MLSys

EarthSight: A Distributed Framework for Low-latency Satellite Intelligence

Ansel Kaplan Erol[1], Seungjun Lee[2], Divya Mahajan[1]

[1]Georgia Institute of Technology, [2]Korean Advanced Institute of Science and Technology

Georgia Tech

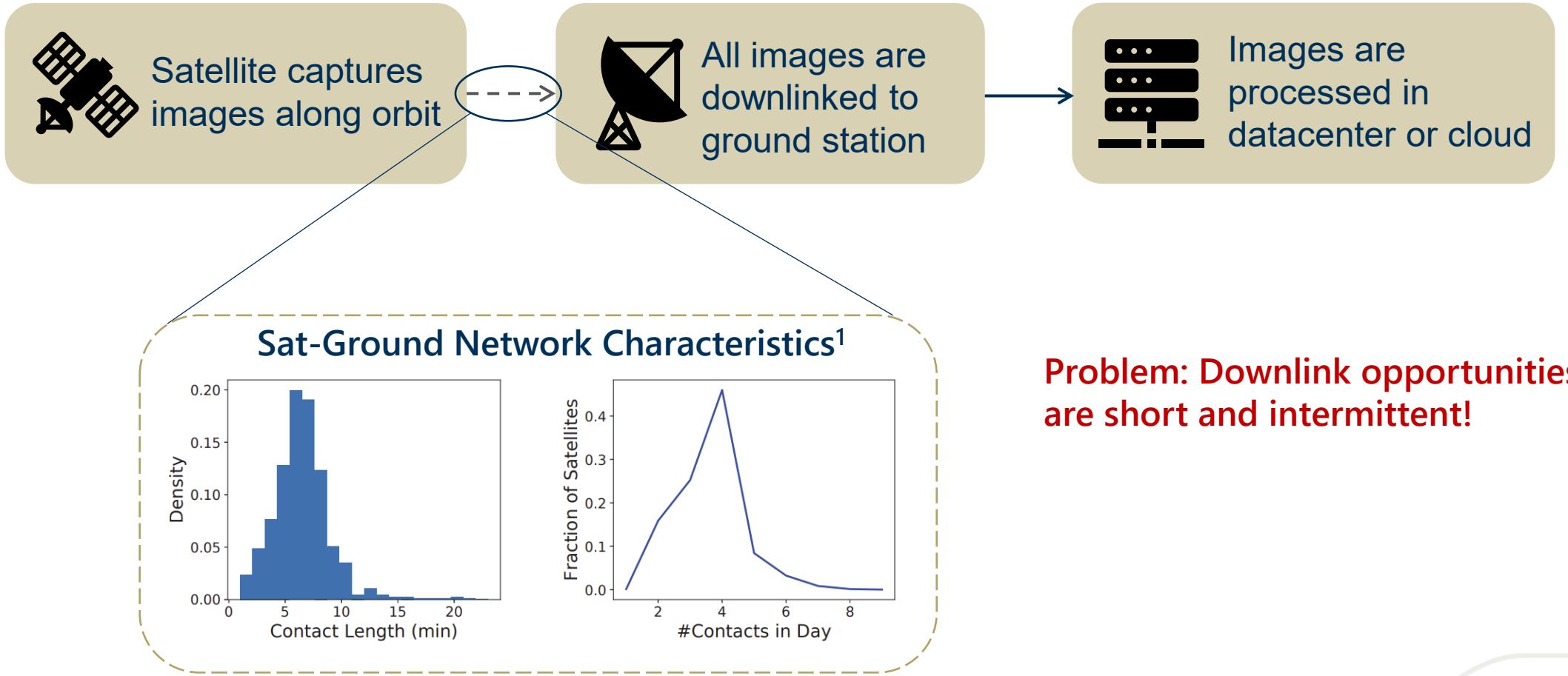
KAIST 1971

SCAI

SYSTEMS FOR SCALABLE INTELLIGENCE

Background: Why Put Compute On Satellites?

- Pre-2020: Bent-pipe



Problem: Downlink opportunities are short and intermittent!

[1]D. Vashisht, J. Shenoy, R. Chandra. *L2D2: Low-latency Distributed Downlink for LEO Satellites* (SIGCOMM '21)

Georgia Tech

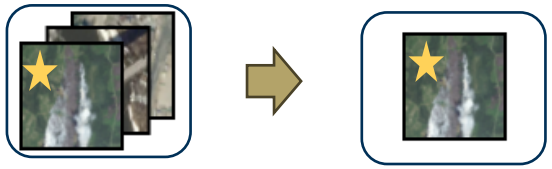
Background: Moving Compute to the Edge

- **Idea: Reduce data volume via onboard compute to alleviate bandwidth pressure**

Applications

Extracting:

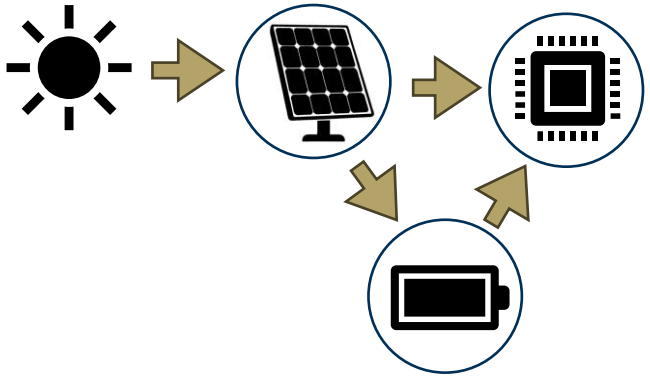
Filtering[1]:



Prioritizing[2]:

Power Limits

All available power comes from the sun



5-30W power available for inference

AI Coprocessors

Must be small, efficient, and radiation-resilient

Google Coral Edge TPU

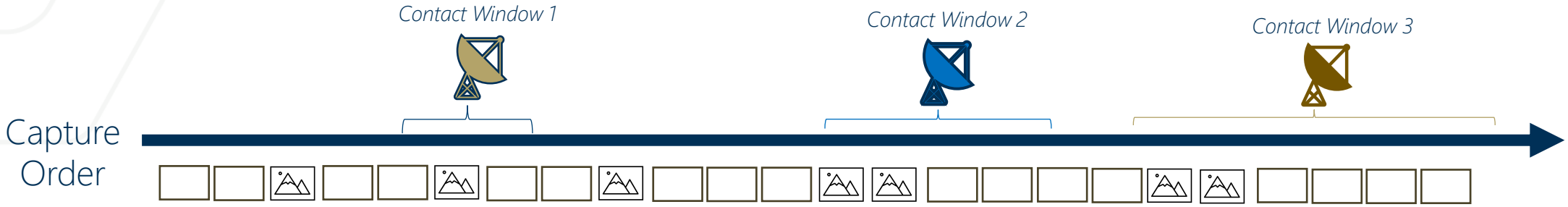
NVIDIA Jetson Series GPU

Initiatives are underway to deploy these in large satellite fleets!

