optics
  - MAE 5180 - Autonomous Mobile Robots
  - MAE 5730 - Intermediate Dynamics
  - MAE 5780 - Feedback Control Systems
  - MAE 5830 - Astronautic Optimization
  - MAE 6760 - Model-Based Estimation
  - MAE 6780 - Multivariable Control Theory
- Audited
  - MAE 6280 - Adaptive and Learning Systems
  - MAE 6700 - Advanced Dynamics
  - CS 5780 - Intro to Machine Learning

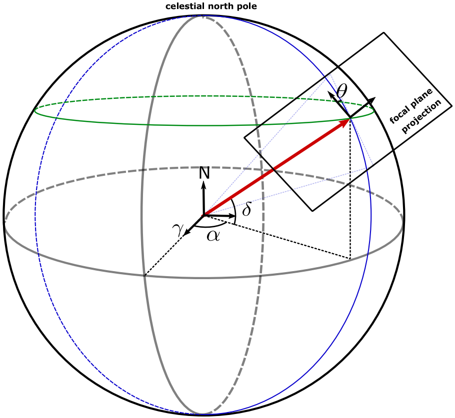
Additional Graduate Level Courses

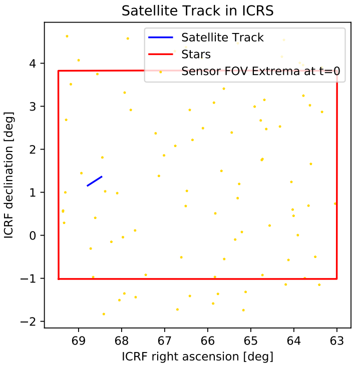
- Taken at the Air Force Institute of Technology
  - Linear Systems Analysis
  - The Space Environment
  - Methods of Applied Mathematics
  - Intermediate Dynamics
  - Space Mission Analysis and Design
  - Satellite Communications
  - Control and State Space Concepts
  - Introduction to Flight Dynamics
  - Applied Linear Alegbra
  - Intermediate Space Flight Dynamics
- Spacecraft Systems Engineering
- Satellite Design and Test
- Modern Methods of Orbital Determination
- Space Surveillance
- Chemical Rocket Propulsion

New Event-based Sensor Model



Satellite and Star Generation



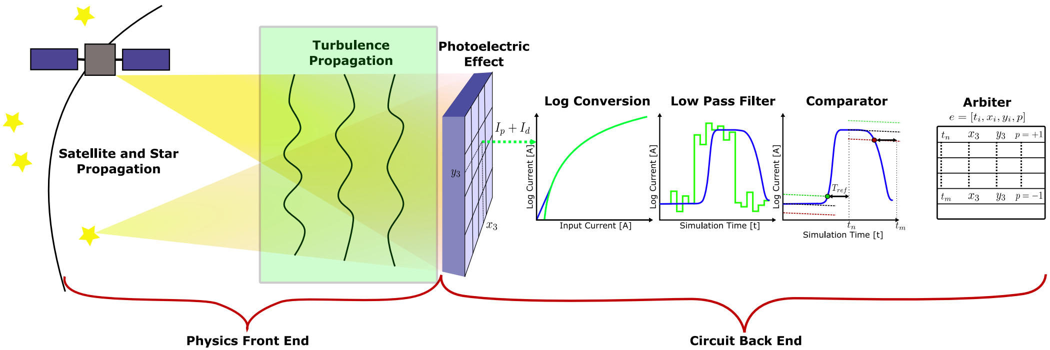


Irradiance Calculations

Return

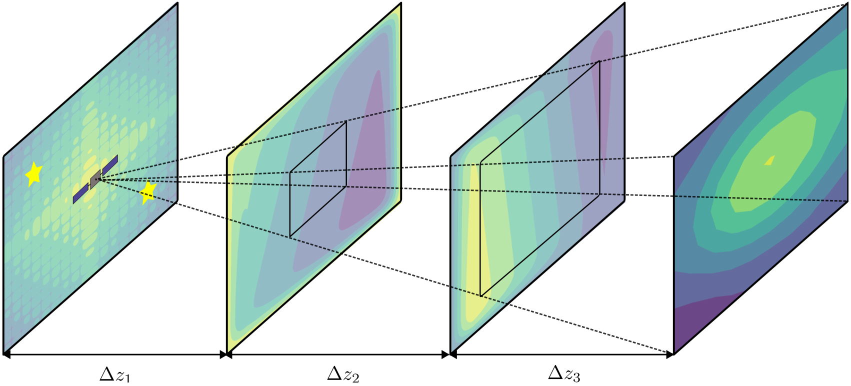
- Star by Gaia G: $V_0 = q_v - 2.5log(F_v)$
  - q_v zero magnitude flux scaled from Vega
  - Rearrange Equation for Flux
  - $F_v = 10^{-0.4(V_0 - q_v)}[\frac{W}{cm^2 Angstrom}]$
  - Multiply by bandpass, filter width, and recieving area
  - $P_v = F_v Area Bandpass$
- Satellite by Point Source:
  - Power sent to hemisphere
  - $I = \frac{L \sum(A_{surf} \rho_{surf})}{2 * \pi}$
  - Calculate energy recieved per steradian
  - $\mathcal{E}_\lambda = \frac{I_\lambda * cos(\theta_R)}{R^2}$
  - Multiply by receiving area
  - $P_{sat} = \mathcal{E}_\lambda Area$

