

---

# Event Cameras for Dynamic Robot Control

**Gabriel Rodriguez**

**Advised by Navid Shahrestani**

**G76 Intern Presentations – 07/28/25**



DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited.

This material is based upon work supported by the Department of the Air Force under Air Force Contract No. FA8702-15-D-0001 or FA8702-25-D-B002. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Air Force.

© 2025 Massachusetts Institute of Technology.

Delivered to the U.S. Government with Unlimited Rights, as defined in DFARS Part 252.227-7013 or 7014 (Feb 2014). Notwithstanding any copyright notice, U.S. Government rights in this work are defined by DFARS 252.227-7013 or DFARS 252.227-7014 as detailed above. Use of this work other than as specifically authorized by the U.S. Government may violate any copyrights that exist in this work.

---



# Outline

---

- **Introduction**
- **Problem Definition**
- **Control Approach**
- **Contributions**
- **Results**
- **Future Work**
- **References**



# Outline

---

- **Introduction**
- Problem Definition
- Control Approach
- Contributions
- Results
- Future Work
- References



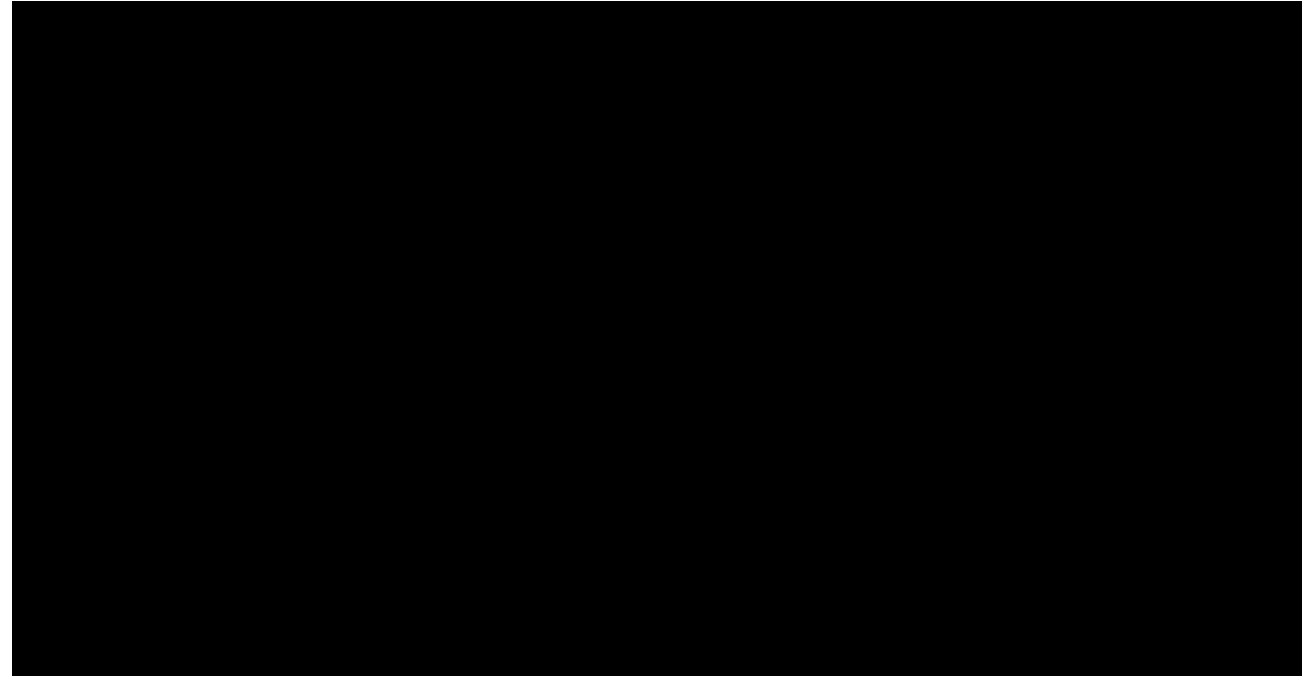
# Introduction

## Traditional Cameras

- Operate at a fixed frame rate
- Render the entire frame at each time step
- Are susceptible to motion blur

## Event Cameras (Dynamic Vision Sensors)

- Record changes in brightness (events) asynchronously
- Have high dynamic range & temporal resolution
- Perform well in low-light conditions



Event camera demonstration [1].



# Outline

---

- Introduction
- **Problem Definition**
- Control Approach
- Contributions
- Results
- Future Work
- References



# Problem Definition

How can we exploit the **temporal resolution** and **dynamic range** of event-cameras for robot control?



Isaac Sim obstacle avoidance environment (left), MuSHR Nano Robot [2] (top-right), Prophesee event camera [3] (bottom-right)



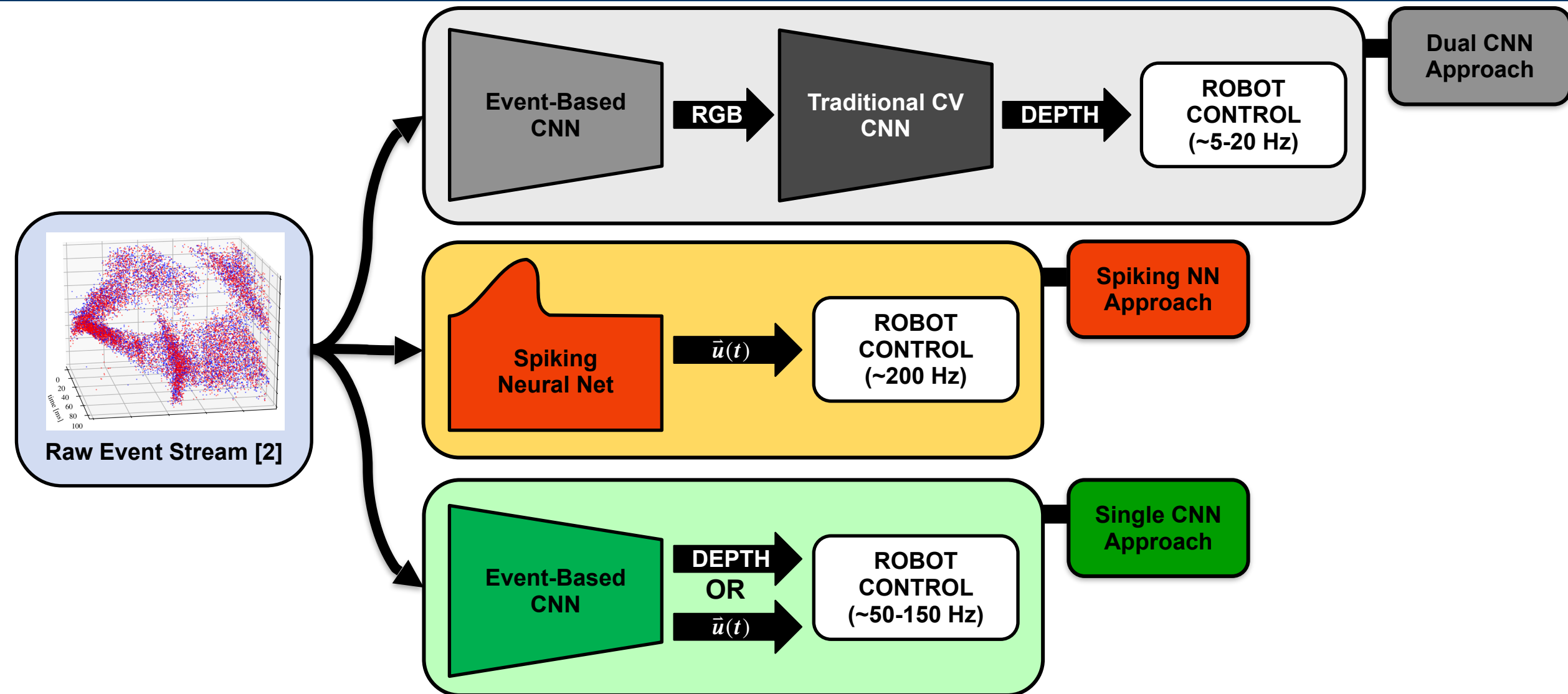
# Outline

---

- Introduction
- Problem Definition
- **Control Approach**
- Contributions
- Results
- Future Work
- References