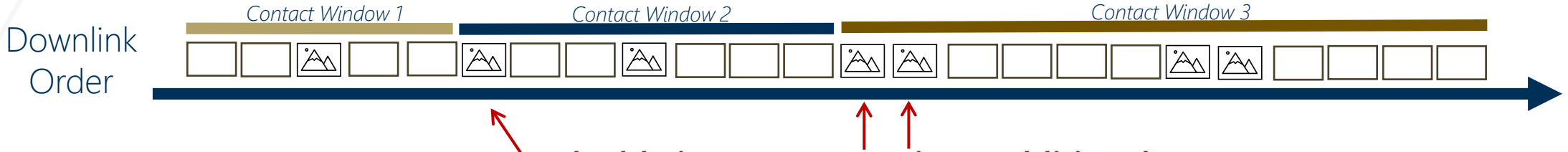
[1]B. Denby, B. Lucia. Orbital Edge Computing: Nanosatellite Constellations as a new class of Computing Systems (ASPLOS '20 Best Paper)
[2]B. Tao, O. Chabra, I. Janveja, I. Gupta, D. Vashisht. Known Knowns and Unknowns: Near Realtime Earth Observation with Serval. (NSDI '24)

Georgia Tech

Latency Challenges for Earth Observation

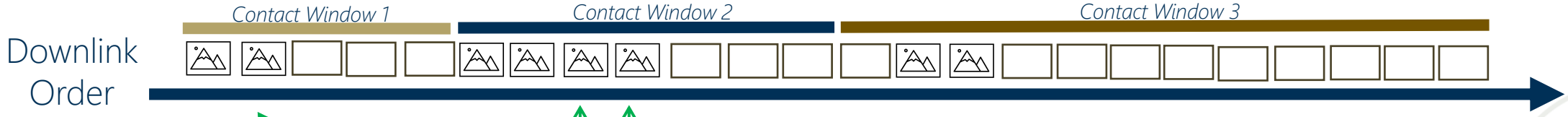


FIFO: Images are downlinked in order of capture



Valuable image must wait an additional contact window before downlink → Latency

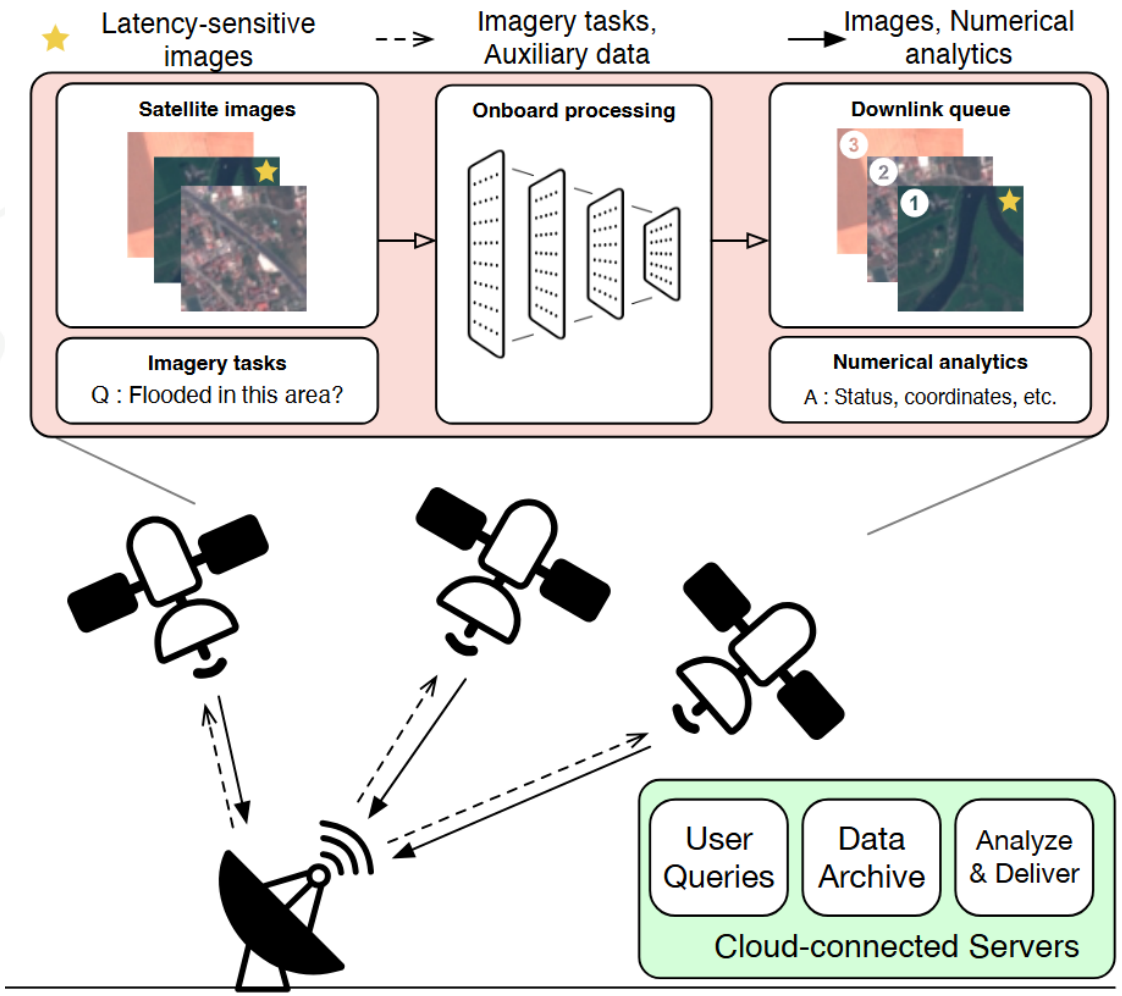
Priority: More useful images are downlinked first



Priority images are downlinked at the first opportunity!