New Event-based Sensor Model



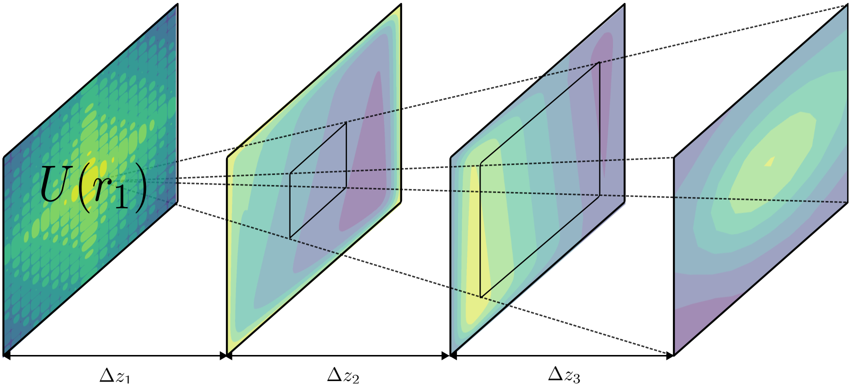
Simulating Atmospheric Propagation

Return



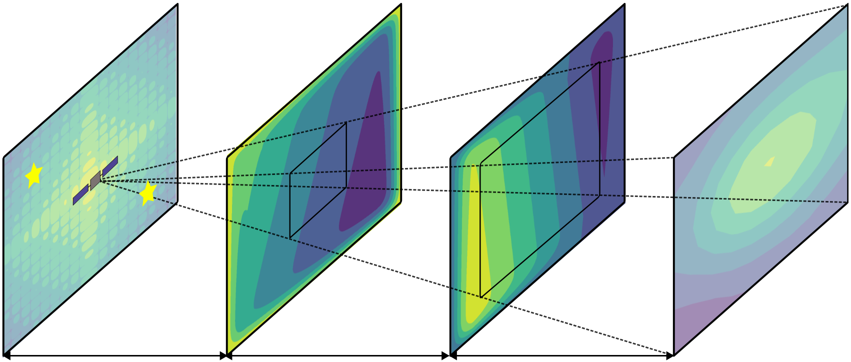
$$U(r_n) = \mathcal{Q}\left[\frac{m_{n-1}-1}{m_{n-1}\Delta z_{n-1}}, r_n\right] \times \prod_{i=1}^{n-1}\left\{\mathcal{T}[z_i, z_{i+1}]\mathcal{F}^{-1}\left[f_i, \frac{r_{i+1}}{m_i}\right]\mathcal{Q}_2\left[-\frac{\Delta z_i}{m_i}, f_i\right]\mathcal{F}[r_i, f_i]\frac{1}{m_i}\right\} \times \left\{\mathcal{Q}\left[\frac{1-m_1}{\Delta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

Simulating Atmospheric Propagation



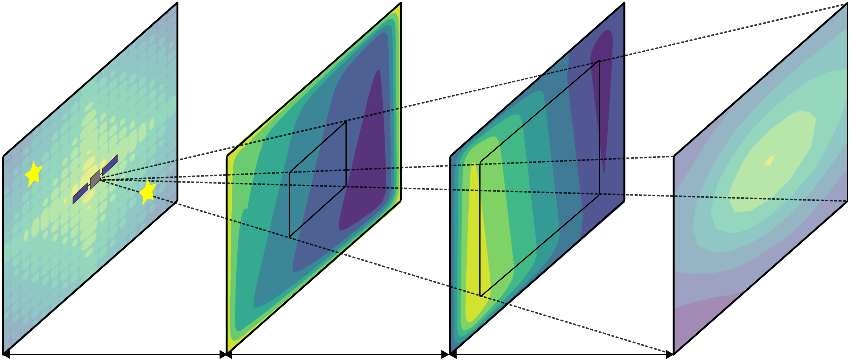
$$U(r_n) = \mathcal{Q}\left[\frac{m_{n-1}-1}{m_{n-1}\Delta z_{n-1}}, r_n\right] \times \prod_{i=1}^{n-1}\left\{\mathcal{T}[z_i, z_{i+1}]\mathcal{F}^{-1}\left[f_i, \frac{r_{i+1}}{m_i}\right]\mathcal{Q}_2\left[-\frac{\Delta z_i}{m_i}, f_i\right]\mathcal{F}[r_i, f_i]\frac{1}{m_i}\right\} \times \left\{\mathcal{Q}\left[\frac{1-m_1}{\Delta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