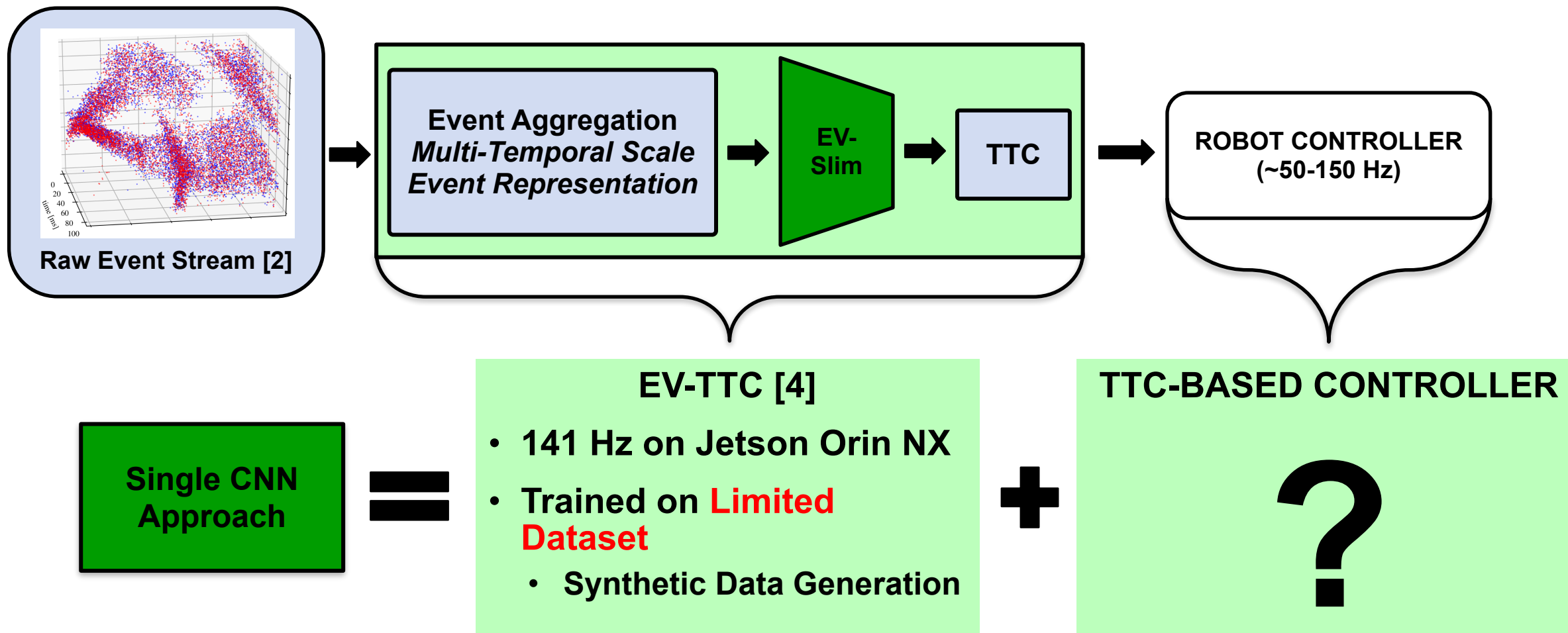


# Event-Based Control Approaches





# Chosen Control Approach





# Outline

---

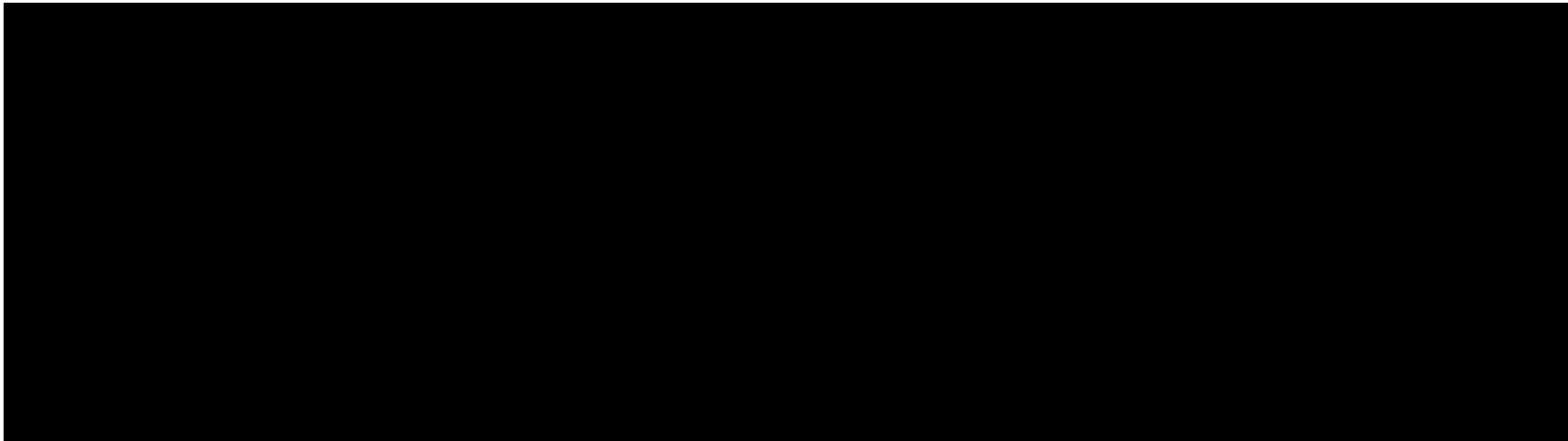
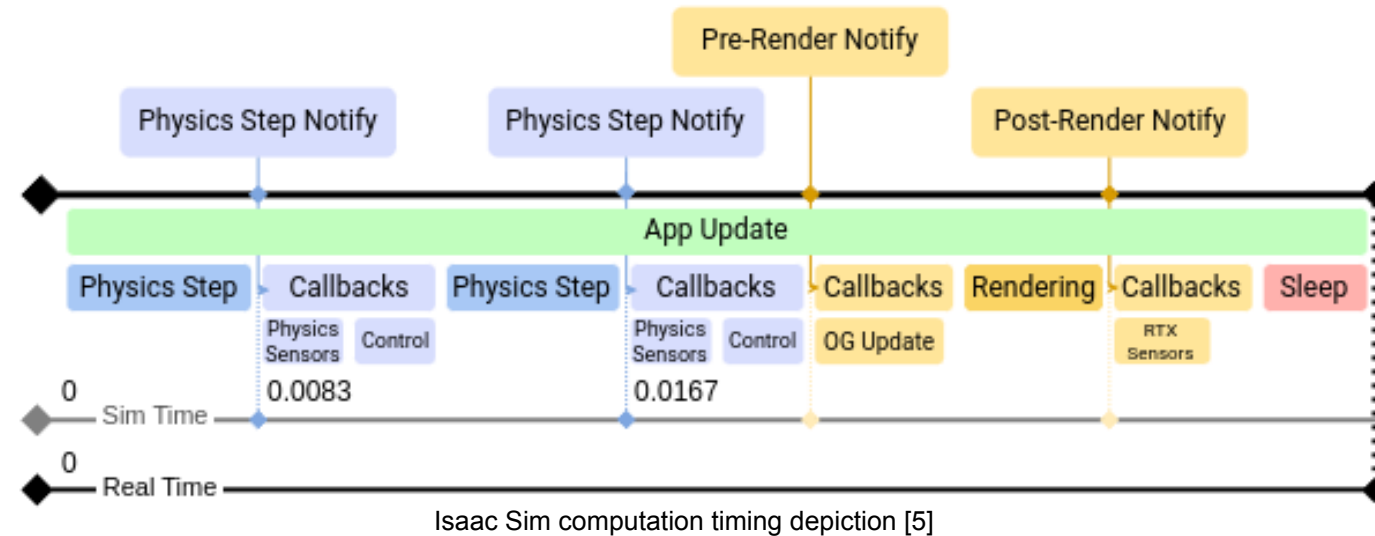
- Introduction
- Problem Definition
- Control Approach
- **Contributions**
- Results
- Future Work
- References