Georgia Tech

Priority Scheme for Low-latency Earth Observation

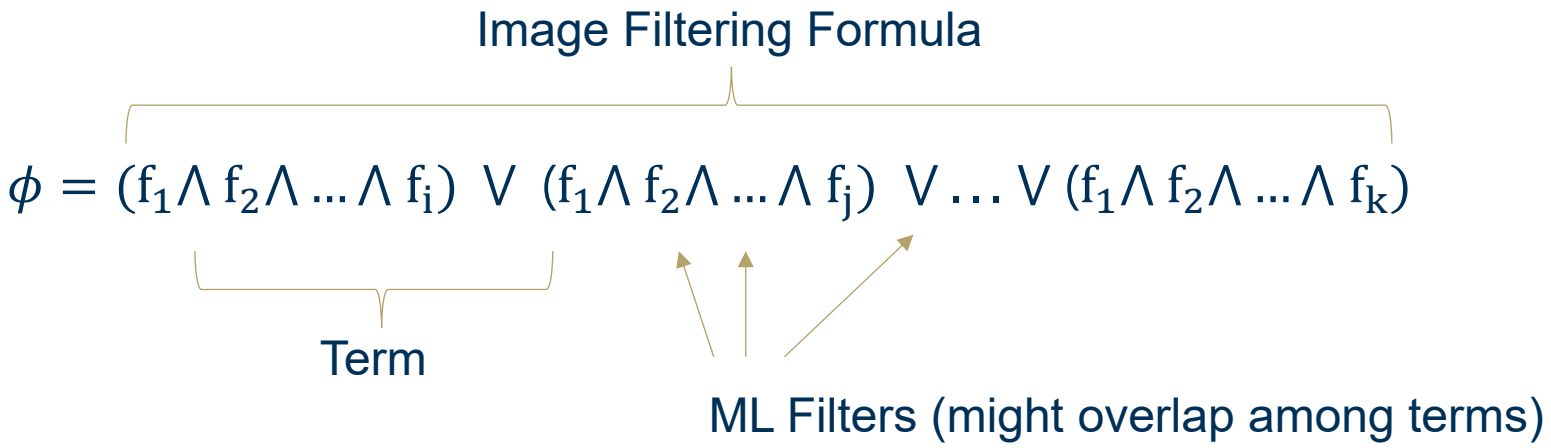


- Images are collected by onboard camera sensors
- ML Vision Inference is performed to detect high-value images, those satisfying user queries.
  - **Example**: (tree-covered AND smoke is present) OR (Visible Flames) OR (trucks present AND logging)
  - Each criteria is a Boolean filter that is set through computer vision inference
- Images are downlinked in order of value, minimizing latency for high-value images.

Georgia Tech

More on Queries

- ML Inference is used to detect high-value images satisfying user queries
- User queries are synthesized to form a Boolean formula for each image, with each filter representing a model.

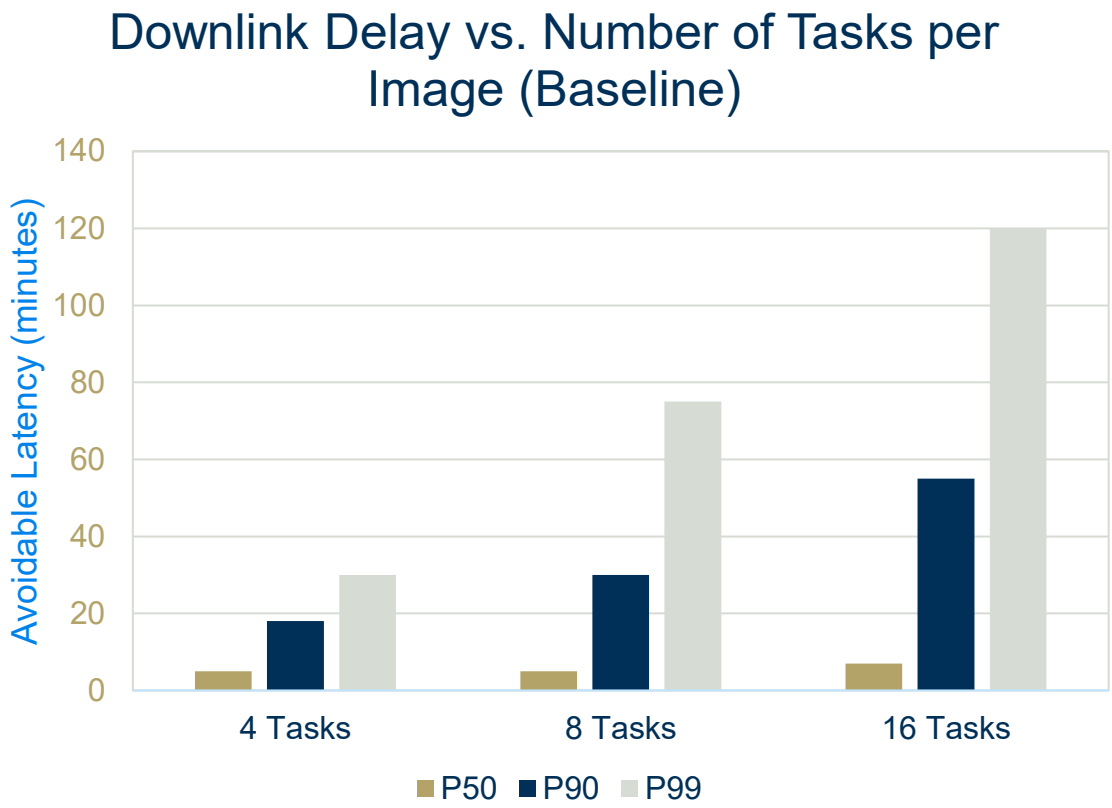


- Goal is to evaluate ϕ as cheaply as possible across as many images as possible
- The complexity of this evaluation scales with the number of tasks (which contributes to the number of terms) or the complexity of each task (which contributes to the number of filters per term)

Georgia Tech

Challenge: Scaling up the Number of Tasks

- Commercial and scale systems must be able to support many tasks per image to be feasible
- However, as the number of inference tasks per grows, there are insufficient resources to classify every image before the ground contact, resulting in excessive tail latency
- **Power** and image processing **throughput** can both be bottlenecks