Simulating Atmospheric Propagation



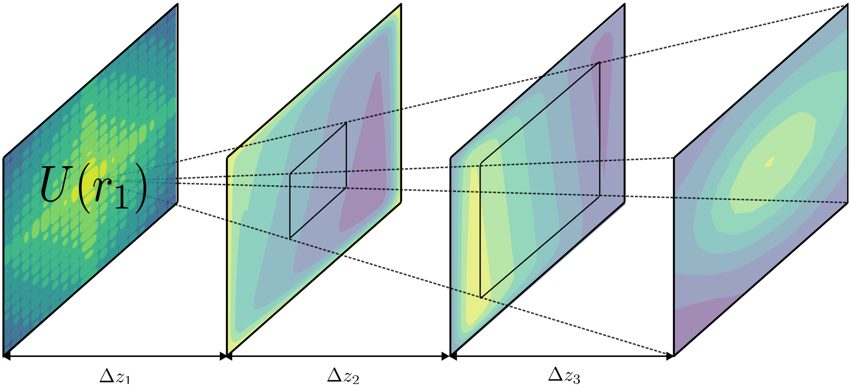
$$U(r_n) = \mathcal{Q}\left[\frac{m_{n-1}-1}{m_{n-1}\Delta z_{n-1}}, r_n\right] \times \prod_{i=1}^{n-1}\left\{\boxed{\mathcal{T}[z_i, z_{i+1}]\mathcal{F}^{-1}\left[f_i, \frac{r_{i+1}}{m_i}\right]\mathcal{Q}_2\left[-\frac{\Delta z_i}{m_i}, f_i\right]\mathcal{F}[r_i, f_i]\frac{1}{m_i}}\right\} \times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

Simulating Atmospheric Propagation



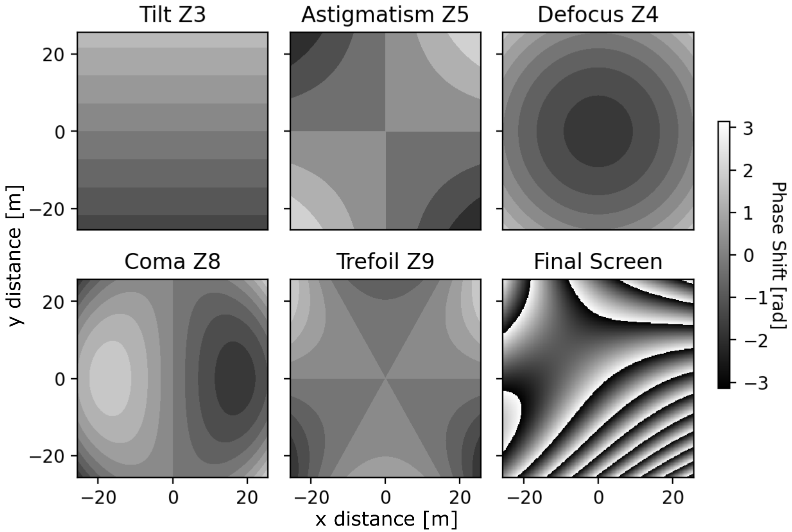
$$U(r_n) = \mathcal{Q}\left[\frac{m_{n-1}-1}{m_{n-1}\Delta z_{n-1}}, r_n\right] \times \prod_{i=1}^{n-1}\left\{\mathcal{T}[z_i, z_{i+1}]\mathcal{F}^{-1}\left[f_i, \frac{r_{i+1}}{m_i}\right]\mathcal{Q}_2\left[-\frac{\Delta z_i}{m_i}, f_i\right]\mathcal{F}[r_i, f_i]\frac{1}{m_i}\right\} \times \left\{\mathcal{Q}\left[\frac{1-m_1}{\Delta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

Simulating Atmospheric Propagation



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

Zernike Mode Phase Screen



Zernike Mode Weighting

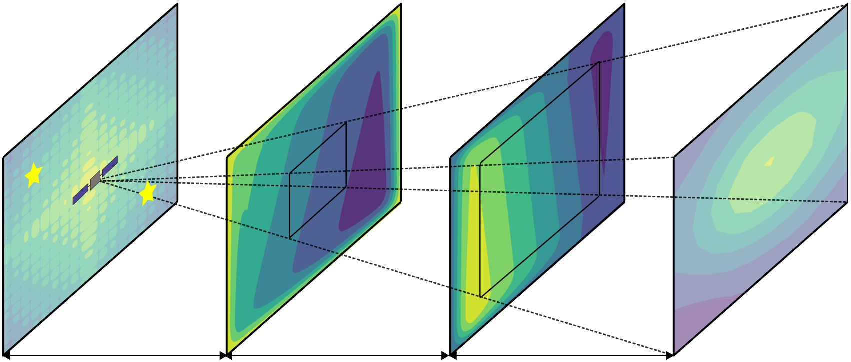
$$Z_{evenj} = \sqrt{n+1}R_n^m(r)\sqrt{2}cos(m\theta)$$
$$Z_{oddj} = \sqrt{n+1}R_n^m(r)\sqrt{2}sin(m\theta)$$
$$Z_j = \sqrt{n+1}R_n^0(r), m = 0$$