# Event Camera Simulation

## MITLL.SYNTHETIC.EVENTS

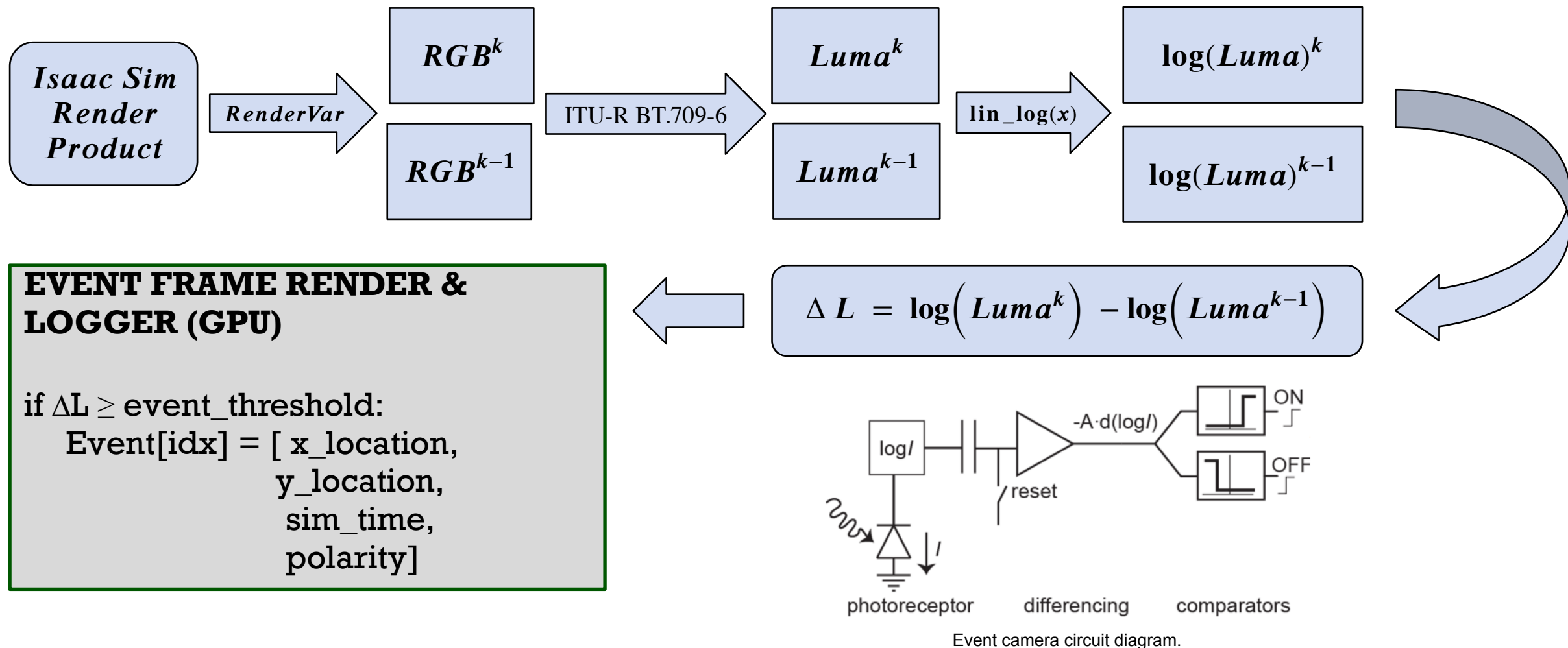
- GPU-Accelerated Event Generation (~5x Faster than CPU, 15 [ms])
- Synthetic Data Writing (HDF5)
- Events to ROS2 “evt3” Encoding
- GUI Displaying Real-Time Events



Event camera simulation visualized through RQt GUI

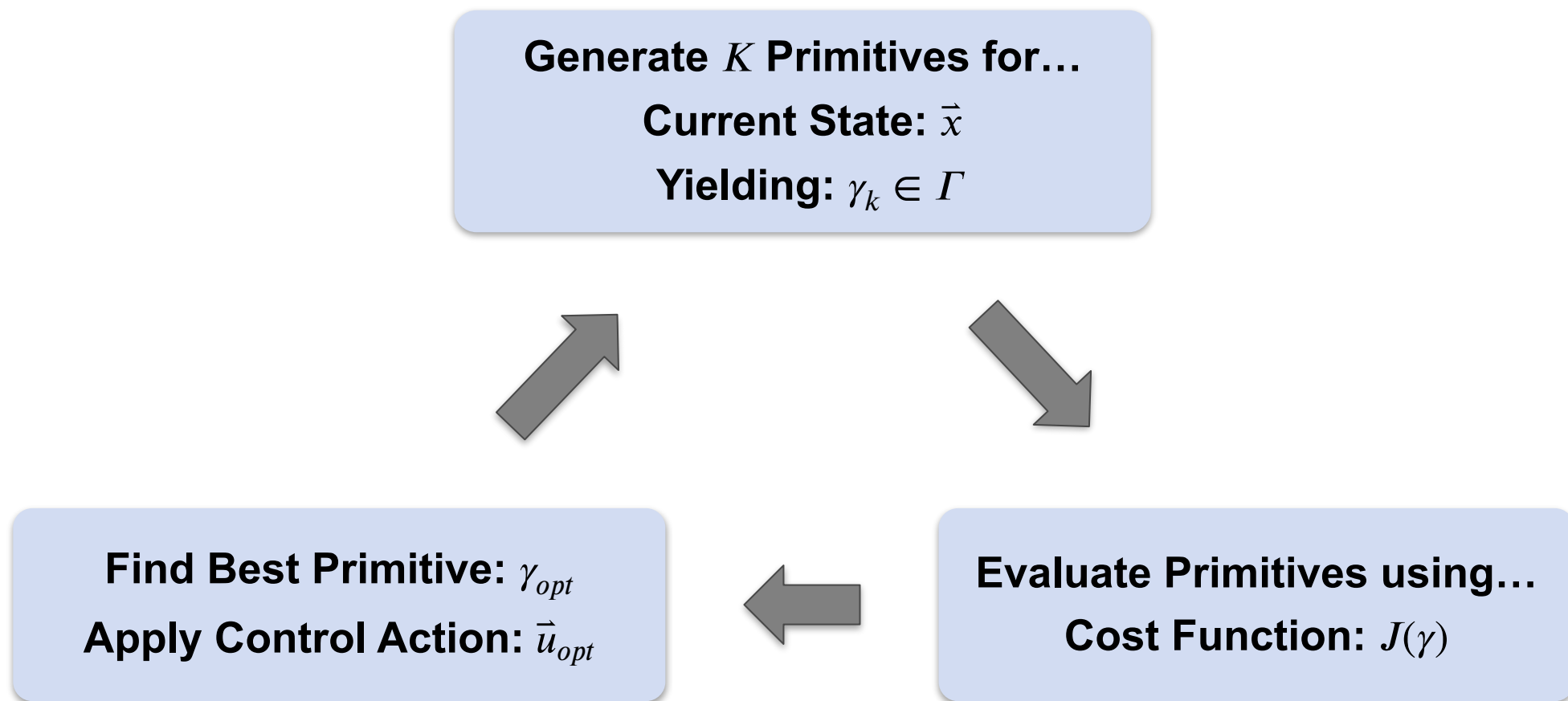


# Event Camera Simulation Cont.





# TTC-Based Motion Primitive Control

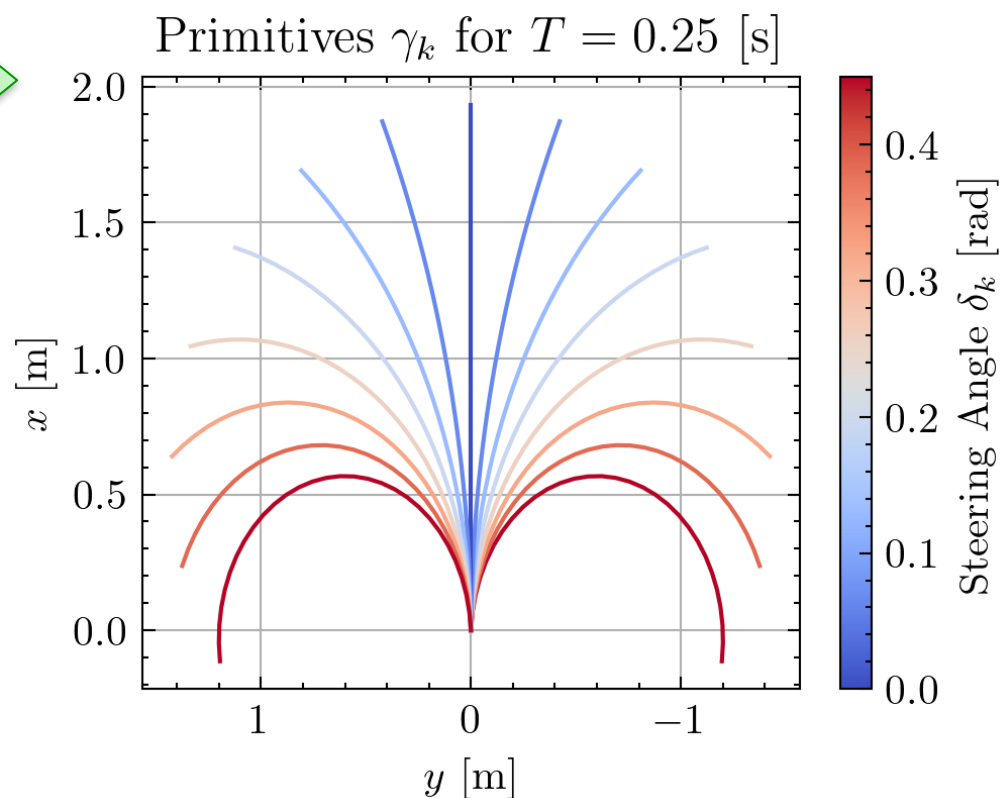
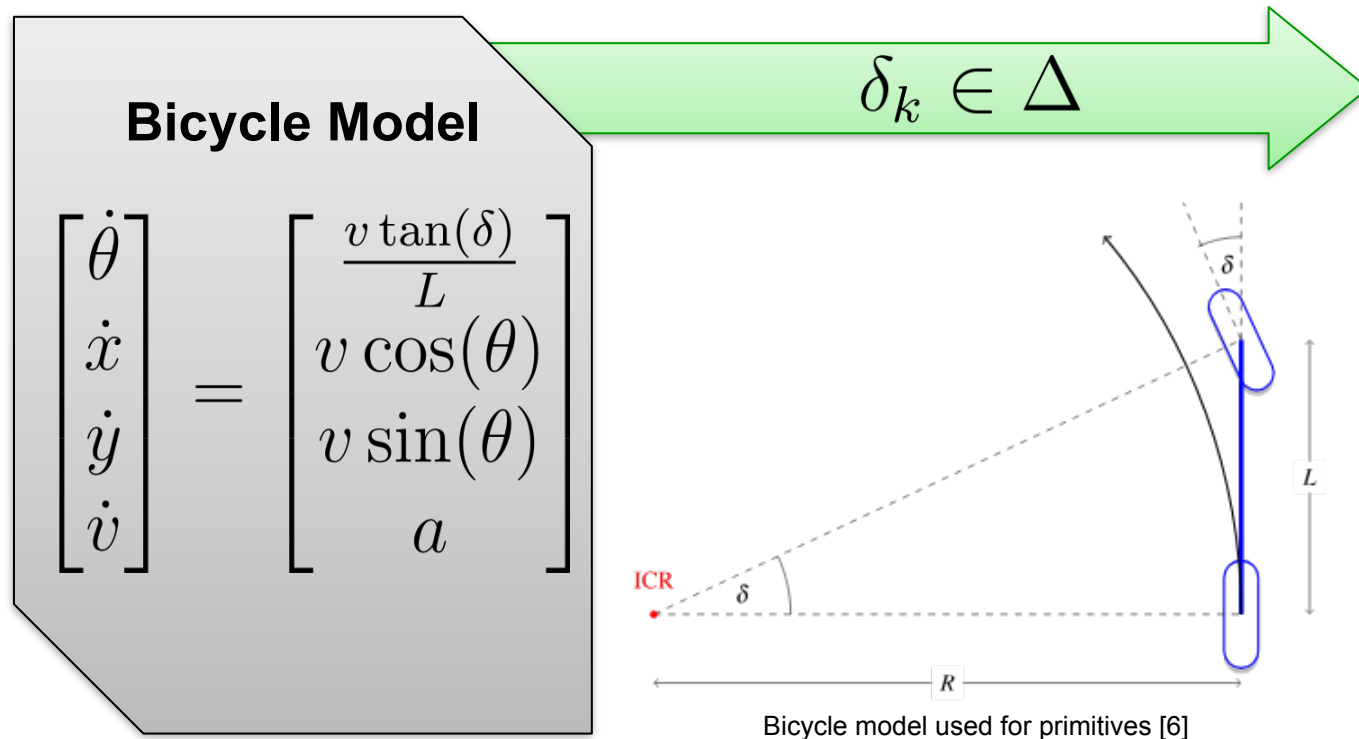