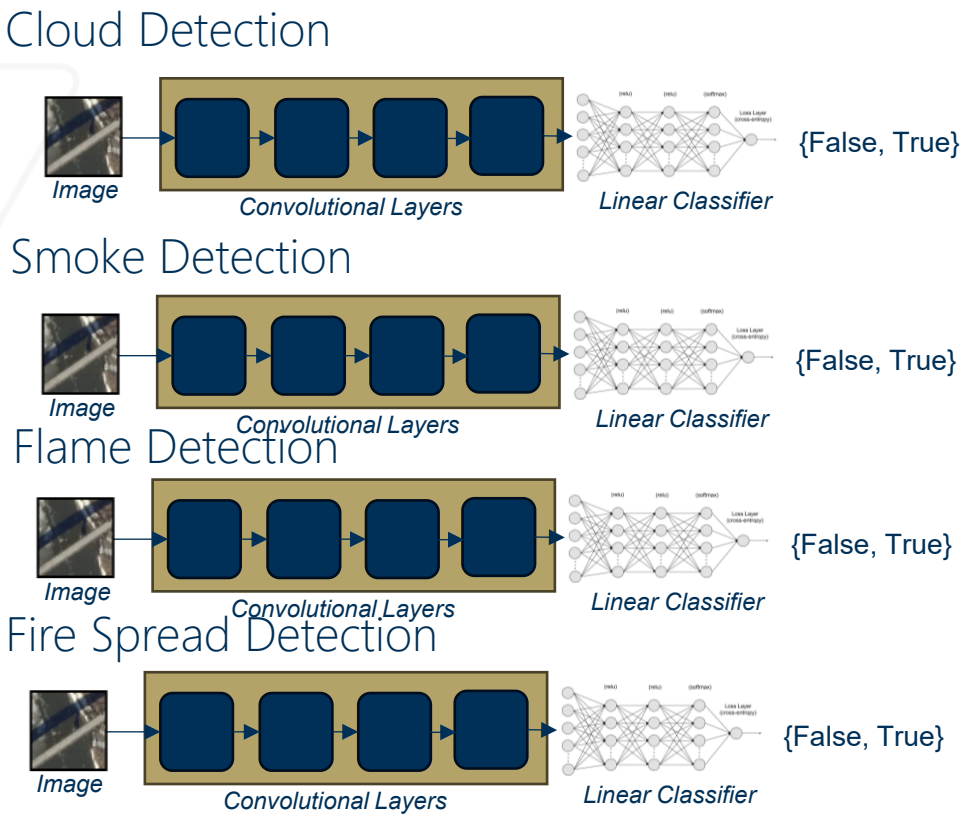
EarthSight's goal: Mitigate image processing bottlenecks to maximize the fraction of classified images and minimize latency