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

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

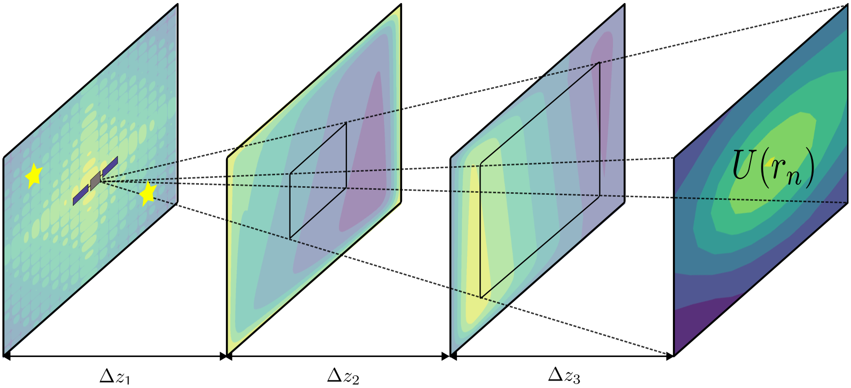
$$a_j = U^T \cdot \overrightarrow{n}.$$

Simulating Atmospheric Propagation



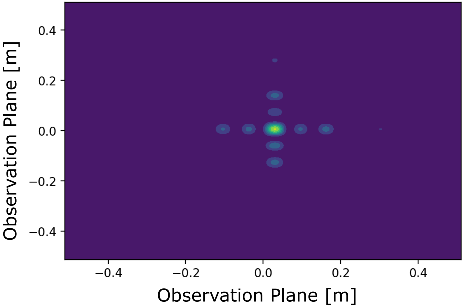
$$U(r_n) = \mathcal{Q}\left[\frac{m_{n-1}-1}{m_{n-1}\Delta z_{n-1}}, r_n\right] \times \prod_{i=1}^{n-1}\left\{\mathcal{T}[z_i, z_{i+1}]\mathcal{F}^{-1}\left[f_i, \frac{r_{i+1}}{m_i}\right]\mathcal{Q}_2\left[-\frac{\Delta z_i}{m_i}, f_i\right]\mathcal{F}[r_i, f_i]\frac{1}{m_i}\right\} \times \left\{\mathcal{Q}\left[\frac{1-m_1}{\Delta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

Simulating Atmospheric Propagation

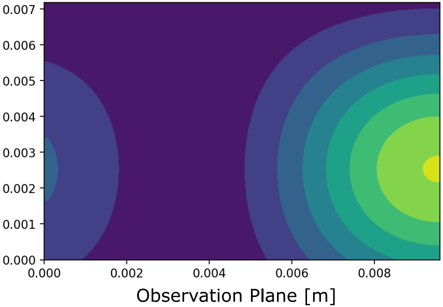


$$\boxed{U(r_n)} = \mathcal{Q}\left[\frac{m_{n-1}-1}{m_{n-1}\Delta z_{n-1}}, r_n\right] \times \prod_{i=1}^{n-1}\left\{\mathcal{T}[z_i, z_{i+1}]\mathcal{F}^{-1}\left[f_i, \frac{r_{i+1}}{m_i}\right]\mathcal{Q}_2\left[-\frac{\Delta z_i}{m_i}, f_i\right]\mathcal{F}[r_i, f_i]\frac{1}{m_i}\right\} \times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

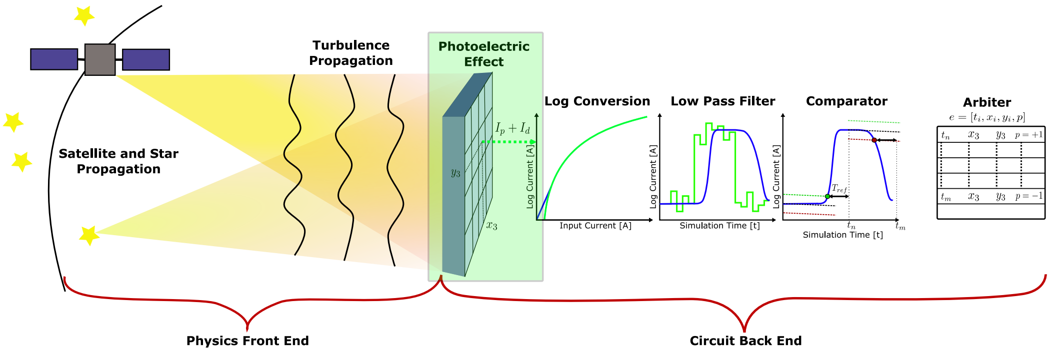
Simulating Atmospheric Propagation



$$g_2 = g_1 \circledast h$$



New Event-based Sensor Model



Induced Photocurrent

◂ Return

- Responsivity linearly scales Power, $\Phi[W]$

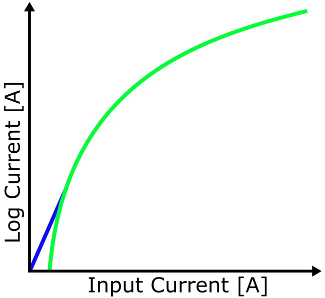
$$I_p = R_\lambda \Phi [A]$$

- Responsivity, R_λ, Scaled by Quantum Efficiency, η

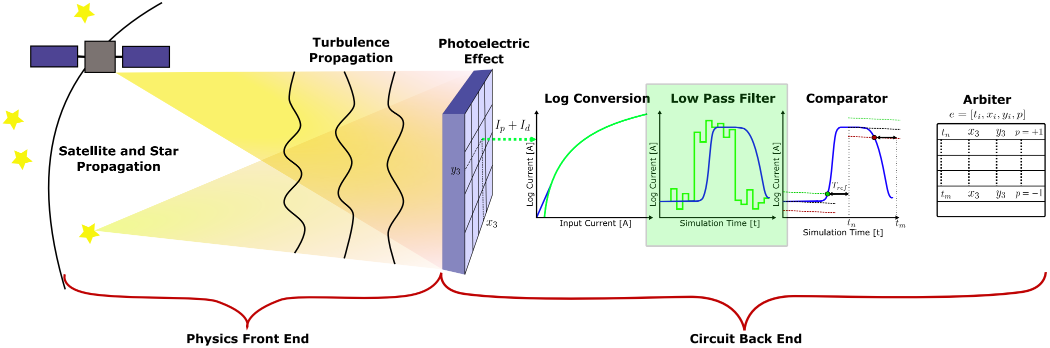
$$R_\lambda = \eta \frac{q}{hf} \approx \eta \frac{\lambda}{1.23985} \left[\frac{A}{W}\right]$$