# Primitive Generation

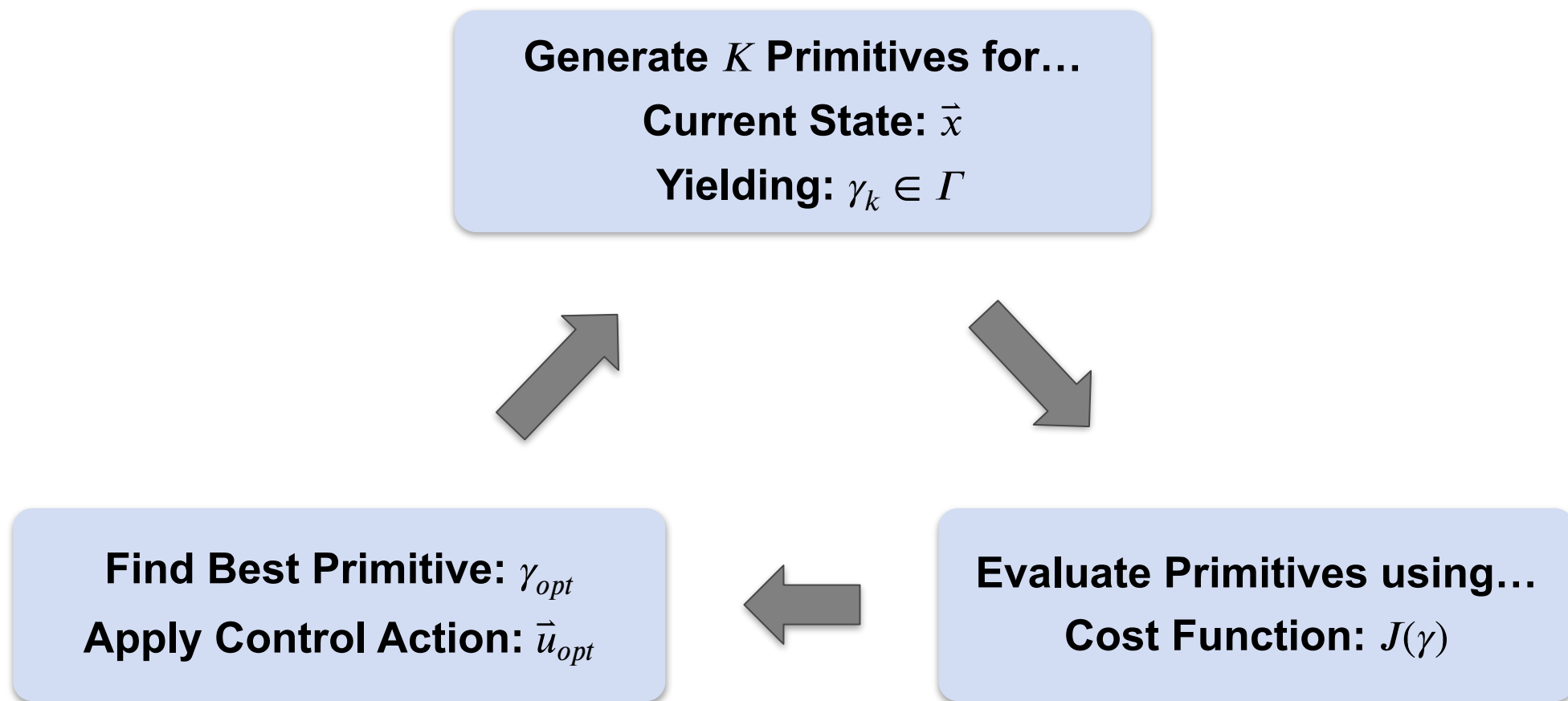
The primitives are generated using...

- **Steering Angle Set:**  $\Delta$
- **Constant Velocity Model:**  $a = 0$





# TTC-Based Motion Primitive Control





# Cost Evaluation

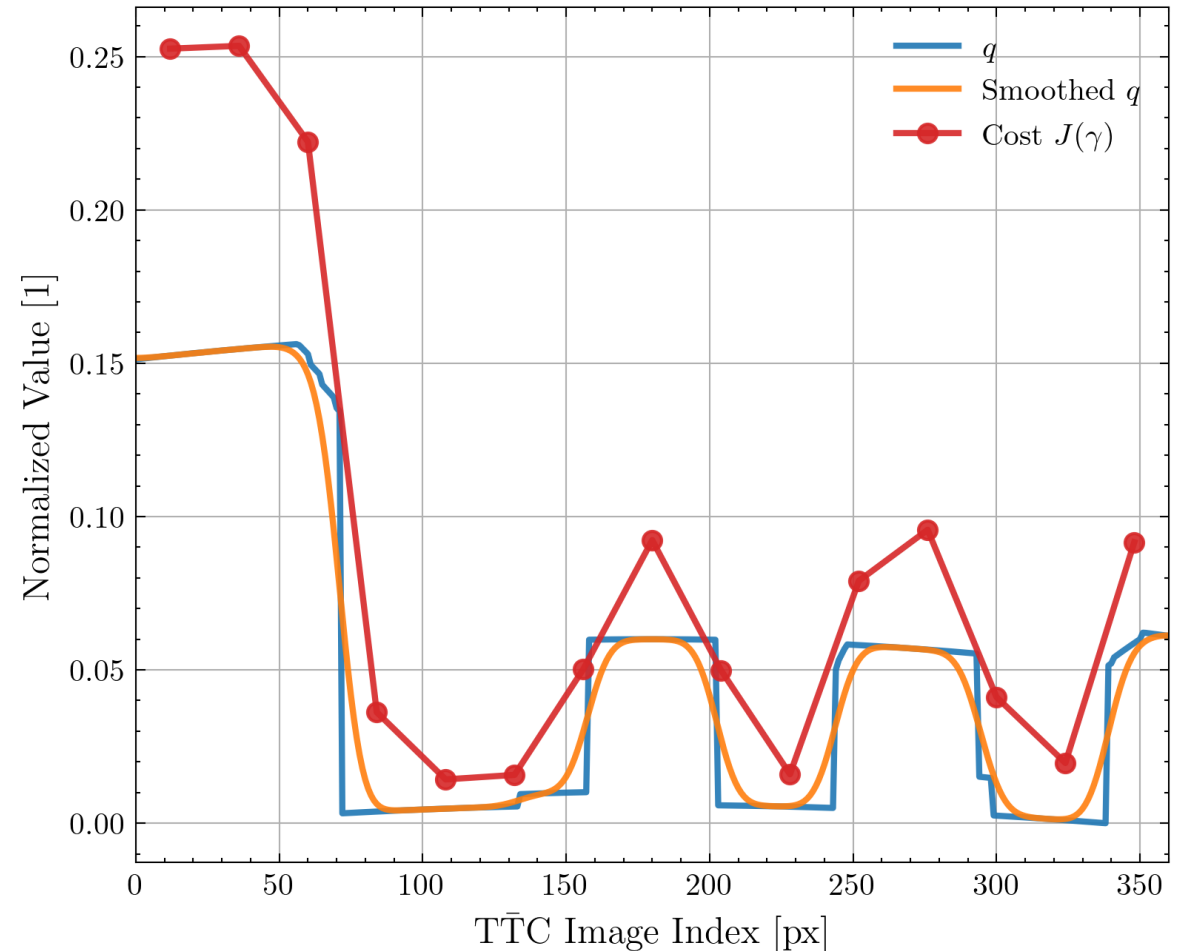
**Inverting and smoothing the vertical mean of the TTC image:**