Georgia Tech

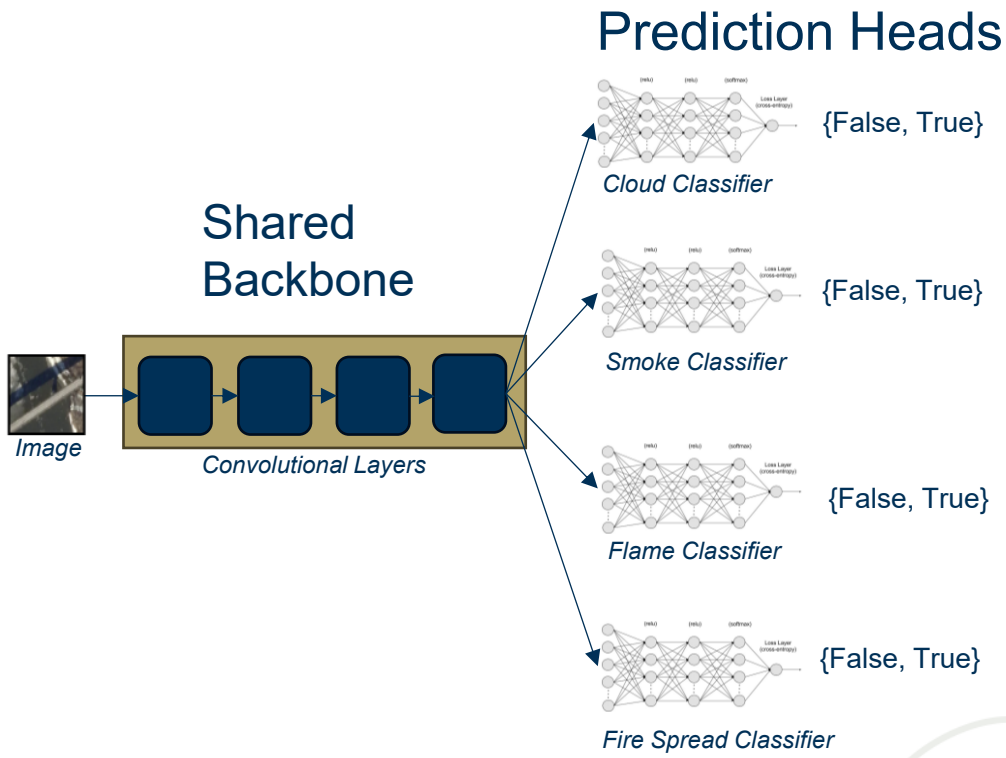
Minimizing Redundant Compute

- **Key Idea #1: Multi-task models**

Prior Work: One Filter = One Model



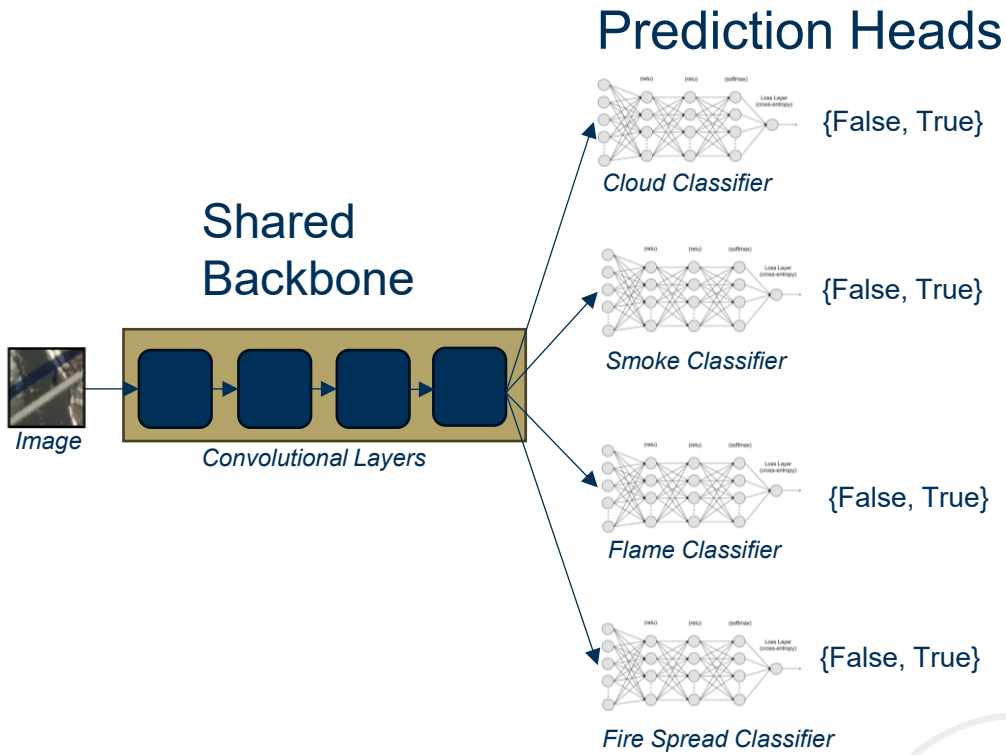
EarthSight: Overlap Backbones



Minimizing Redundant Compute

- **Key Idea #1: Multi-task models**
  - To preserve accuracy, filters with similar features are clustered together into a backbone
  - At inference, only the backbones and classifiers related to the image are executed
  - Additional scheduling constraint: backbones before heads!

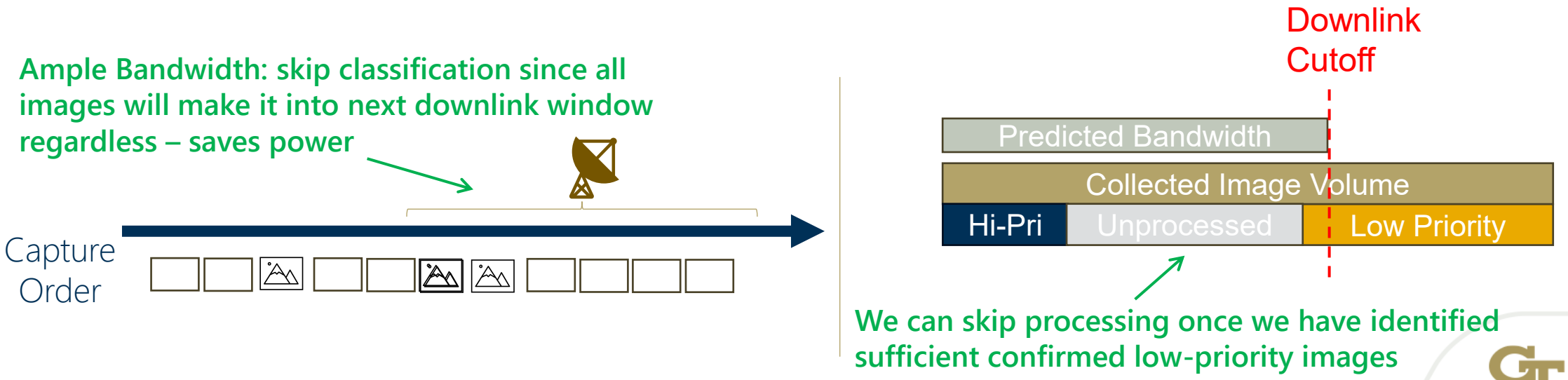
EarthSight: Overlap Backbones



Georgia Tech

Global Context

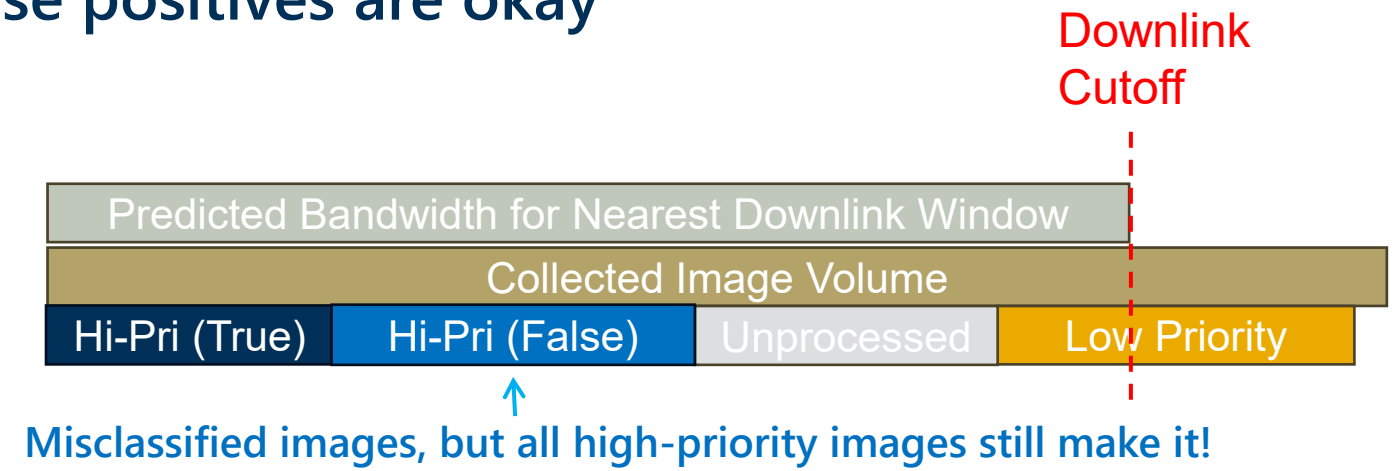
- **Key Idea #2: Integrate global context**
  - We can predict, in advance, what downlink bandwidth will be available for each satellite
  - This enables us to prune computation that will not affect downlink
  - Lookahead simulation with global state is required to determine bandwidth contention, which is impacted by all satellites in the fleet



Georgia Tech

Local Adaptation

- Insight: False positives are okay



- The system can sustain high-priority annotations of some low-priority ground-truth images (up to 2-3x typically) without significant latency impacts.
- We can skip inference of some filters for each image if our confidence is sufficiently high.
  - We assume that we have empirical forecasts of how likely each filter is to return the desired outcome; after that, calculating confidence is simple probability!

Georgia Tech

Local Adaptation

- **Key Idea #3: Local adaptation via dynamic confidence thresholds**
  - Tolerance to false positives should adapt to should adapt to real execution state.

