USENIX ATC '21

Video Analytics with Zero-streaming Cameras

(*)**Mengwei Xu**[1,2], (*)Tiantu Xu[3], Yunxin Liu[4], Felix Xiaozhu Lin[5]

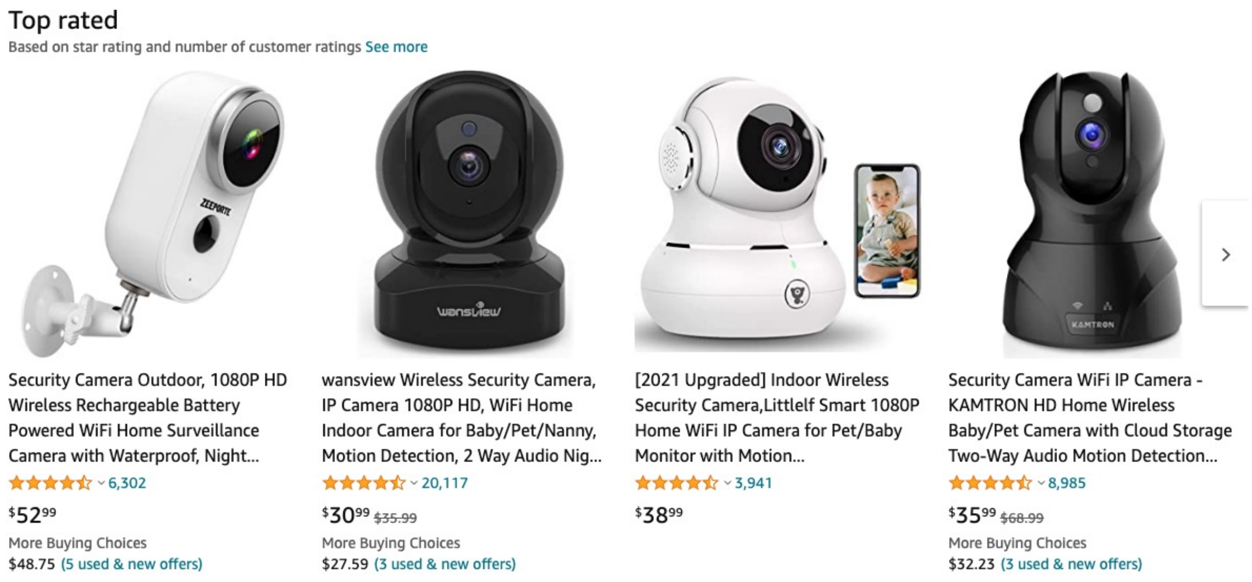
(*) = co-primary

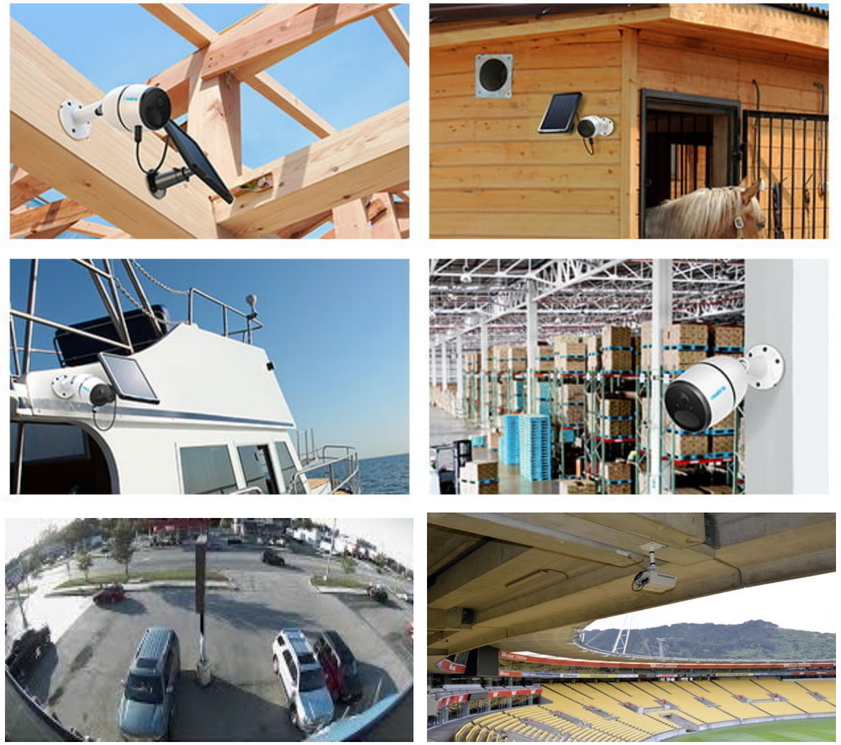
[1]Peking University, [2]Beijing University of Posts and Telecommunications,

[3]Purdue University, [4]Tsinghua University, [5]University of Virginia

Trends of surveillance cameras

- Low-cost, wireless cameras are growing exponentially and enabling ubiquitous intelligence

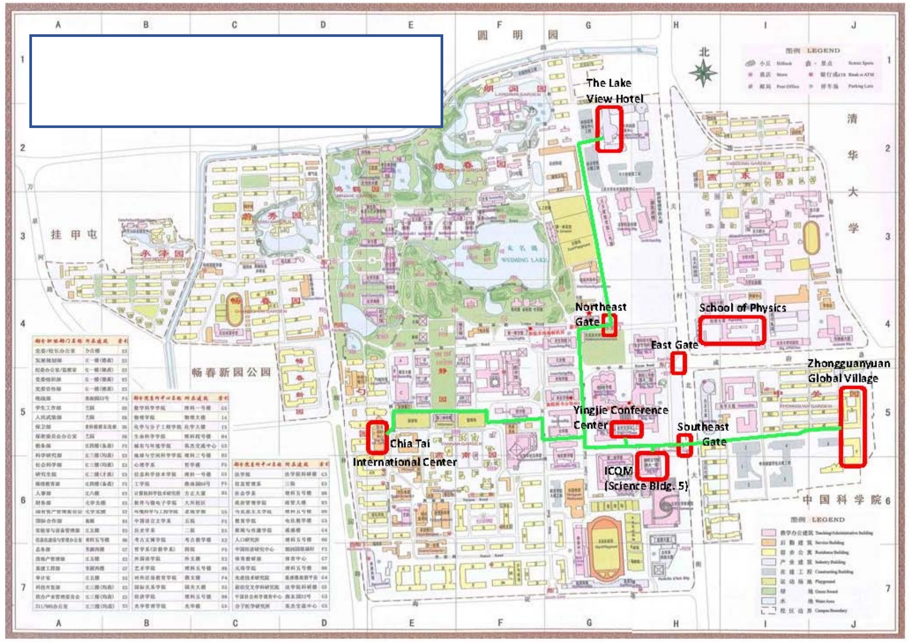




Query: return all frames last week that contains a bus

Trends of surveillance cameras

- Low-cost, wireless cameras grow exponentially and enable ubiquitous intelligence
- Most videos are *cold* (i.e., never used till deletion)
  - We target <u>retrospective query</u>



A campus spanning around 1mi^2

- equipped more than 1,000 cameras

Analysis over 6-month 3,000,000 hours of videos (around 5.4PB) show that:

- Only <2% cameras were used
- Only <0.005% video data was used

Trends of surveillance cameras

- Low-cost, wireless cameras grow exponentially and enable ubiquitous intelligence
- Most videos are *cold* (e.g., never used till deletion)
- Transmitting cold videos wastes precious wireless bandwidth

$$\text{Network } Usage = Bitrate * Cameras * Time$$

24*7 always-on!