$$q = \text{GaussianSmooth} \left( \frac{1}{\text{TTC}}, \sigma = 6 \right)$$

**We can evaluate each primitive with the following cost equation:**

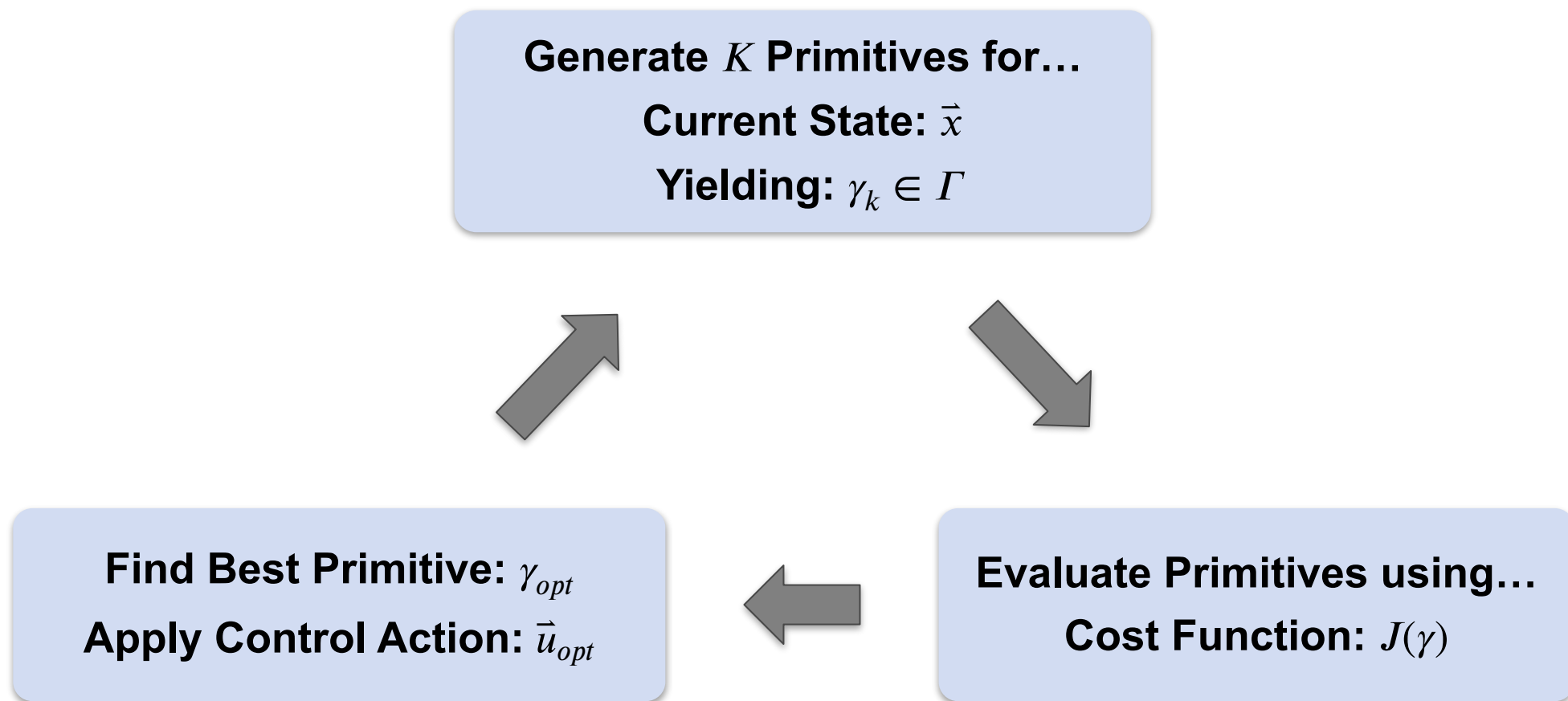
$$J_k(\gamma_k) = \underbrace{\alpha |y_k^{\text{final}}|}_{\text{Deviation from Straight}} + \underbrace{\beta \sum_{i \in \mathcal{I}_k} q_i}_{\text{Obstacle Field}}$$