Confidence Required to Accept an Image

Power \ Bandwidth	available	scarce
high	*moderate*	*high*
low	*low*	*best-effort*

- For each image, execute filters until:
  - Confidence = 0: This means for every "term," at least one filter returned false
  - Confidence > α, where $\alpha_t = \min(1, \alpha_{t-1} + \lambda_1(r_{\text{power},t} - 1) + \lambda_2(r_{\text{dep},t} - r_{\text{reject}}))$

Georgia Tech

Reorganizing Computation

- **Key Idea #4: Optimize filter execution order**
  - How do we decide which filter to execute next to minimize the execution time until confidence is equal to 0 or at least α?
  - This problem, *"Stochastic Boolean Function Evaluation,"* is NP-hard
  - Solution – greedy selection of filter with highest decisiveness/cost

Probability the filter returns False

True-positive rate

Number of terms filter could invalidate

$$U_\phi(f_i, \mathcal{E}) = \frac{(1 - p_i) \cdot \mathrm{tpr}_i \cdot n_i}{t_{\mathrm{eff}}(f_i, \mathcal{E})}$$

Utility of executing the i'th filter given current execution state

Execution time of the i'th filter, considering backbone if it has not been executed already

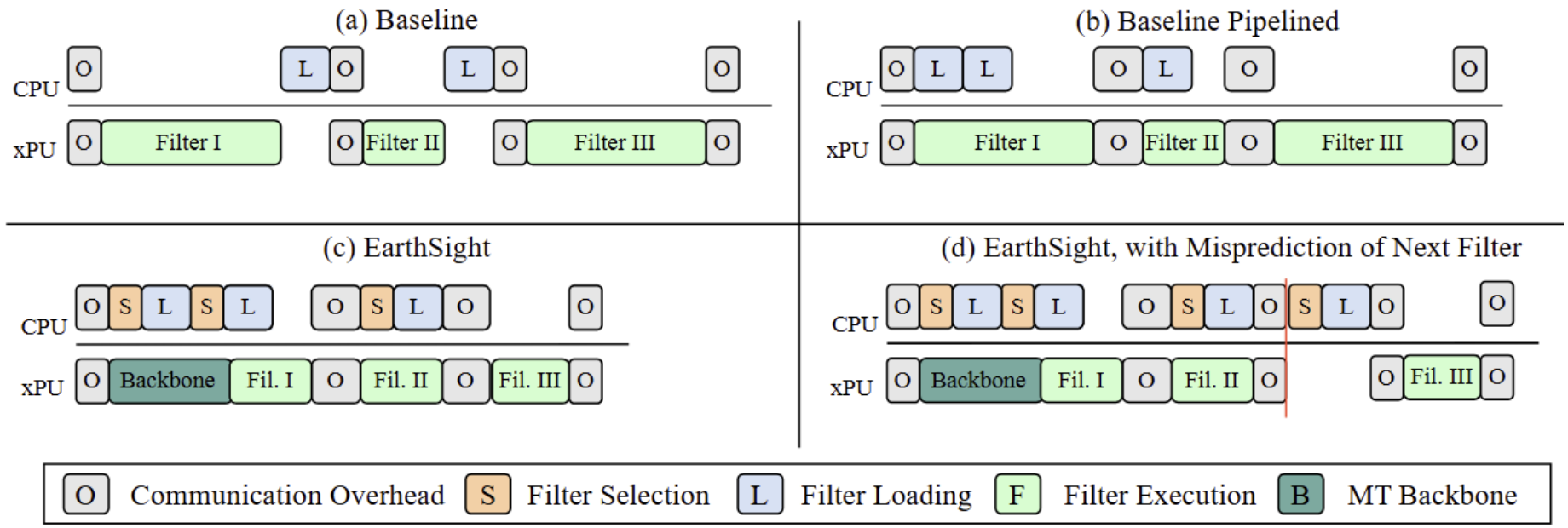
Decisiveness: contribution of the filter to early-reject of the image *(anomalies are rare)*