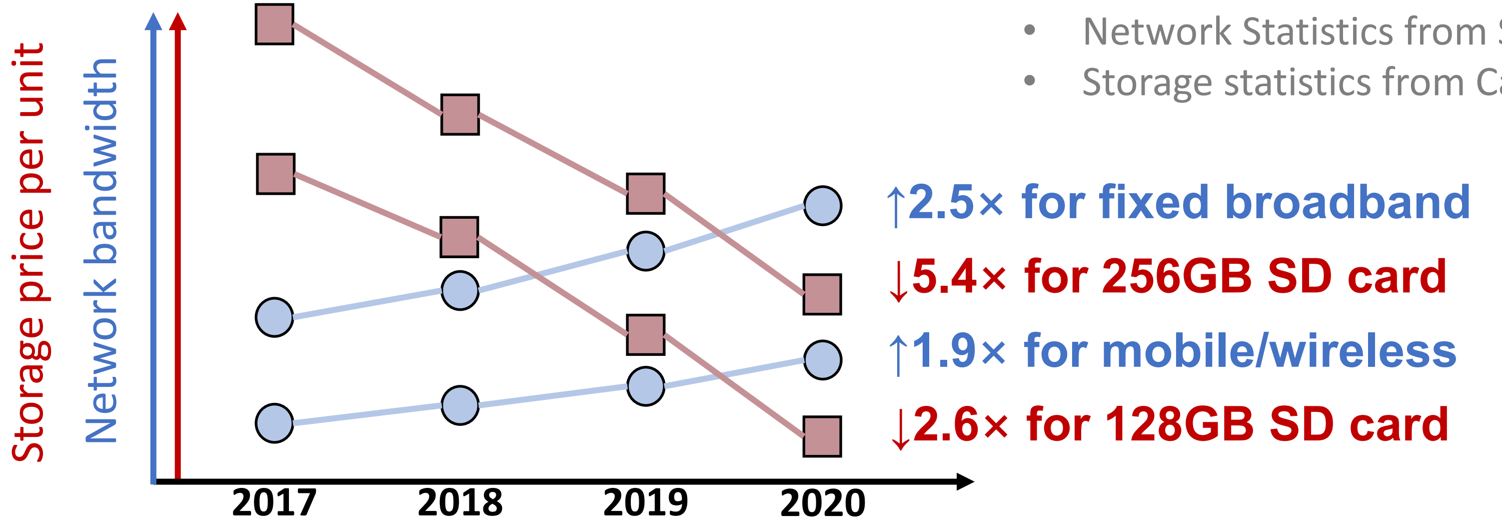
Densely-deployed, large-scale!

1-2Mbps for 720P@30FPS!

(Wireless) bandwidth is for user applications (e.g., video streaming), not cold videos!!

Trends of surveillance cameras

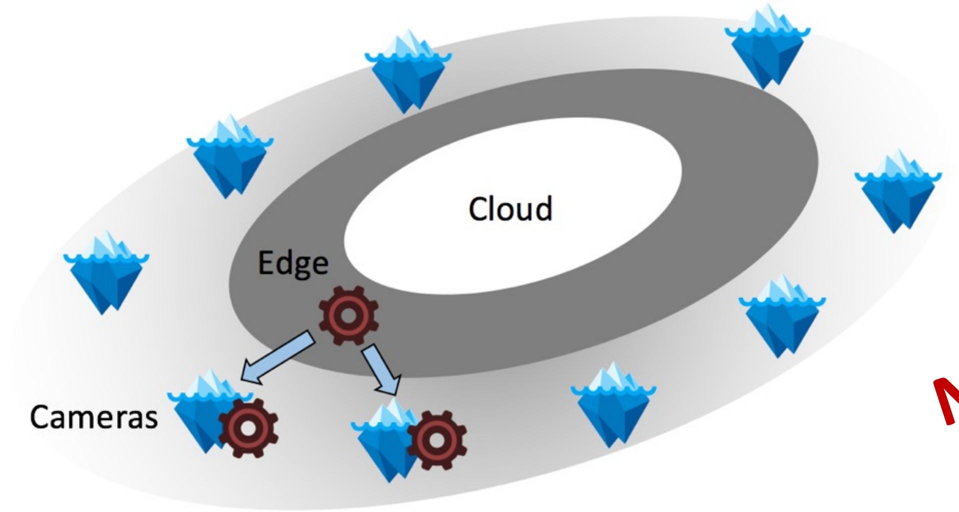
- Low-cost, wireless cameras grow exponentially and enable ubiquitous intelligence
- Most videos are *cold* (e.g., never used till deletion)
- Transmitting cold videos wastes precious wireless bandwidth
- Cheap camera storage can retain videos long enough (weeks to months)



- Network Statistics from SpeedTest
- Storage statistics from Camelcamelcamel

Zero-streaming (ZS) cameras

1. Cameras store videos locally during capture time
2. Cameras respond to servers during query time



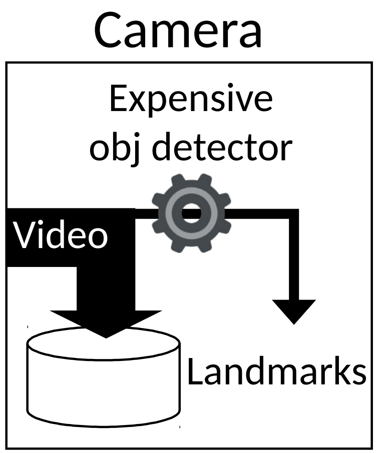
Network/compute resource provisioned on-demand – good for resource efficiency

Zero-streaming (ZS) cameras

1. Cameras store videos locally during ingestion time
2. Cameras respond to servers during query time

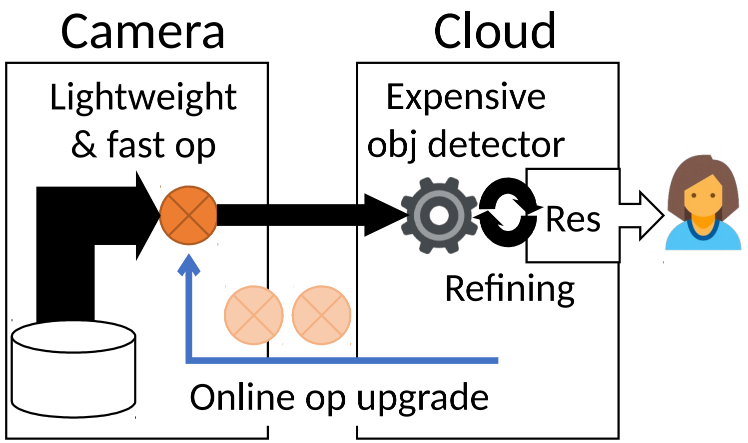
- A key question: how can we query **fast**?
- Challenges we are facing:
    - ❑ Cameras are wimpy (No GPU, RaspberryPi-like)
    - ❑ Network limited (the bottleneck!)
    - ❑ User are waiting (return something useful AFAP)
    - ❑ …