Obstacle Density and Primitive Cost vs. TTC Image Index





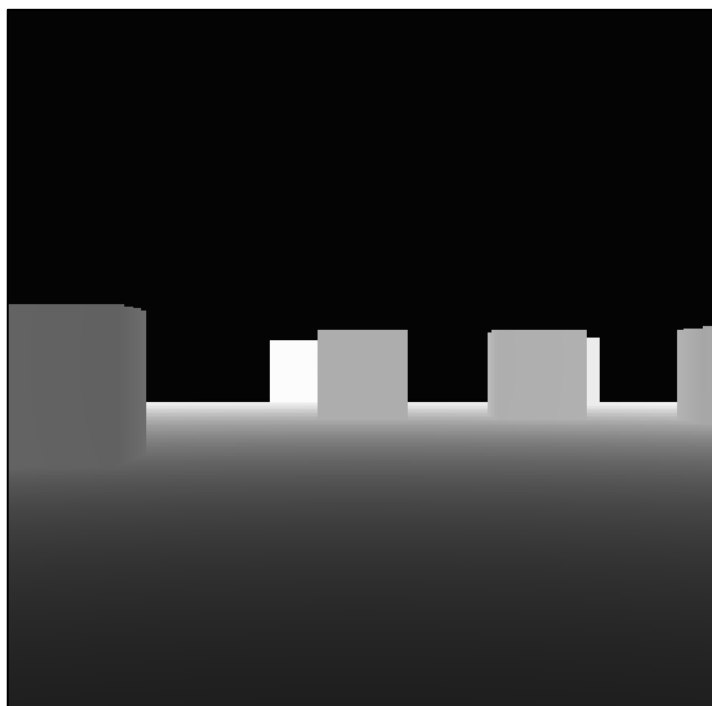
# TTC-Based Motion Primitive Control



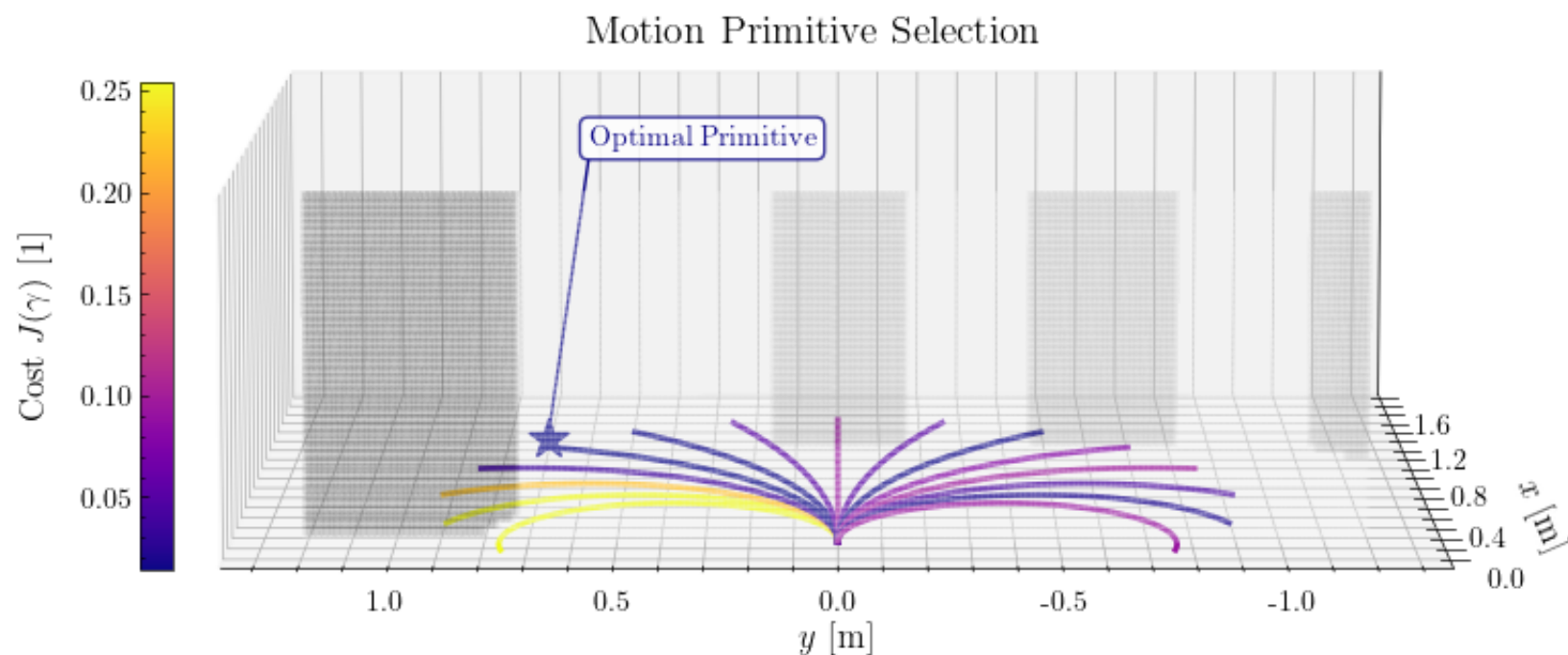


# Primitive Selection

Looping through the set of primitives,  $\Gamma$ , we can find  $\gamma_{opt}$ , and apply associated  $u_{opt} = \delta_{opt}$  to the low-level controller.



Example TTC image from simulation





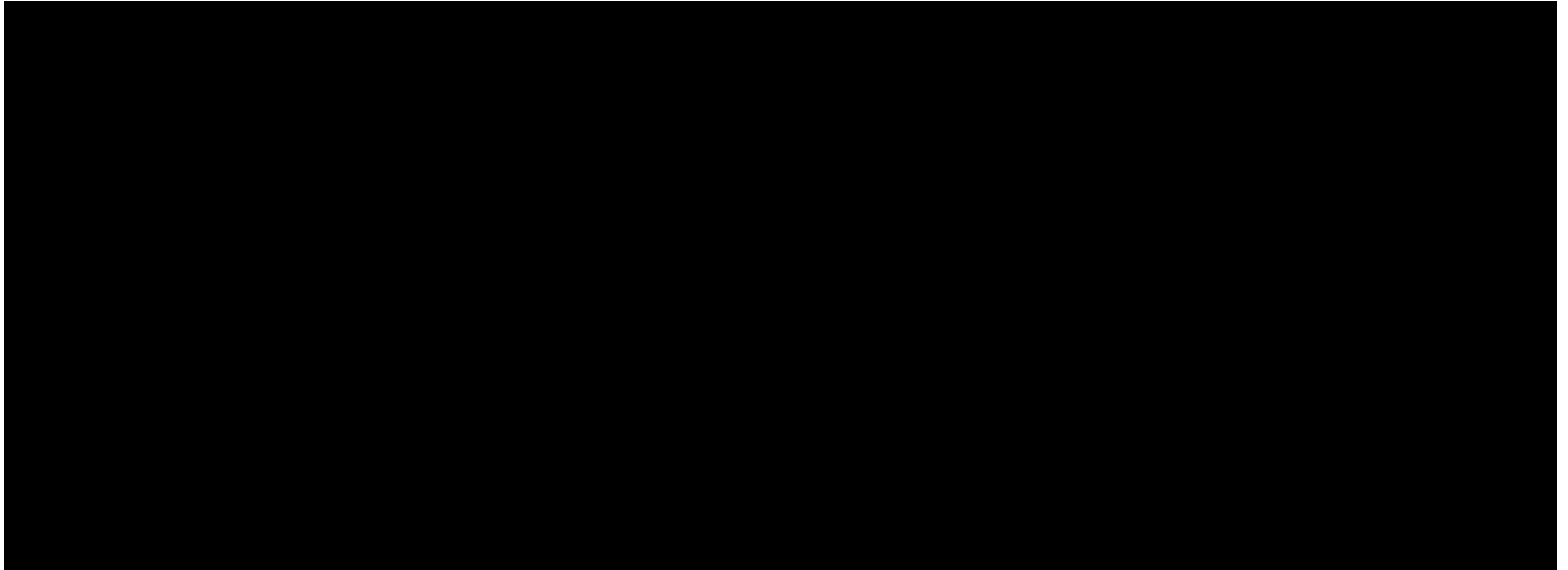
# Outline

---

- Introduction
- Problem Definition
- Control Approach
- Contributions
- **Results**
- Future Work
- References