- Temperature, T, Dependent Dark Current, $I_{darklog}$

$$I_{darklog} = \ln(I_{dark}) = constant - \frac{E_a}{kT}$$



New Event-based Sensor Model



Low Pass Filter

Return

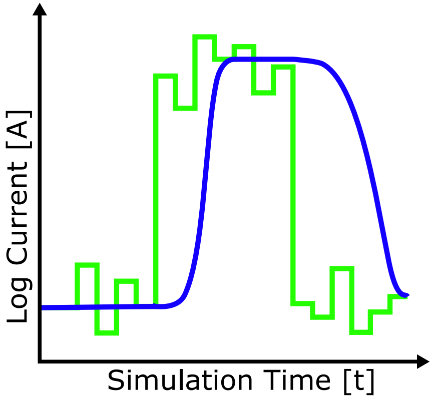
- Maximum bandwidth, $f_{3dBmax} \approx 3000[Hz]$
- Resultant bandwidth, f_{3dB}

$$f_{3dB} = \frac{I_{in} + (\frac{I_{max}}{10})}{I_{max}} \times f_{3dBmax}$$

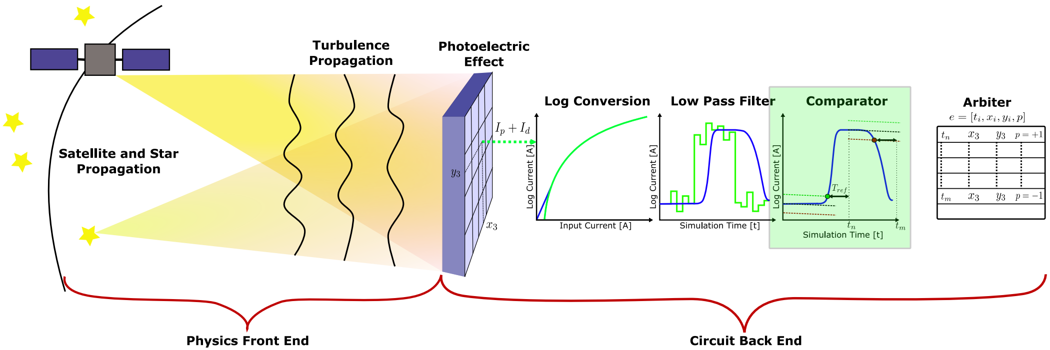
$$\epsilon = e^{2\pi * \Delta t f_{3db}}$$