DIVA: a runtime for *ZS* cameras



Capture time: building **landmarks** to capture reliable video knowledge

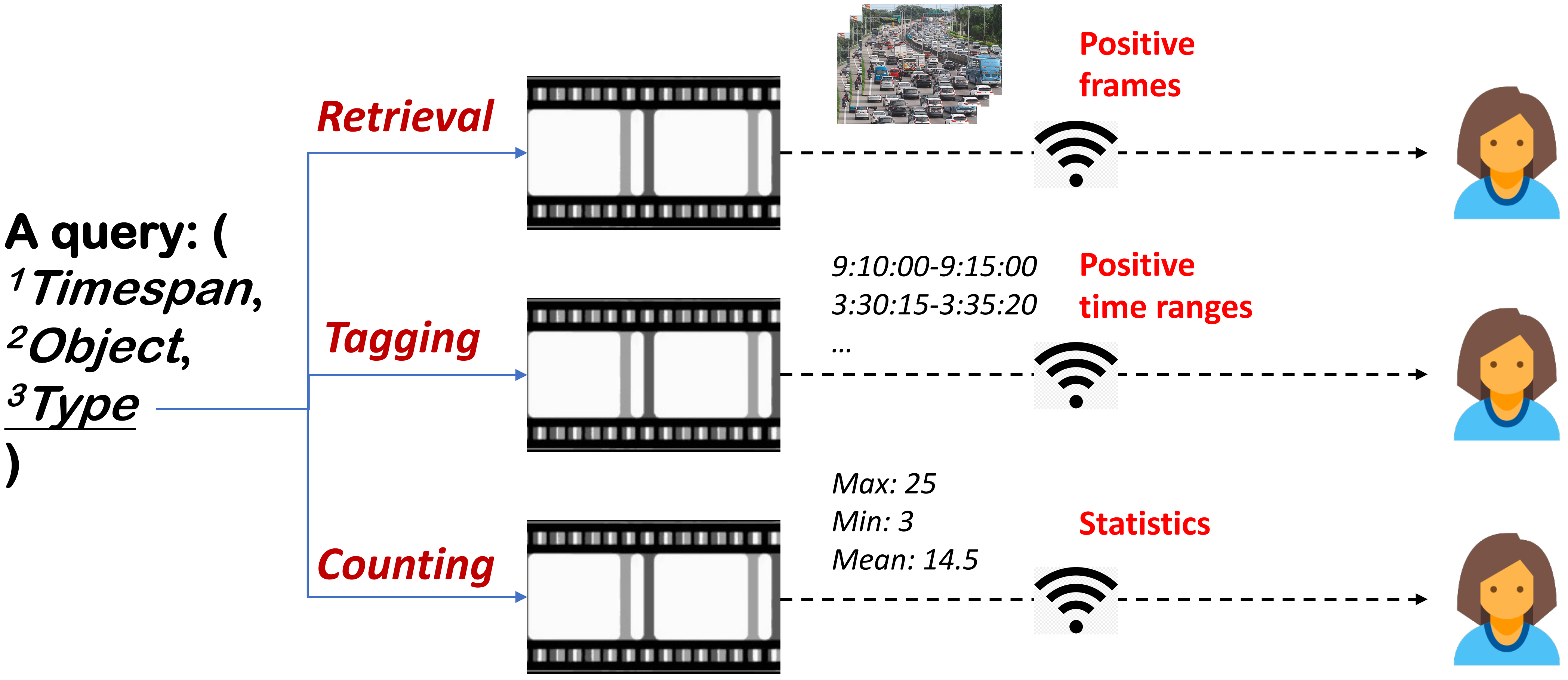
- e.g., in which video areas buses usually appear



Query time: run NNs on camera to prioritize/filter frames to be sent, and update the NNs

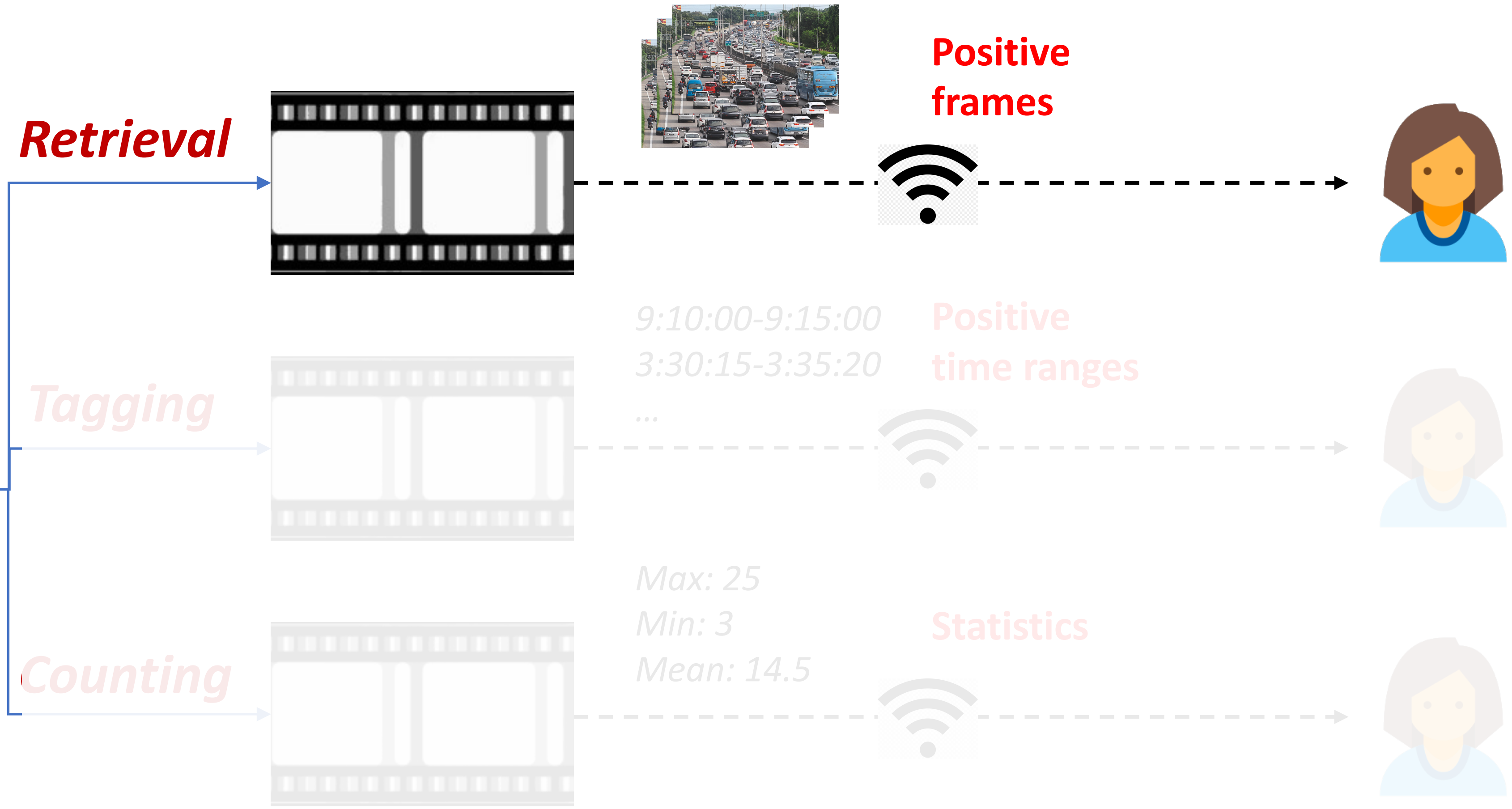
- Results to users are continuously refined