Cost: Execution Time *(∝ Power)*

Georgia Tech

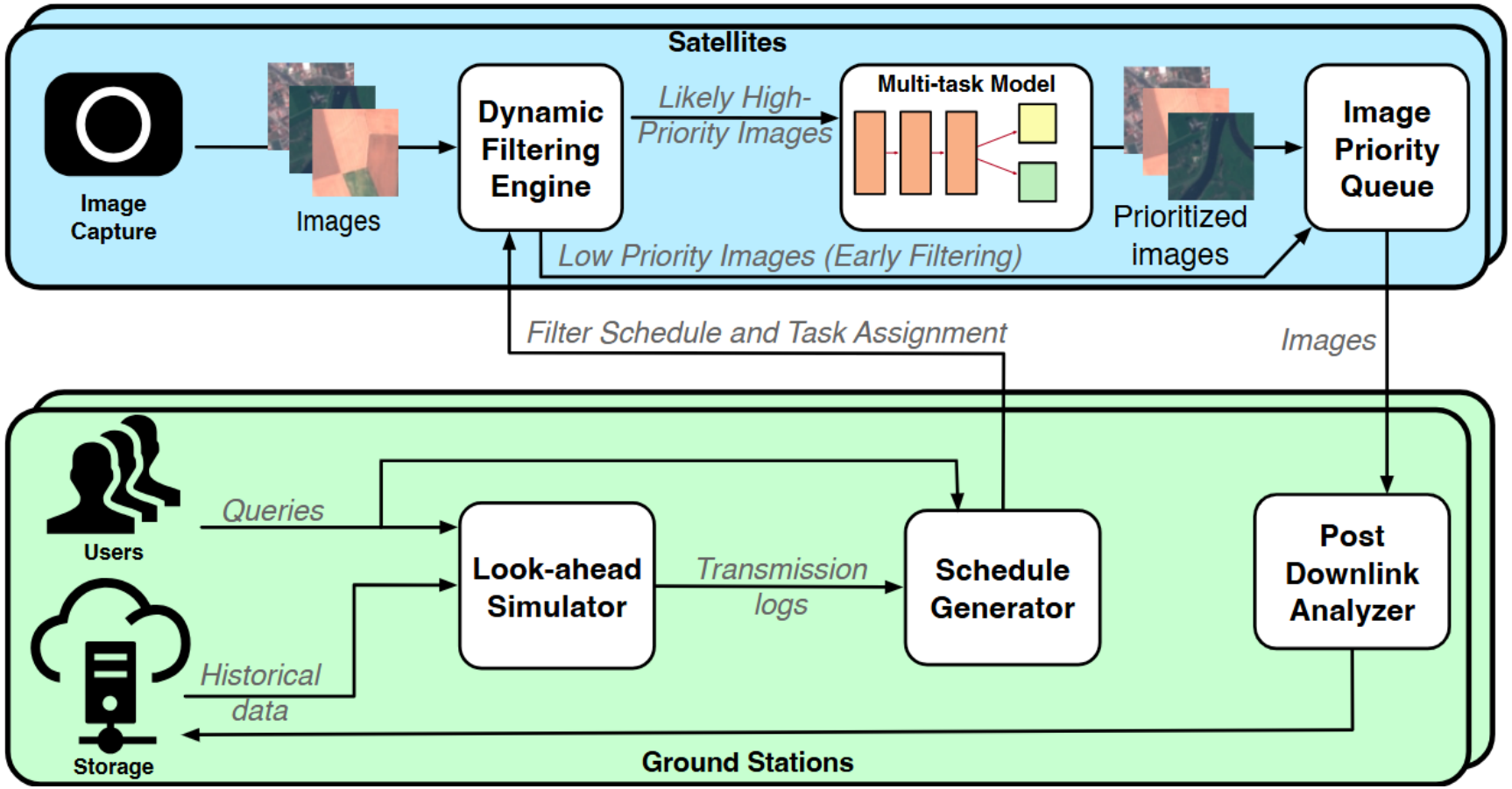
Reorganizing Computation

- **Overlap model selection with model execution**
  - Filter selection is O(n) – somewhat expensive
  - While filter *f* executes, calculate the optimal next filter under the assumption the image fails *f* (and if time, passes) and pre-load the weights



Putting it All Together

EarthSight combines global context with local adaptation and efficient filter ordering to achieve low-latency satellite intelligence.

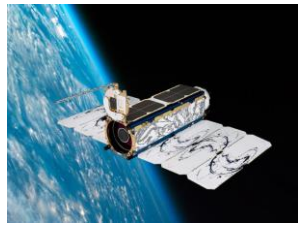


Georgia Tech

Evaluation Setup

EarthSight is evaluated via simulation over 153 satellites using real hardware traces on *Jetson Orin* and *Coral Edge TPU* accelerators

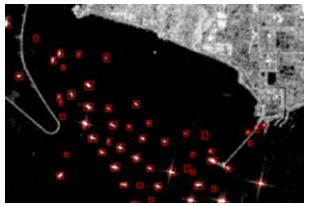
Constellation Configuration



Planet Dove
Fleet Orbits,
Ground Station
Locations

Task Scenarios

Natural Disasters



Military Intelligence

Urban Monitoring

AI Co-processors

Two Alternatives

Google Coral Edge TPU

2W
4TOPS

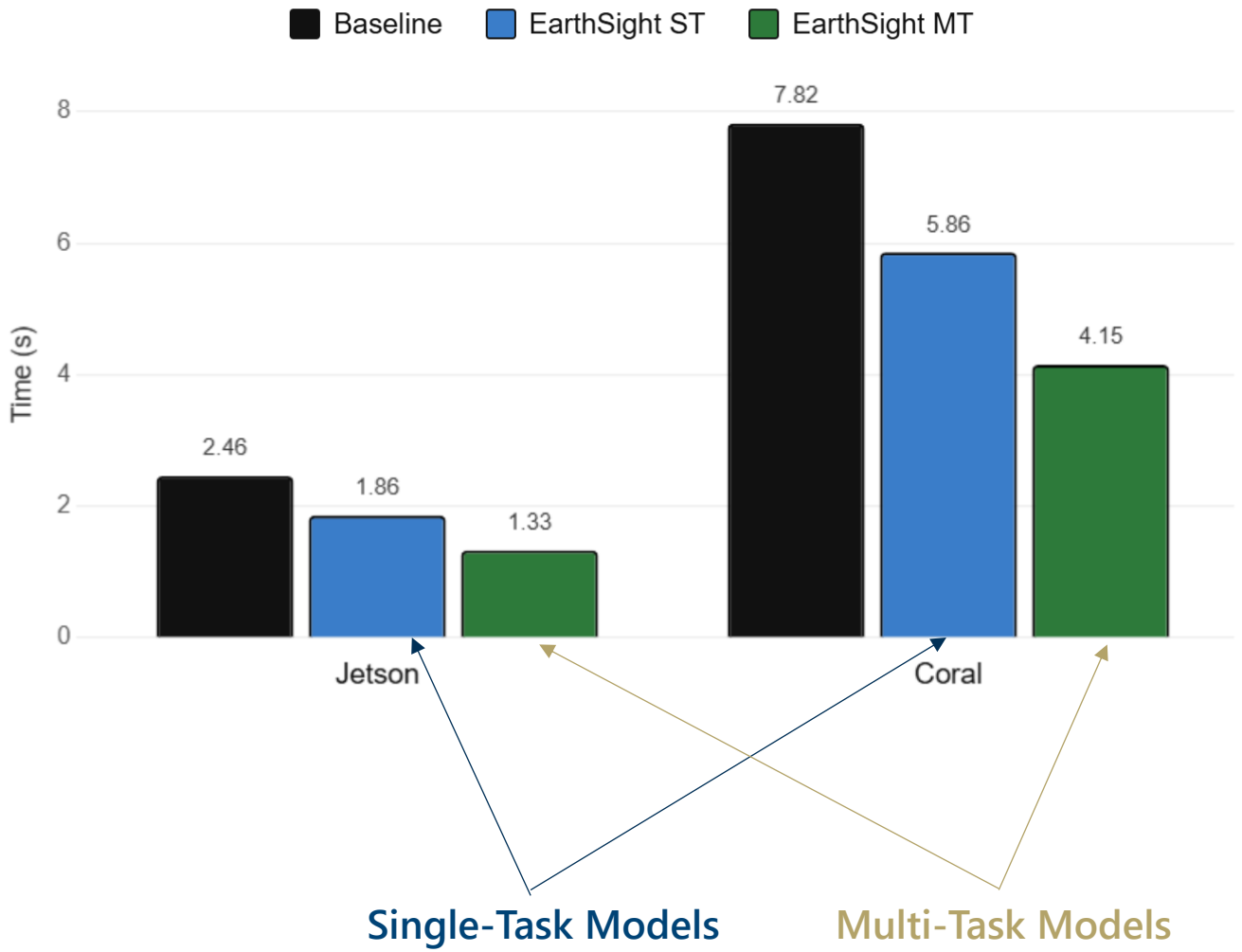
NVIDIA Jetson Orin Nano

15-30W
67 TOPS

Initiatives are underway to deploy these in large satellite fleets!