$$I_{p_{int}} \longleftarrow (1-\epsilon)I_{p-1} + \epsilon I_{in}$$

$$I_p \longleftarrow (1-\epsilon)I_{p_{int}} + \epsilon I_{in}$$

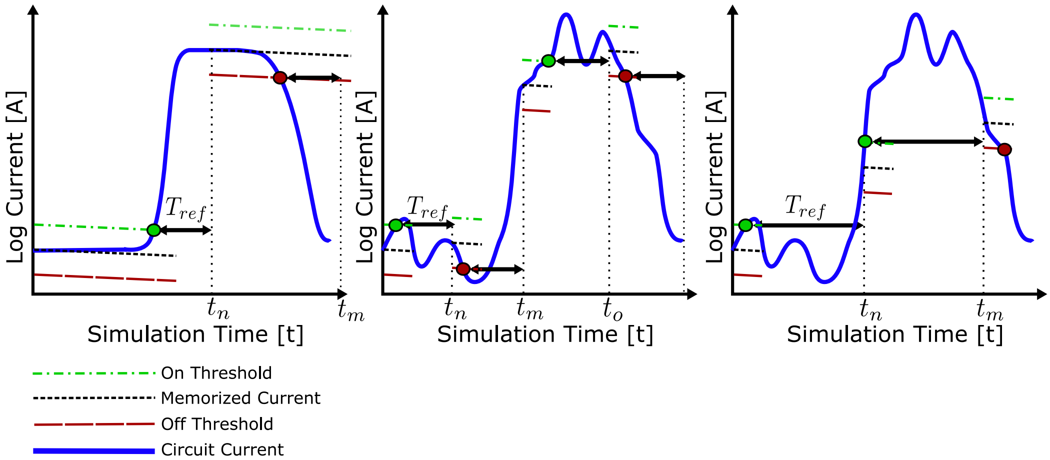


New Event-based Sensor Model



Comparator and Refractory Circuit

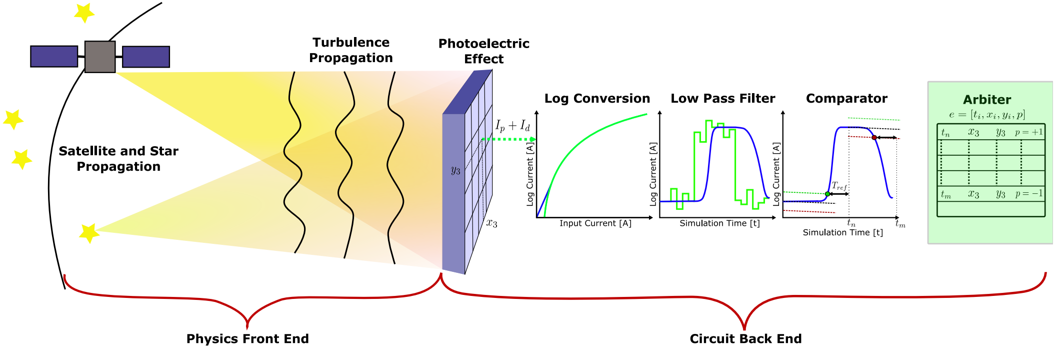
Return



Event Trigger Time Determination

$$t_1 = \begin{cases} t_{pos} = \frac{(\theta_{ON} - I_{p1} + I_{memlog1})\Delta t}{\Delta I_p - \Delta I_{memlog}} + t_0 & \Delta I_p > 0 \\ t_{neg} = \frac{(\theta_{OFF} + I_{p1} - I_{memlog1})\Delta t}{\Delta I_{memlog} - \Delta I_p} + t_0 & \Delta I_p < 0 \end{cases}$$

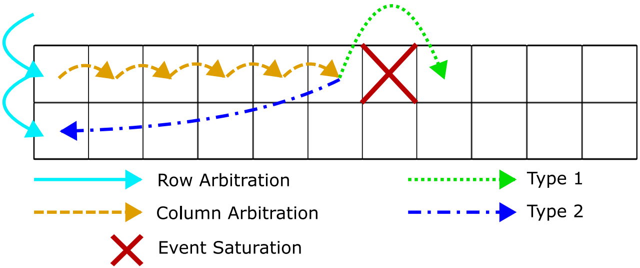
New Event-based Sensor Model



Arbitration

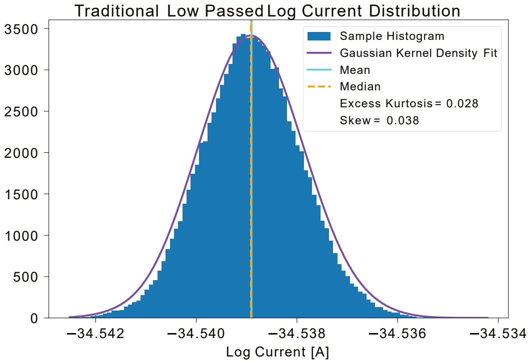
Return

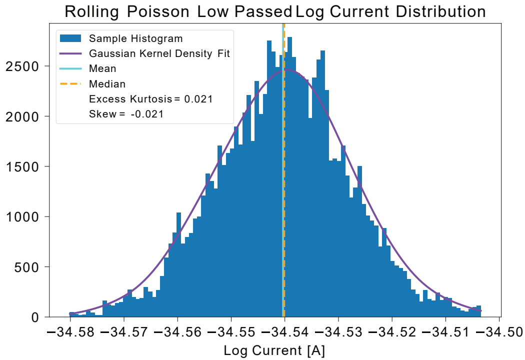
- Ordered row then column query
- Restricted maximum number of events
- Allows and tracks loss of events
  - Saturation
  - Refractory Period



Rolling Poisson Distribution Analysis

◂ Return





Verification

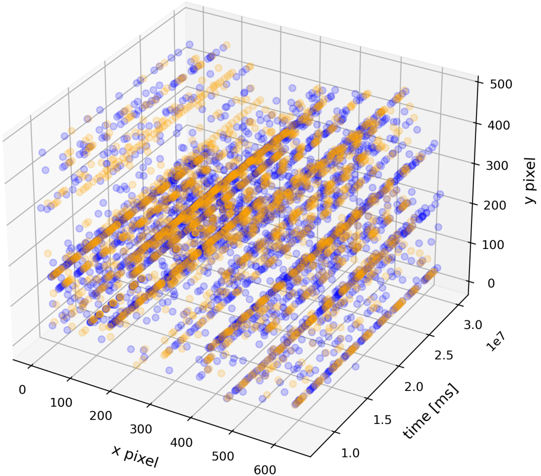
- Goal: Prove Model Creates Data Analogous to Real Data
  - Build trust in simulation output
  - Produce simulated data for algorithm development
  - Test camera settings and designs to maximize low-light performance
- Challenges:
  - Non-deterministic: no one-to-one matching
  - Unknown importance of data characteristics

Data Collection

- Prophessee Gen3 (VGA) camera
- 85mm f1.4 Lens
- January - March 2021
- Collection Procedure
  1. Slew to fixed azimuth and elevation
  2. Stare in fixed position for 30 seconds
  3. Slew to next location
- Courtesy of Dr. David Monet of AFRL Space Vehicles Directorate