Supported query types in DIVA

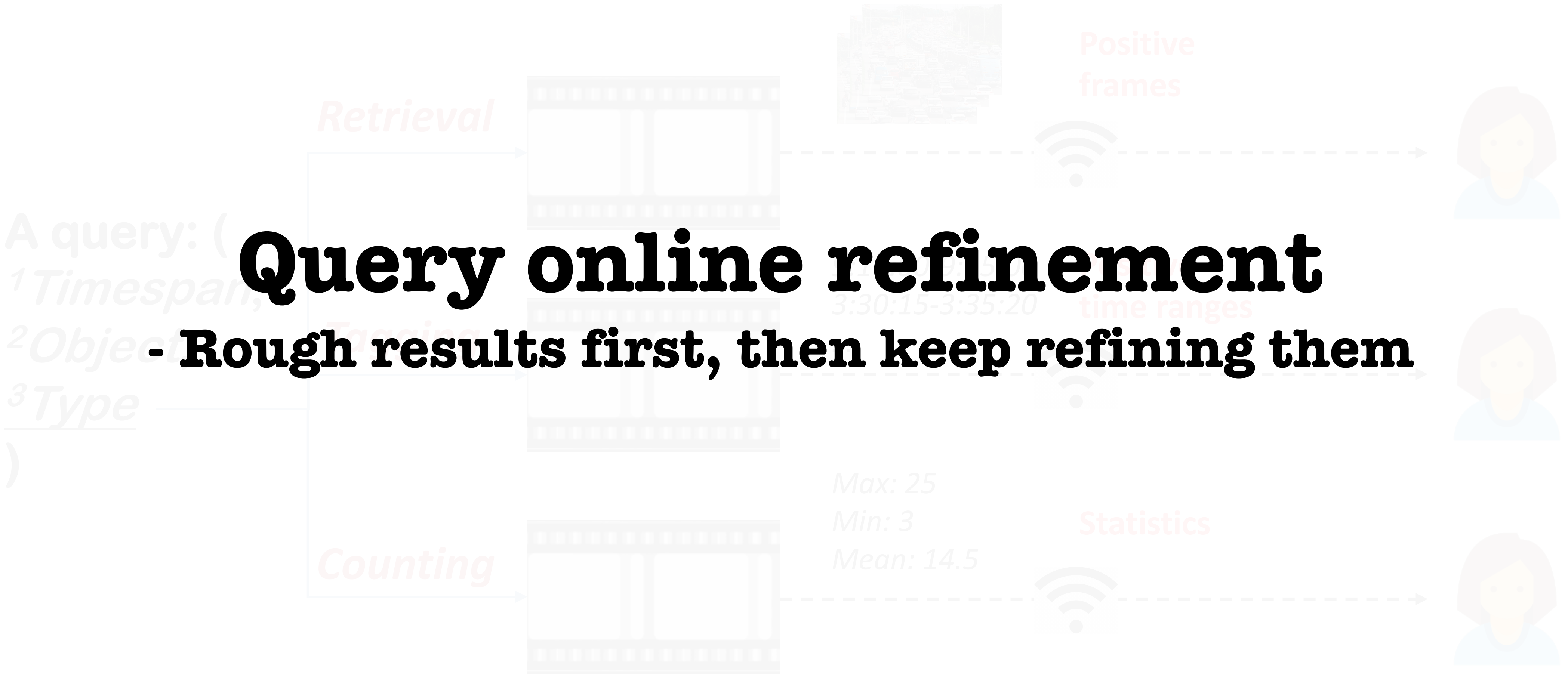


Supported query types in DIVA

A query: (
[1]Timespan,
[2]Object,
[3]Type
)



Supported query types in DIVA



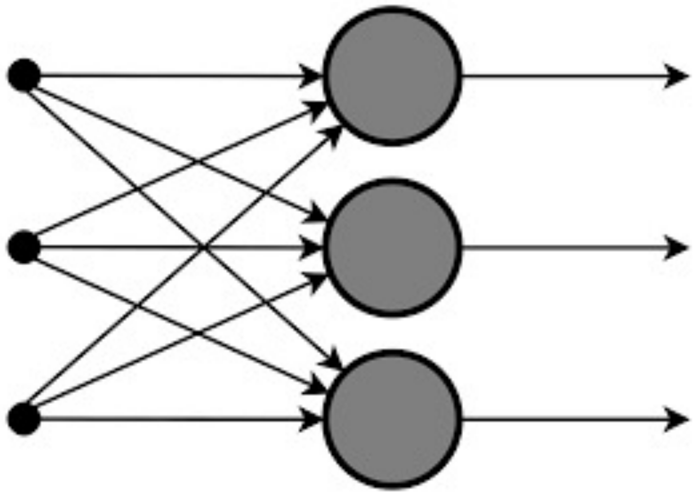
Query online refinement

- Rough results first, then keep refining them

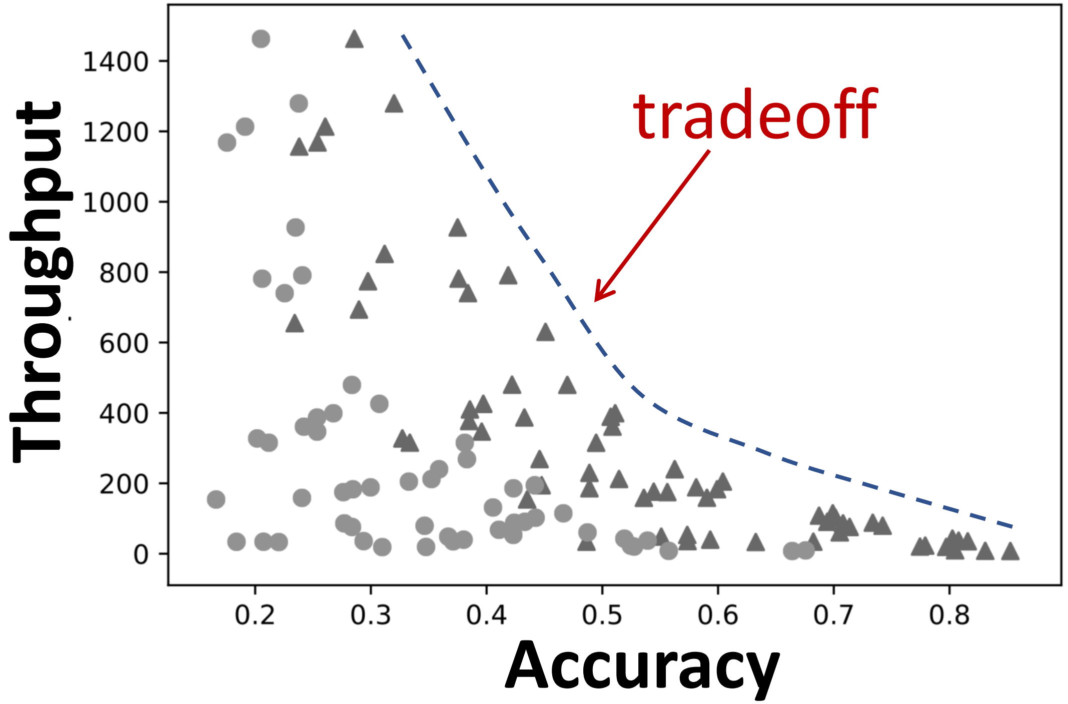
Basics: operators

- Nothing but small neural networks (NNs)
  - Small enough to run fast on cameras (x100s FPS)
  - Rich *accuracy-throughput* tradeoff

Small-sized input (e.g., 50x50)