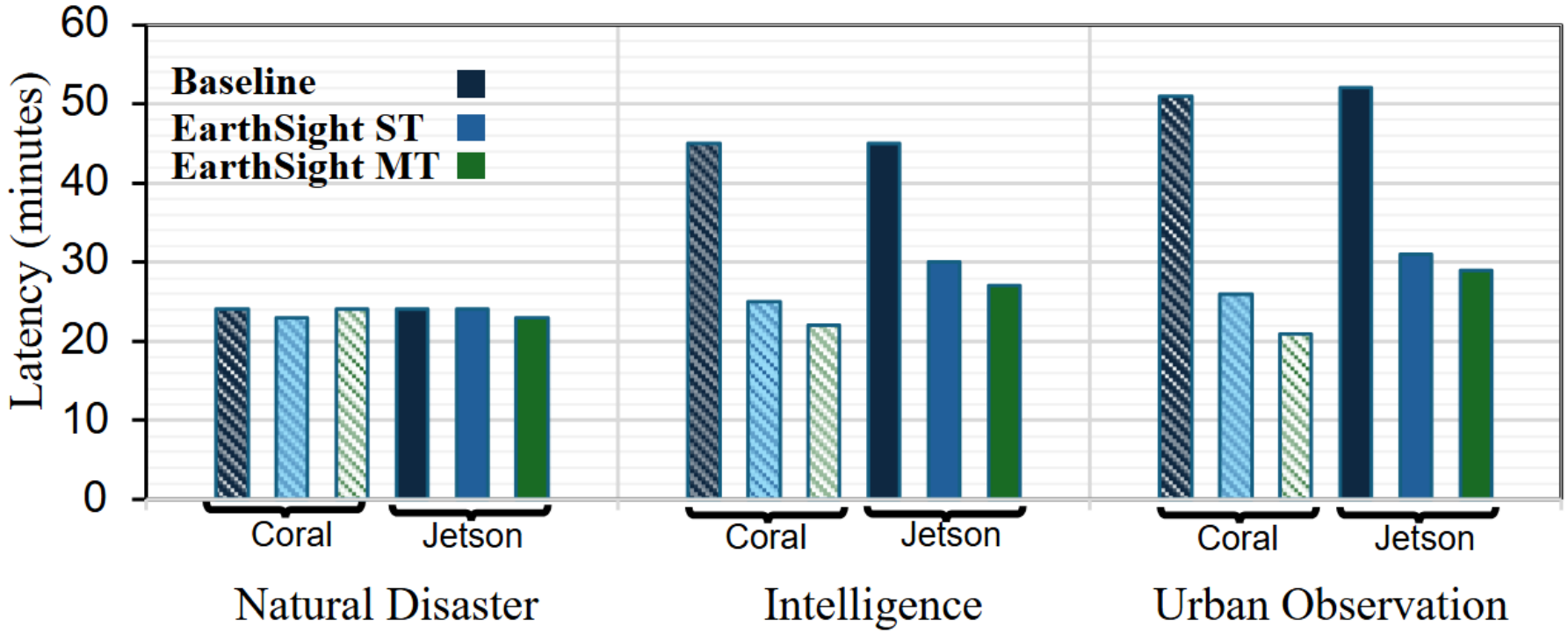
Georgia Tech

EarthSight reduces the average time to annotate each image by 1.9x



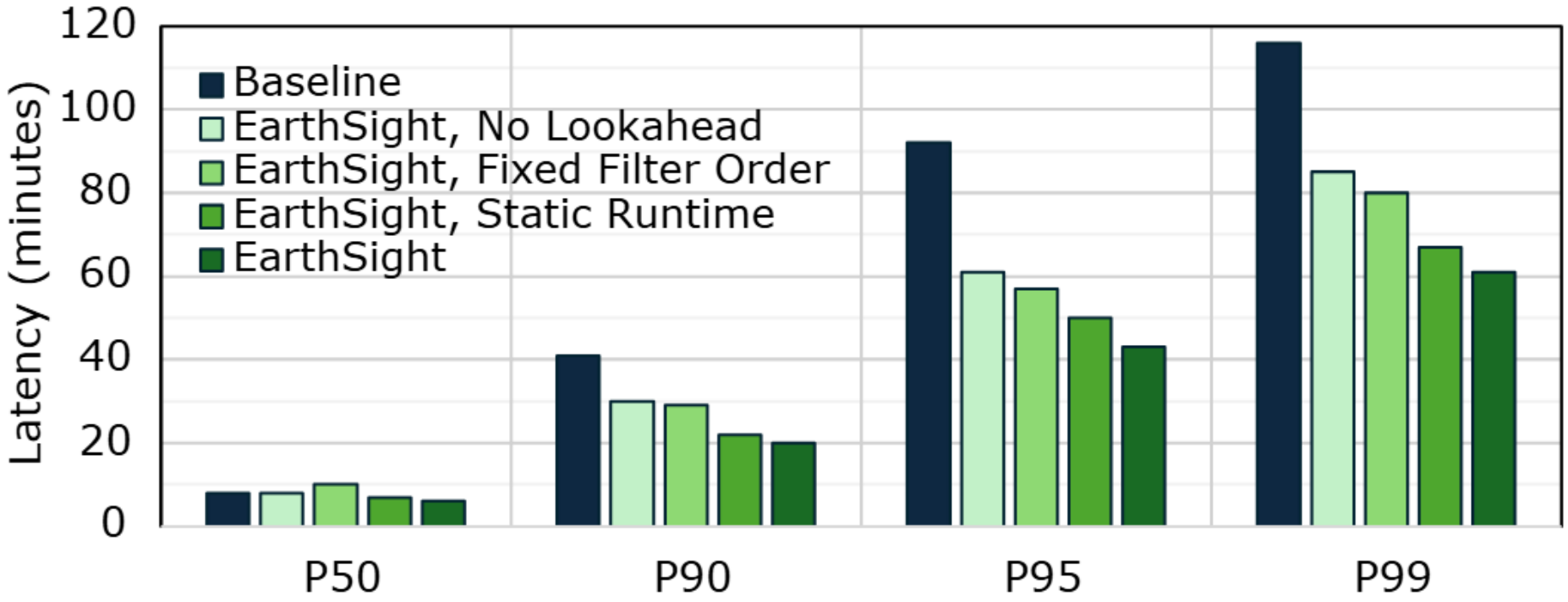
Georgia Tech

EarthSight reduces P90 image delivery latency from 51 to 21 minutes under load



Georgia Tech

All Components of EarthSight are Important



Georgia Tech

Future Work

- **Our work assumes satellites cannot communicate directly with each other**
  - Realistic for status quo, but recent deployments might change that
  - If intersatellite-linkages grow in adoption:
    - We can balance load and energy among satellites
    - We need more complex, constellation-wide scheduling!

Thank you!

Paper and Code:

https://mlsys.org/virtual/2026/oral/3792

Georgia Tech

KAIST

SCAI

SYSTEMS FOR SCALABLE INTELLIGENCE

Georgia Tech