Sony Prophesee, 2022
`https://www.prophesee.ai/event-based-evaluation-kits/`

Example Data

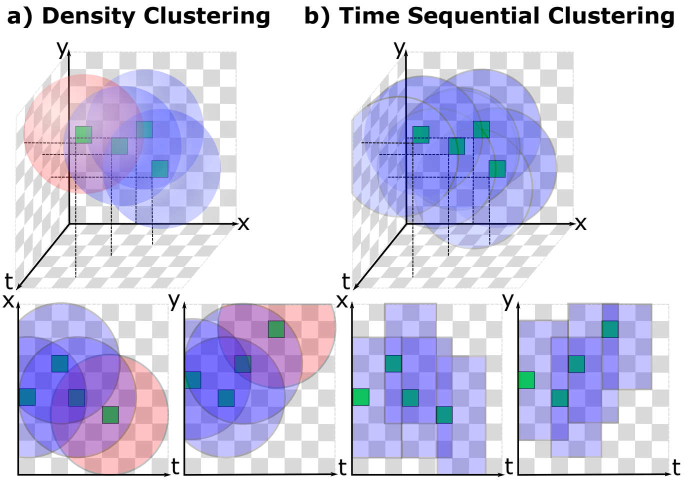


Analysis Approaches

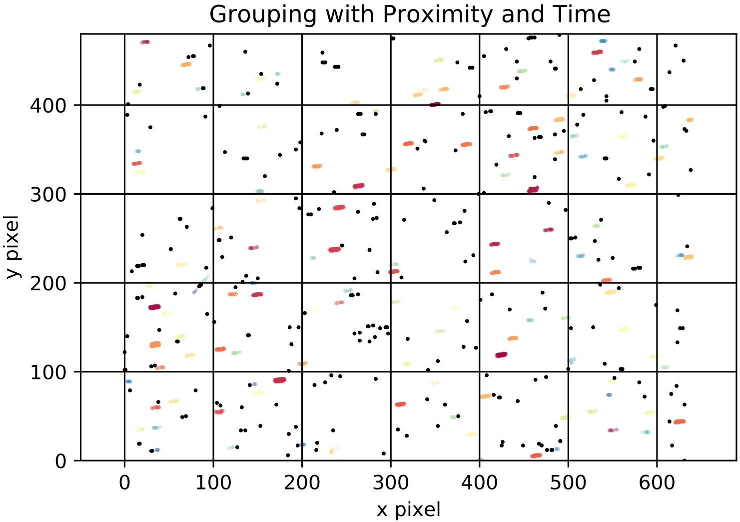
- Classic Post Processing
  1. Integrate data over 3 second time slice FITS image
  2. SExtractor yields: $(column, row, flux)$
  3. Conversion of flux to visual magnitude: $(column, row, v_{mag})$
  4. Astrometry.net maps $(column, row)$ to (RA, DEC) and identifies reference stars and satellites
  5. Correlate satellite (RA, DEC) from each time slice with a satellite detection

- New Temporal Data Association
  1. Cluster data in 3-dimensions
  2. Filter clusters with too few pixels and/or events
  3. Combine clusters with Hough transform
  4. Assign cluster numbers to satellites
  5. Extract signature and temporal characteristics