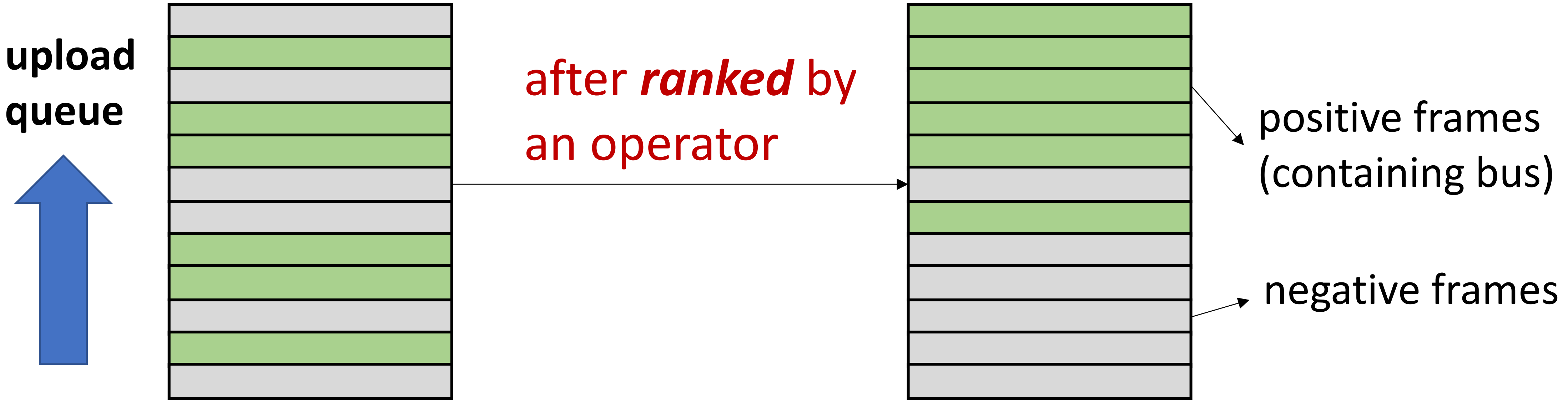
An operator



Basics: operators

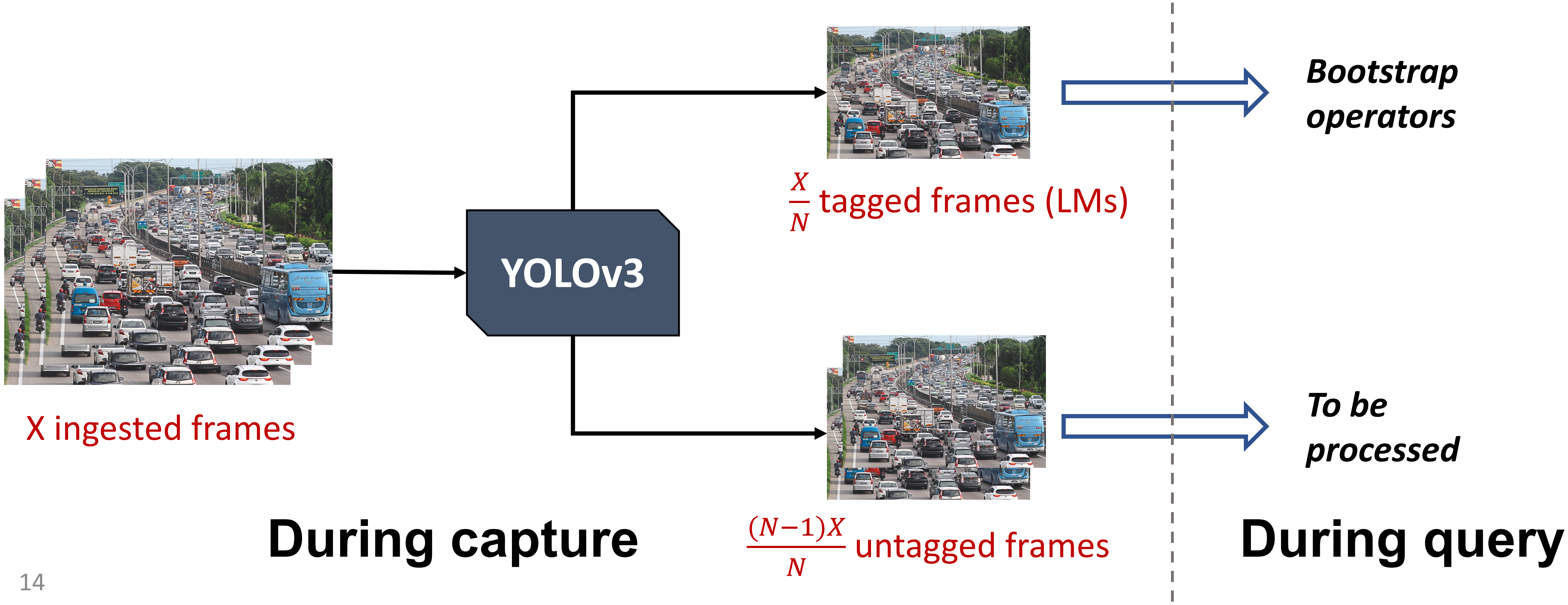
- Nothing but small neural networks (NNs)
  - Small enough to run fast on cameras (x100s FPS)
  - Rich *accuracy-throughput* tradeoff

- How operators serve? As ***rankers*** or ***filters***



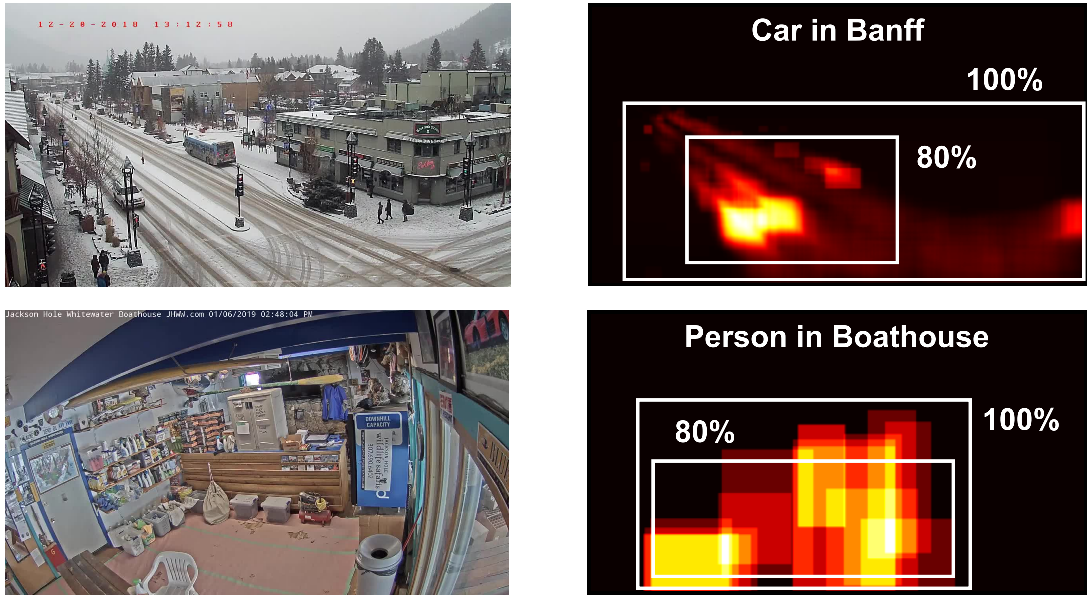
Design #1: landmarks (capture time)

- Running the expensive model on captured frames regularly (sparsely)
- Landmarks are used to train operators

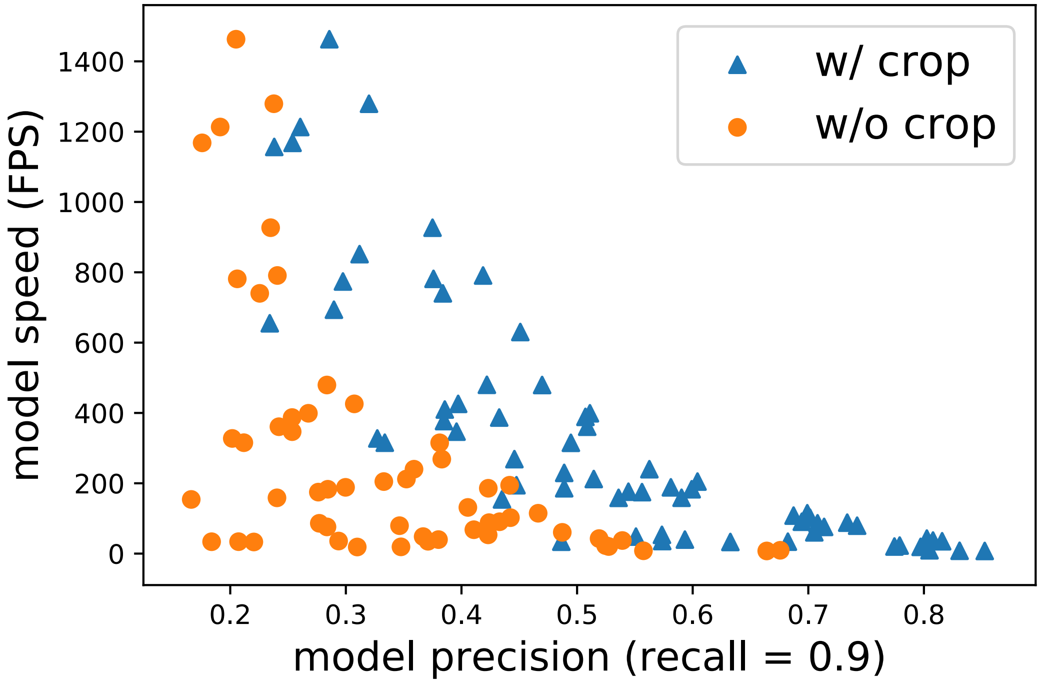


Design #1: landmarks (capture time)