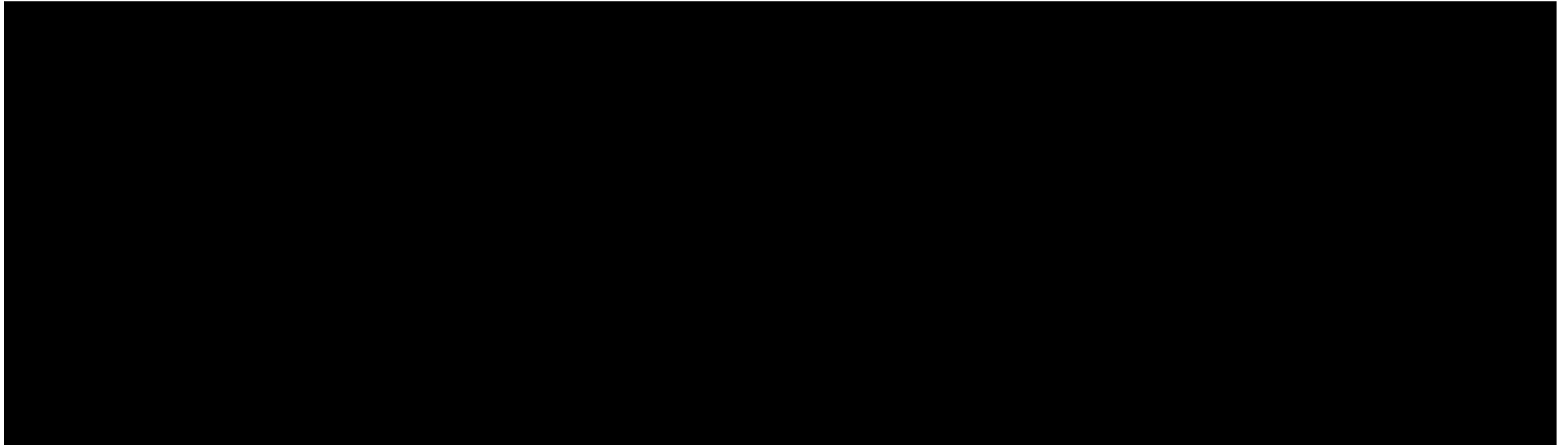
# EV-TTC in Simulation



EV-TTC running on CPU from simulation data



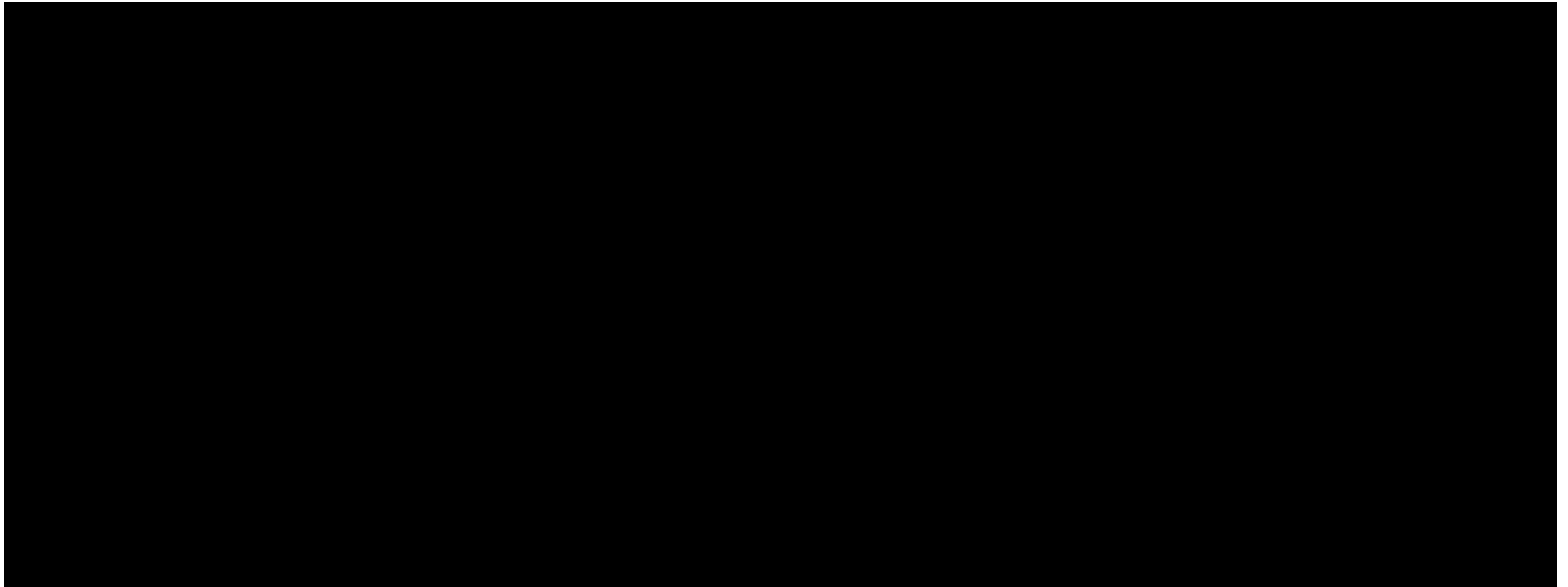
# E2VID in Simulation



E2VID running on GPU from simulation data



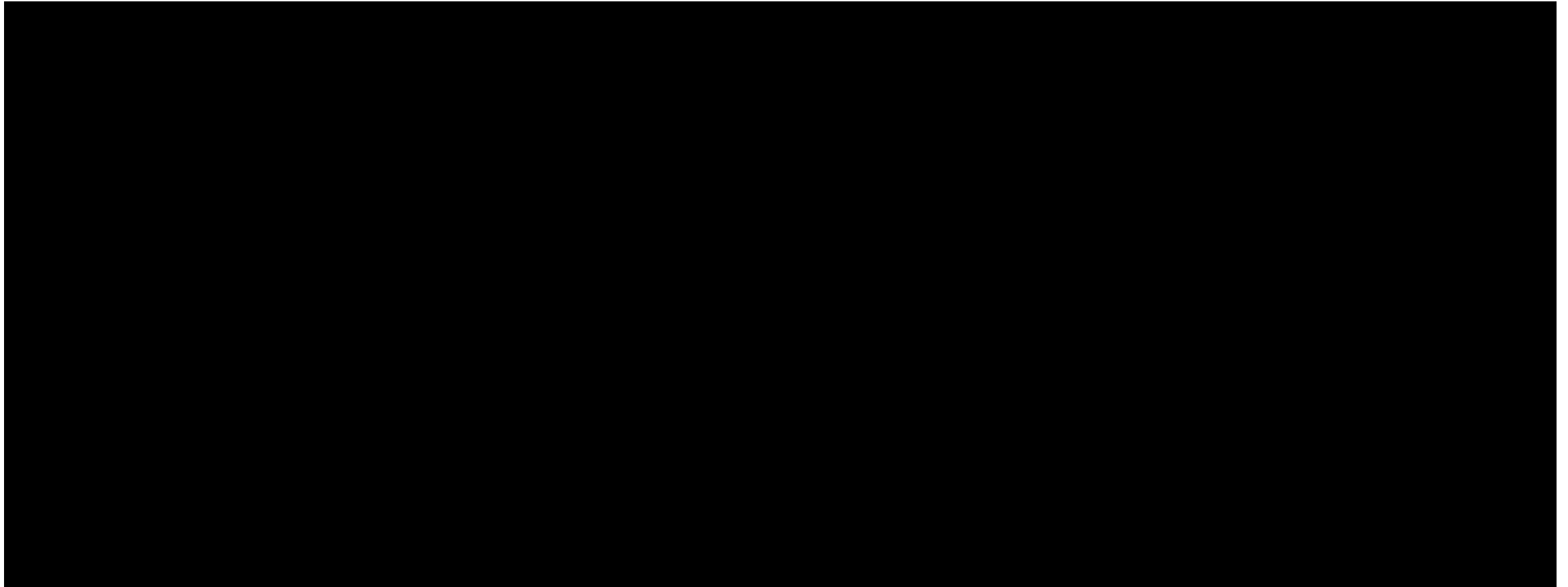
# Results (Bright)



Primitive controller running on ground truth TTC



# Results (Low-Light)



Primitive controller running on ground truth TTC



# Outline

---

- Introduction
- Problem Definition
- Control Approach
- Contributions
- Results
- **Future Work**
- References



# Future Work

## Short-Term:

1. Re-train EV-TTC with M3ED + Synthetic Data
2. Open-Source Isaac Sim Extension
3. Add Noise Modeling to Improve Sim2Real Transfer

## Long-Term:

1. Integrate System Components on Hardware
2. Fast Model of Car & Drone
3. Isaac Lab Event-Based RL
4. Expand Controller to Drone Navigation (TTC Convolution)

## Application:

1. Dense, High-Speed Forest Navigation for Searching



# Outline

---

- Introduction
- Problem Definition
- Control Approach
- Contributions
- Results
- Future Work
- **References**