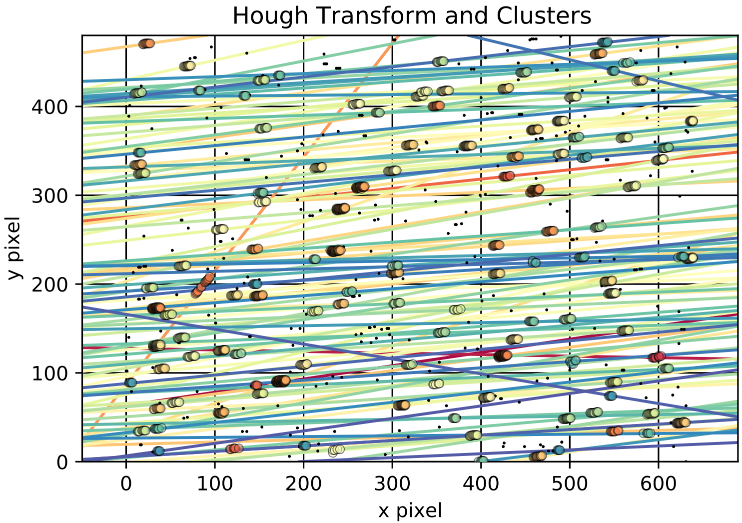
Clustering


a) Density Clustering
b) Time Sequential Clustering
y
x
t
x
t
y
t
y
x
t
x
t
y
t

Clustering

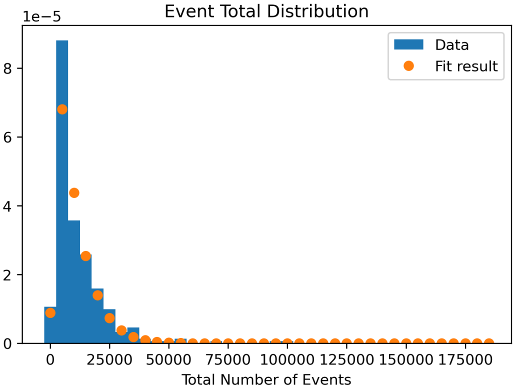

Grouping with Proximity and Time
y pixel
x pixel
0
100
200
300
400
500
600

Hough Transform

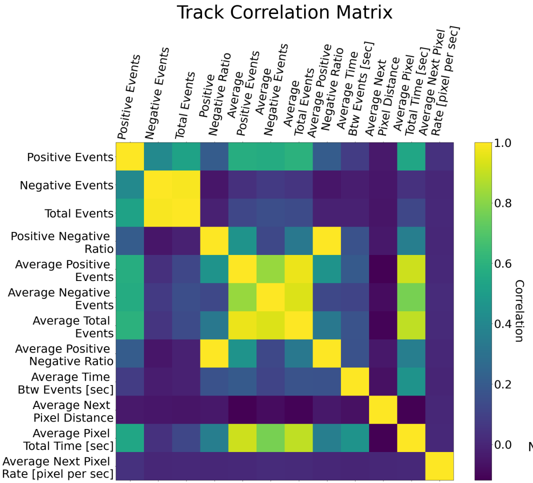


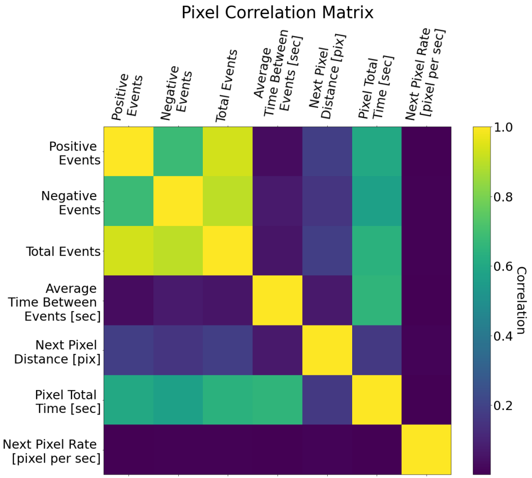
New Approach Limitations

- Human involvement is necessary
  - Easier to access in 2-dimensions
  - Poor signal to noise datasets are hard to process
- Time clustering is more consistent, but slow

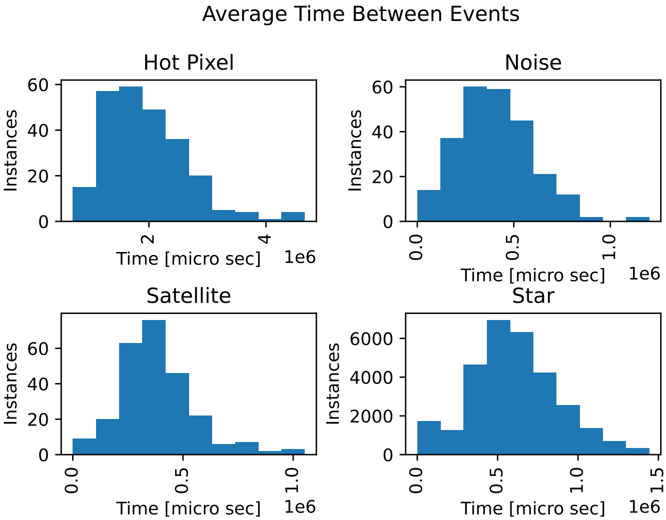


Selected Characteristics



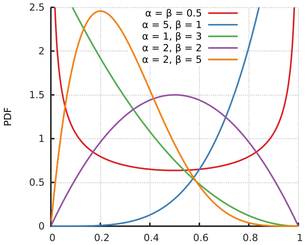


Characteristic Distributions



Beta Binomial Distribution Fit

- Unknown or random probability of success in each of a fixed or known number of Bernoulli trials
- Binomial distribution modification: probability of success is drawn from beta distribution
- Fit with selection of approximate α and β values defining the continuous beta distribution
- Defined the bin size and center values of those bins
- Curve fit the center locations to a continuous beta distribution



Horas, 2014

Validation Plan

1. Select at random a set of data from those post-processed
2. Simulate the physics front-end once
3. Run sensitivity analysis on camera parameters
4. Create Jacobian from sensitivity analysis
5. Find least squares fit of camera parameters
6. Use best fit parameters to simulate other collections
7. Compare the distributions with the Kolmogorov Smirnov Test

Non-linear Least Squares

- Sensitivity analysis is a finite difference approximation
- Produces Jacobian, J_{ij}

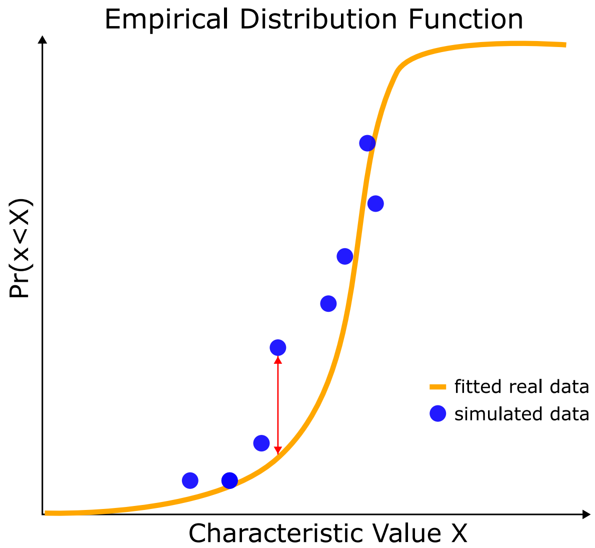
$$J_{ij} = -\frac{\delta r_i}{\delta \beta_j}$$

- Residuals, r_i
- Input Parameters, β_j
- Least Squares to find change in output, Δy

$$(J^T J)\Delta\beta = J^T \Delta y$$

$$\begin{bmatrix} \frac{\delta r_1}{\delta \beta_1} & \frac{\delta r_1}{\delta \beta_2} & \cdots & \frac{\delta r_1}{\delta \beta_j} \\ \frac{\delta r_2}{\delta \beta_1} & \frac{\delta r_2}{\delta \beta_2} & & \\ \vdots & & \ddots & \vdots \\ \frac{\delta r_i}{\delta \beta_1} & & \cdots & \frac{\delta r_i}{\delta \beta_j} \end{bmatrix}$$

Kolmogorov Smirnov Test



Event-based Sensor Star Tracker