- Key idea: exploiting spatial skews of video objects
  - So operators can be more focused on areas of interests



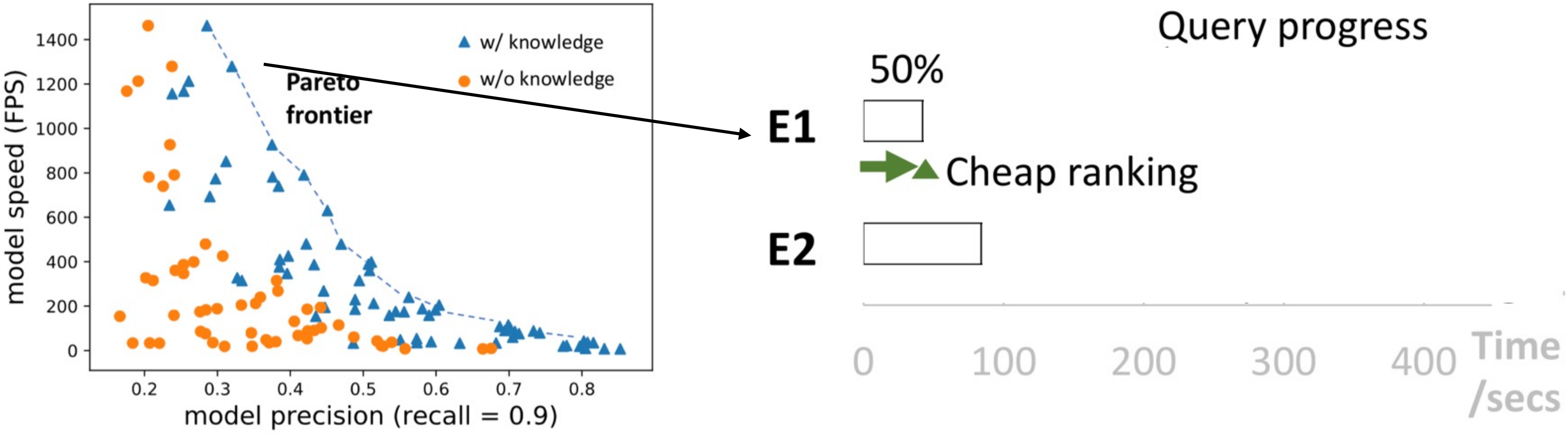
Object spatial skews is pervasive



Cropping improves op performance

Design #2: operator upgrade (query time)

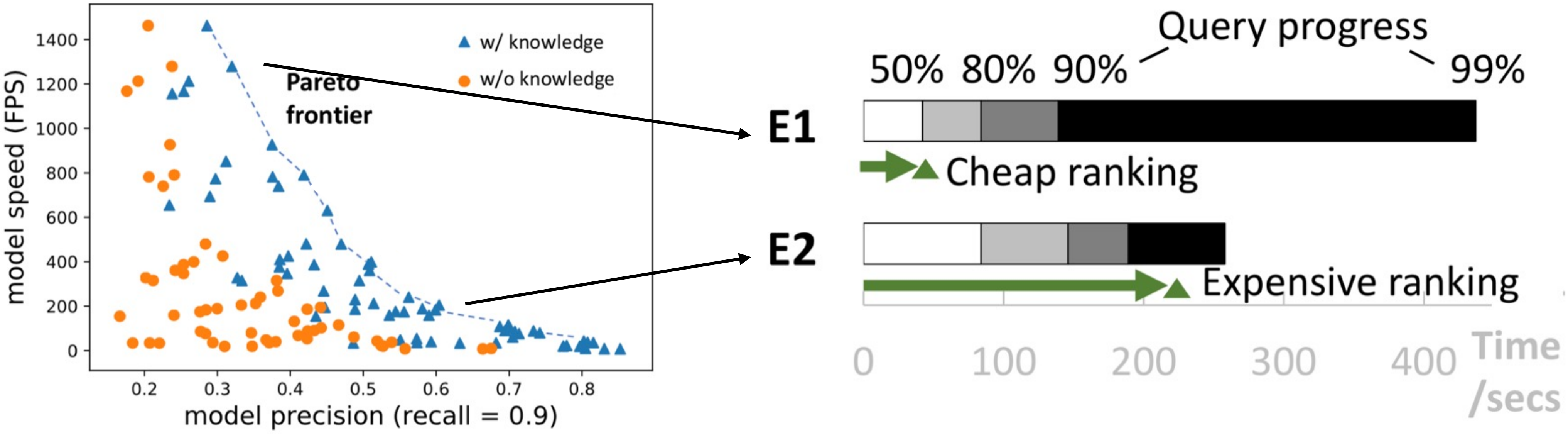
- What operator to use? No **silver bullet**!



Fast (yet inaccurate) operators win at early stage

Design #2: operator upgrade (query time)

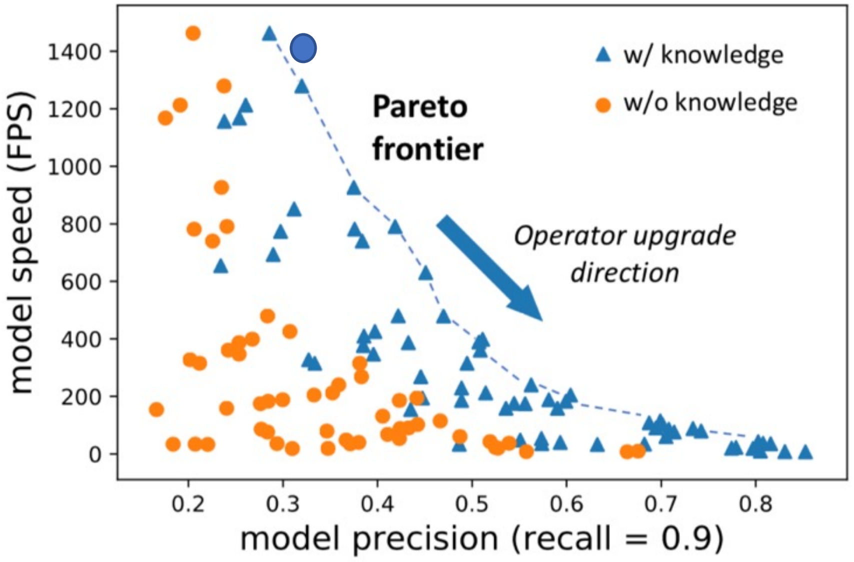
- What operator to use? No **silver bullet**!