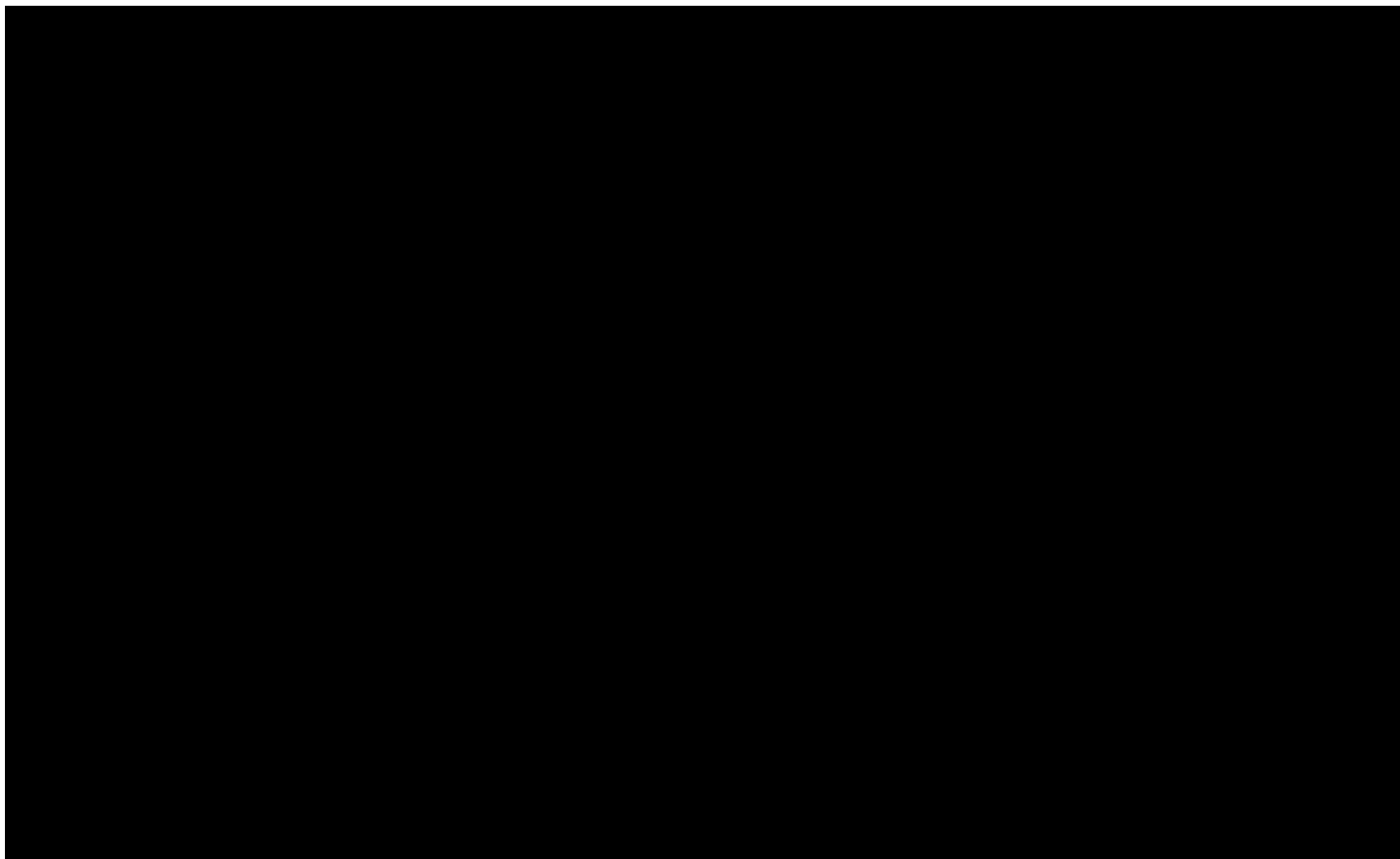


# References

- [1] YouTube, *Event Based Vision Diagram* [Video]. [Online]. Available: <https://www.youtube.com/watch?v=LauQ6LWTkxM&t=25s>
- [2] MUSHR Team, *MUSHR Nano – Noetic Quickstart*. [Online]. Available: [https://mushr.io/tutorials/noetic\\_quickstart/](https://mushr.io/tutorials/noetic_quickstart/)
- [3] FRAMOS, *Metavision Evaluation Kit 4 HD*. [Online]. Available: <https://framos.com/products/sensors/sensor-evaluation-boards/metavision-evaluation-kit-4-hd-26142/>
- [4] A. Bisulco, V. Kumar and K. Daniilidis, "EV-TTC: Event-Based Time to Collision Under Low Light Conditions," in *IEEE Robotics and Automation Letters*, vol. 10, no. 6, pp. 6151-6158, June 2025, doi: 10.1109/LRA.2025.3565150.
- [5] NVIDIA, *Isaac Sim Timeline: Simulation Fundamentals*. [Online]. Available: [https://docs.isaacsim.omniverse.nvidia.com/4.5.0/physics/simulation\\_fundamentals.html#simulation-timeline](https://docs.isaacsim.omniverse.nvidia.com/4.5.0/physics/simulation_fundamentals.html#simulation-timeline)
- [6] T. Fermi, *Bicycle Model*. In *Algorithms for Automated Driving*. [Online]. Available: <https://thomasfermi.github.io/Algorithms-for-Automated-Driving/Control/BicycleModel.html>
- [7] J. Raut, *Latex2Image*. [Online]. Available: <https://latex2image.joeraut.com/>
- [8] J. Lee, A. Rathod, K. Goel, J. Stecklein, and W. Tabib, "Rapid quadrotor navigation in diverse environments using an onboard depth camera," in *Proc. 2024 IEEE Int. Symp. Safety Security and Rescue Robotics (SSRR)*, Nov. 2024, doi: 10.48550/arXiv.2411.04326.

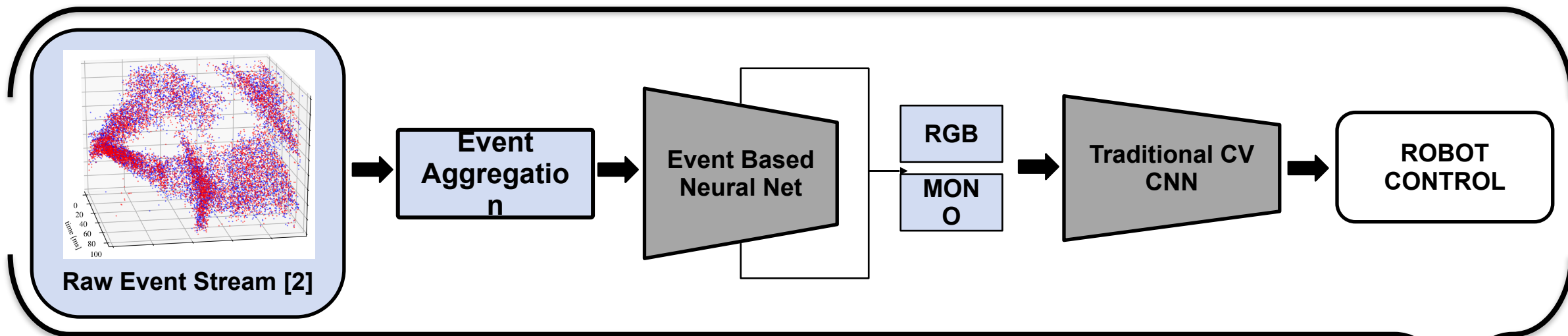


# Thank You! Questions?

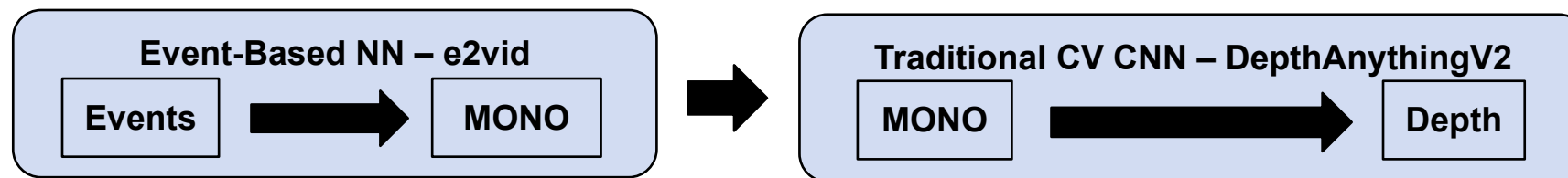




# Initial Event-Based Control Approach



Why not use the open-source options for both NNs?





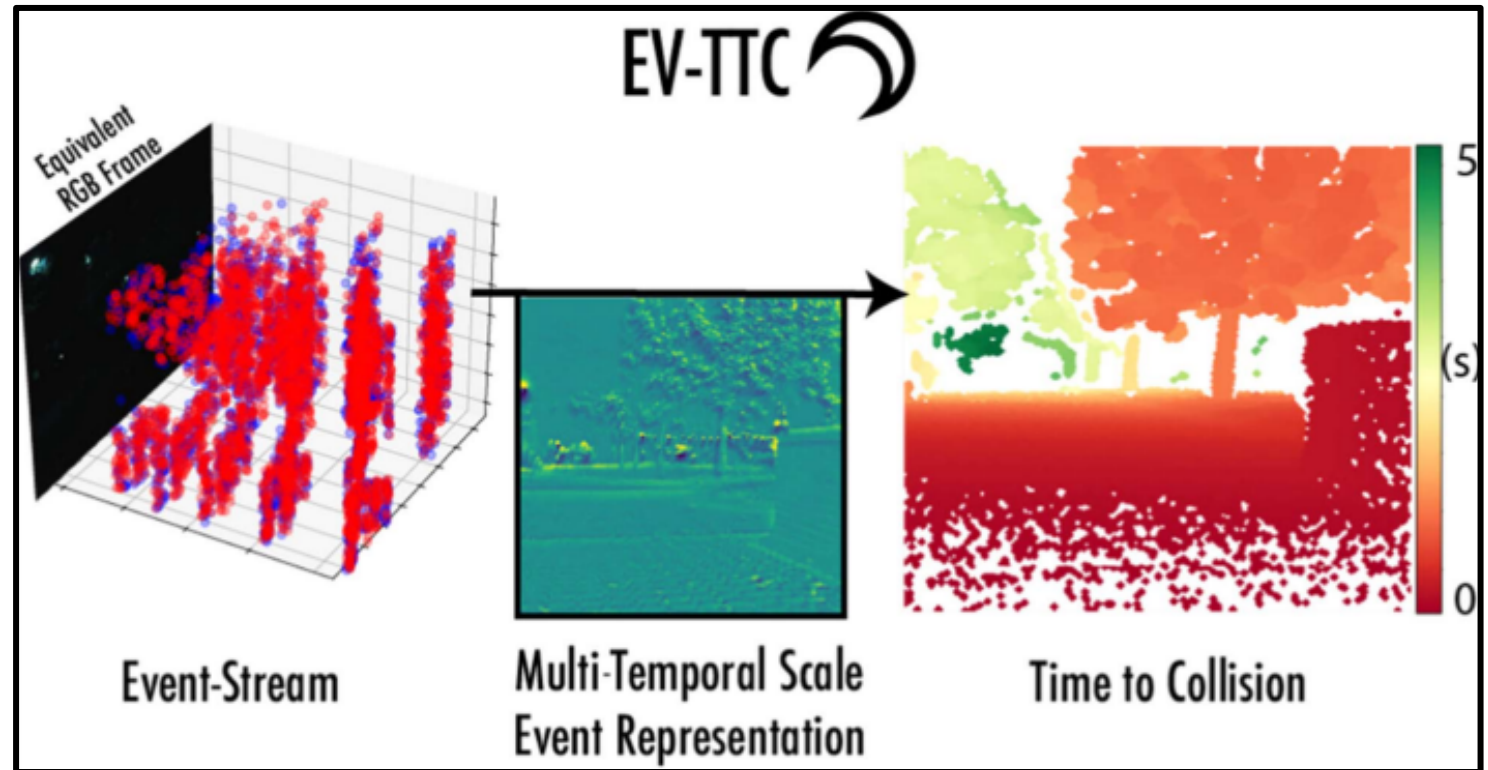
# EV-TTC Summary

## EV-TTC: Event-Based Time to Collision Under Low Light Conditions [3]

- Multi-Temporal Scale Aggregation Module
- Regression NN
- Real-Time Freq: **141 Hz**

### Limitation:

- Trained on a **limited dataset**





# Event-Based Control Approaches (In-Depth)