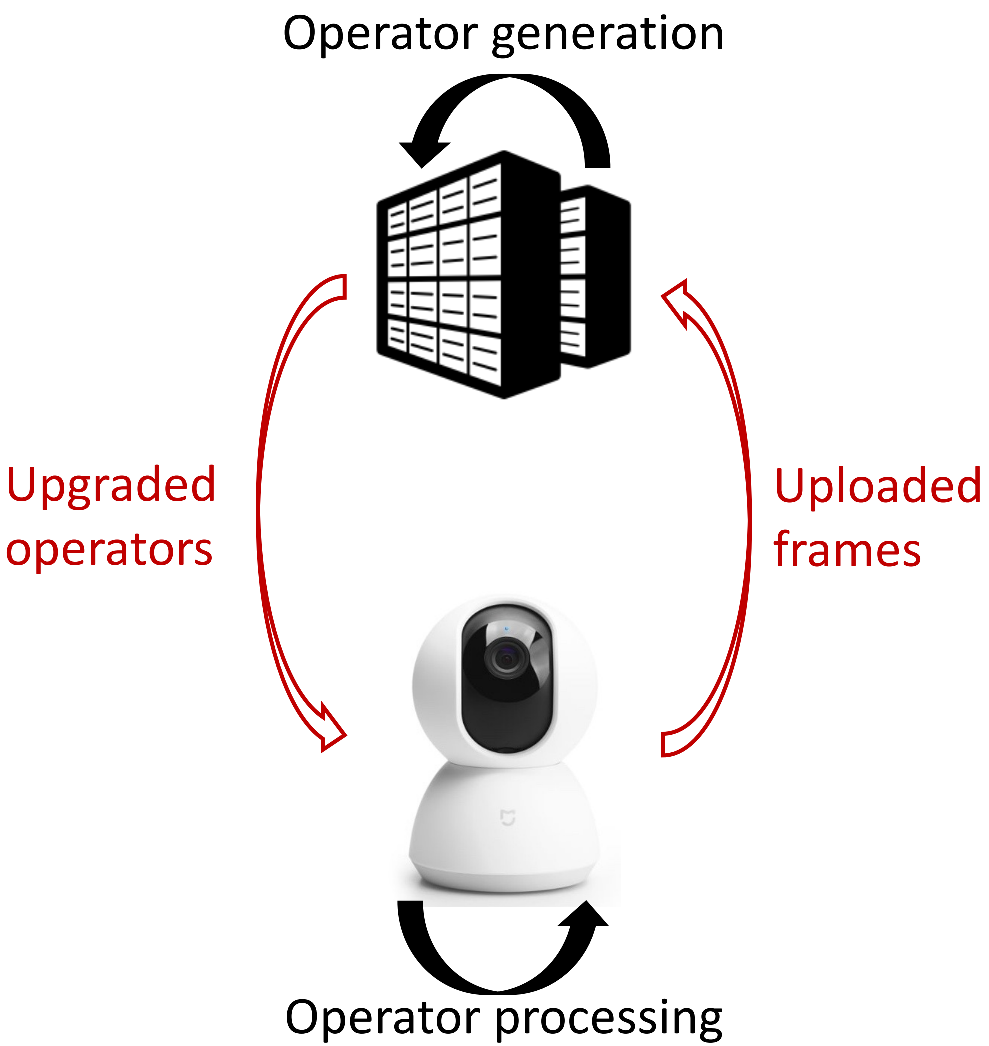


Accurate (yet slow) operators win at later stage

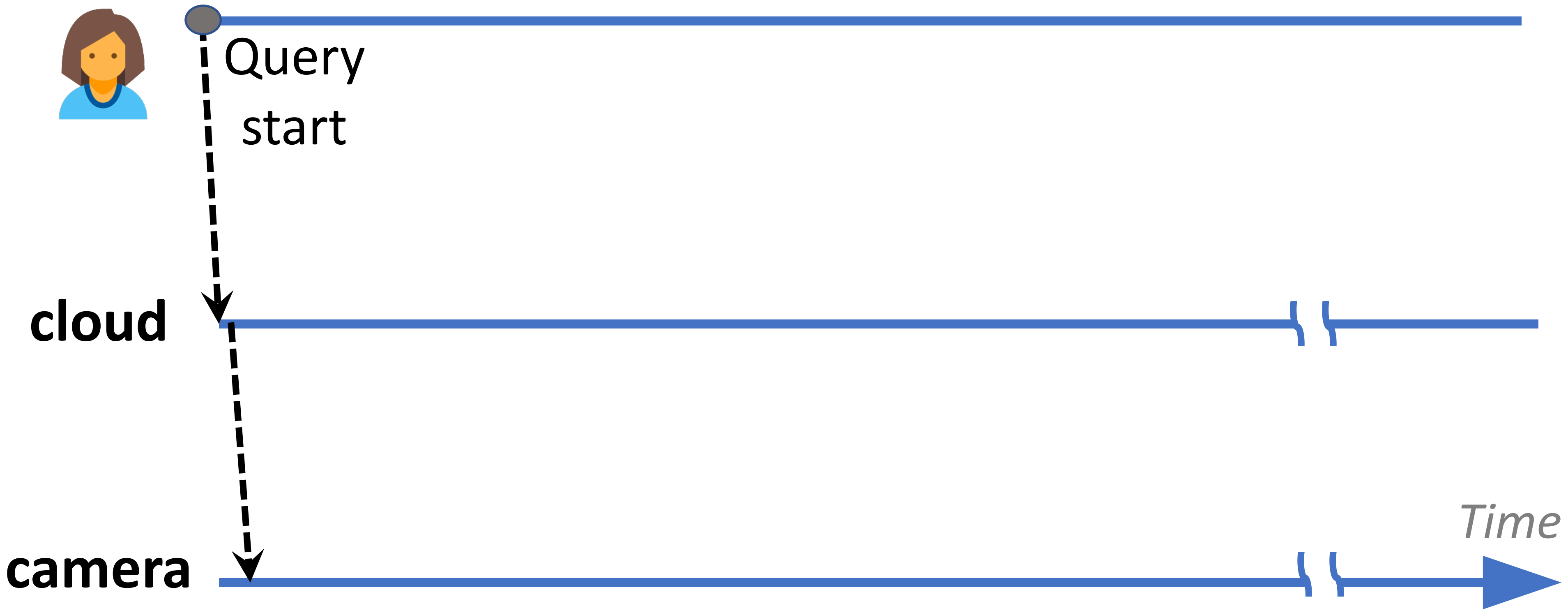
Design #2: operator upgrade (query time)

- Multipass, multi-operator execution

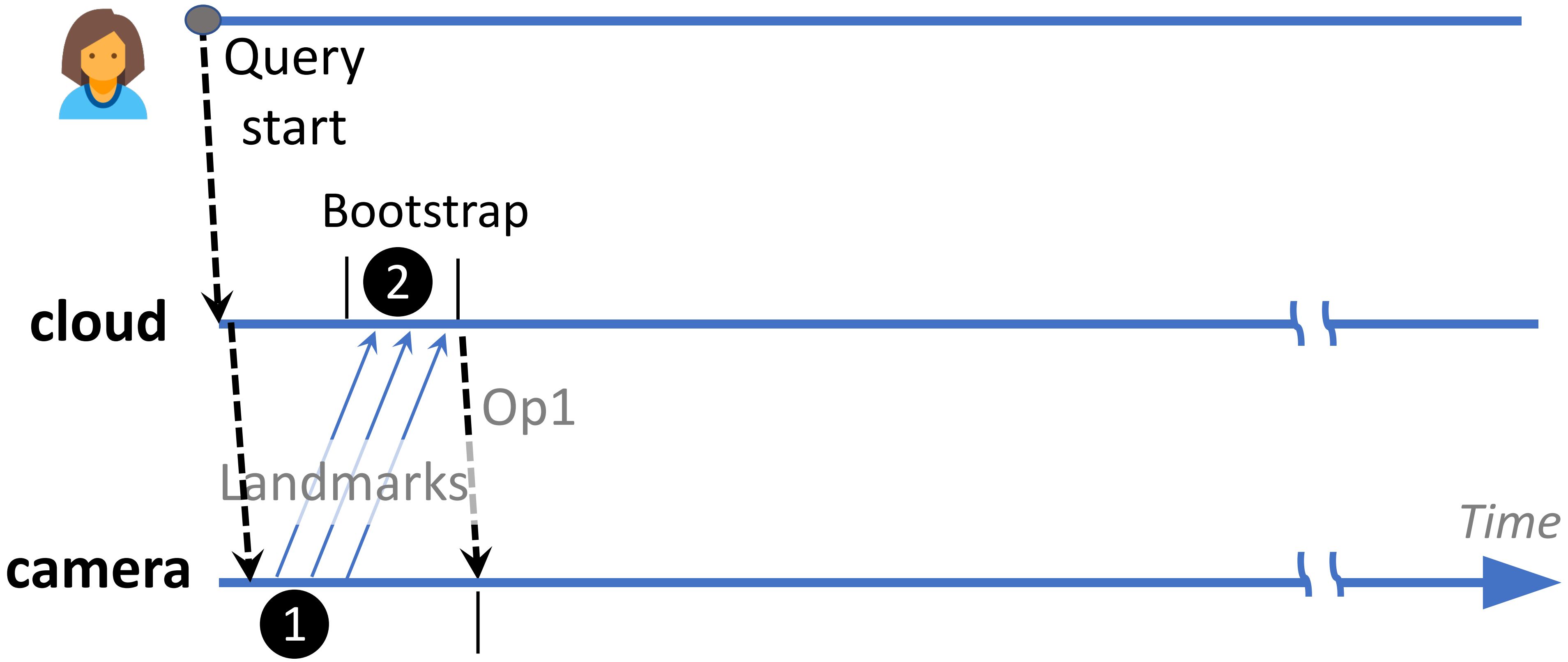




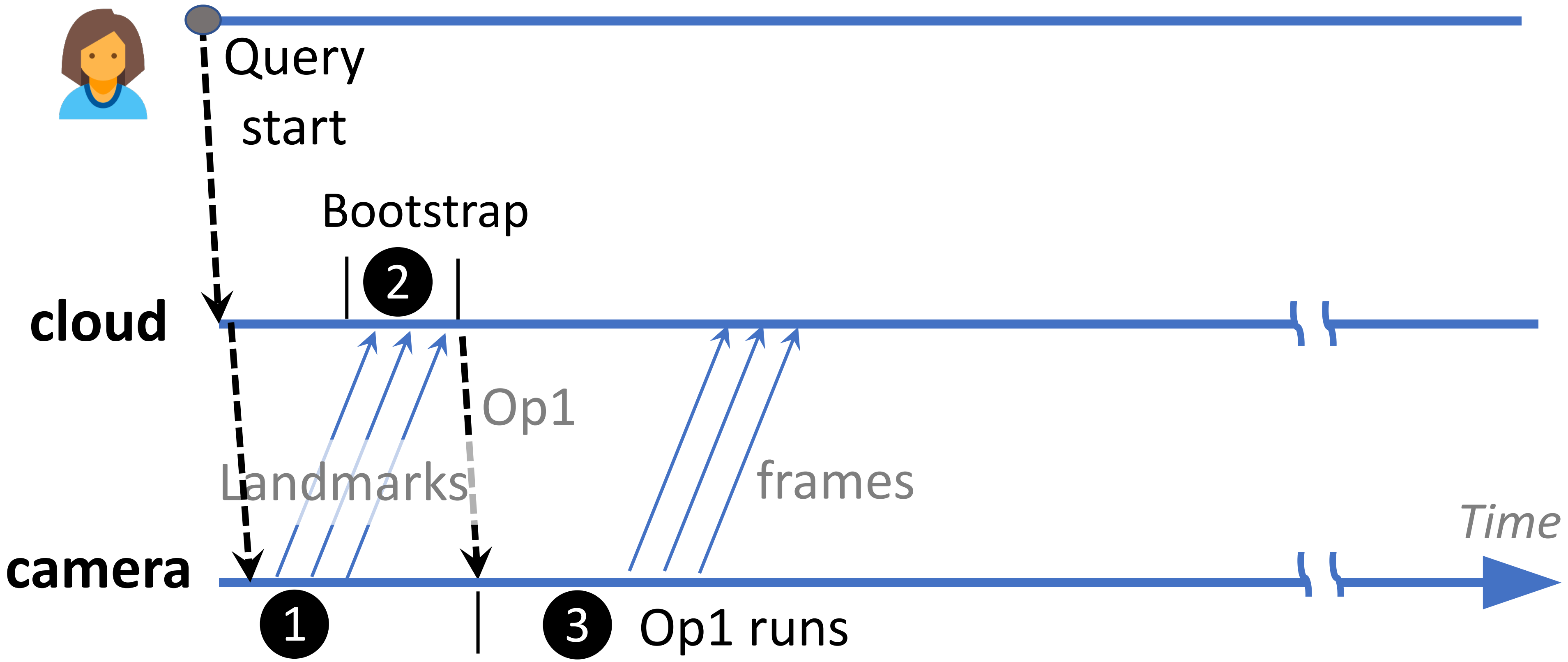
Concrete execution plan



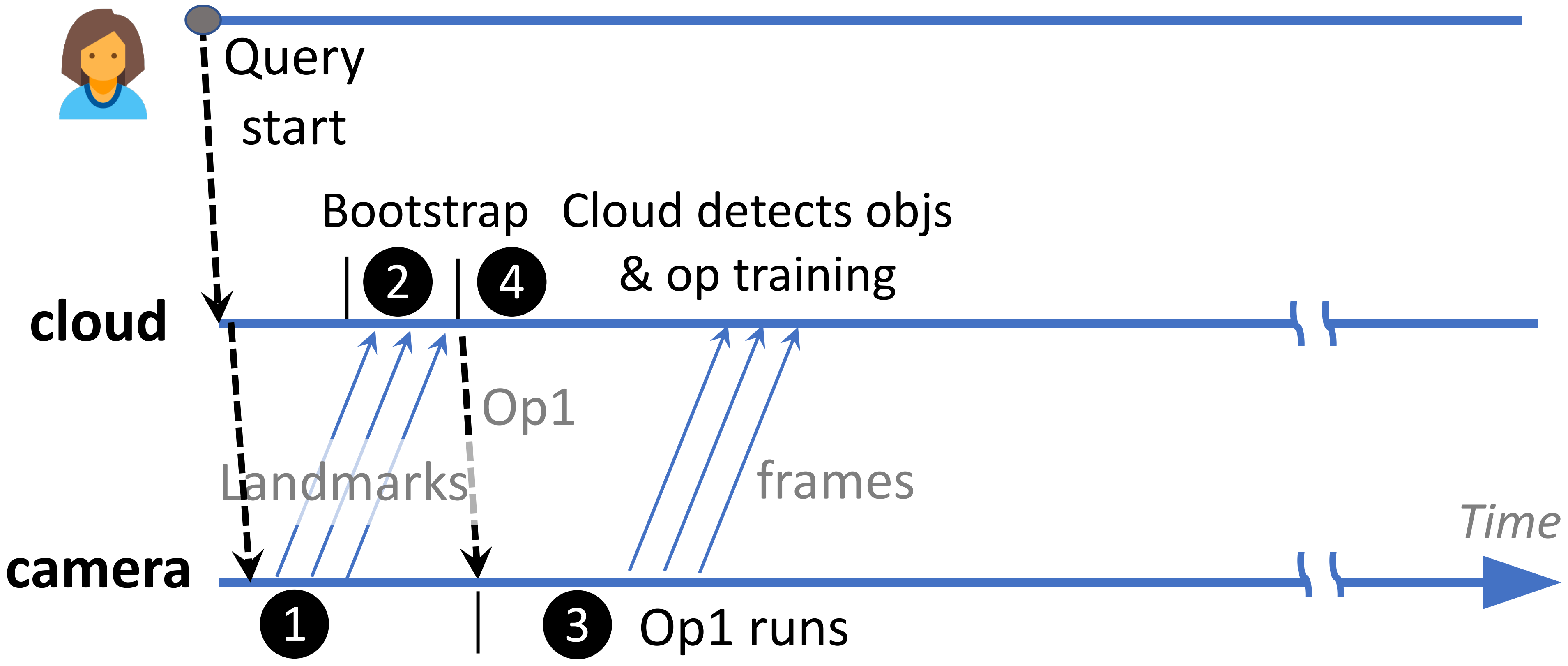
Concrete execution plan



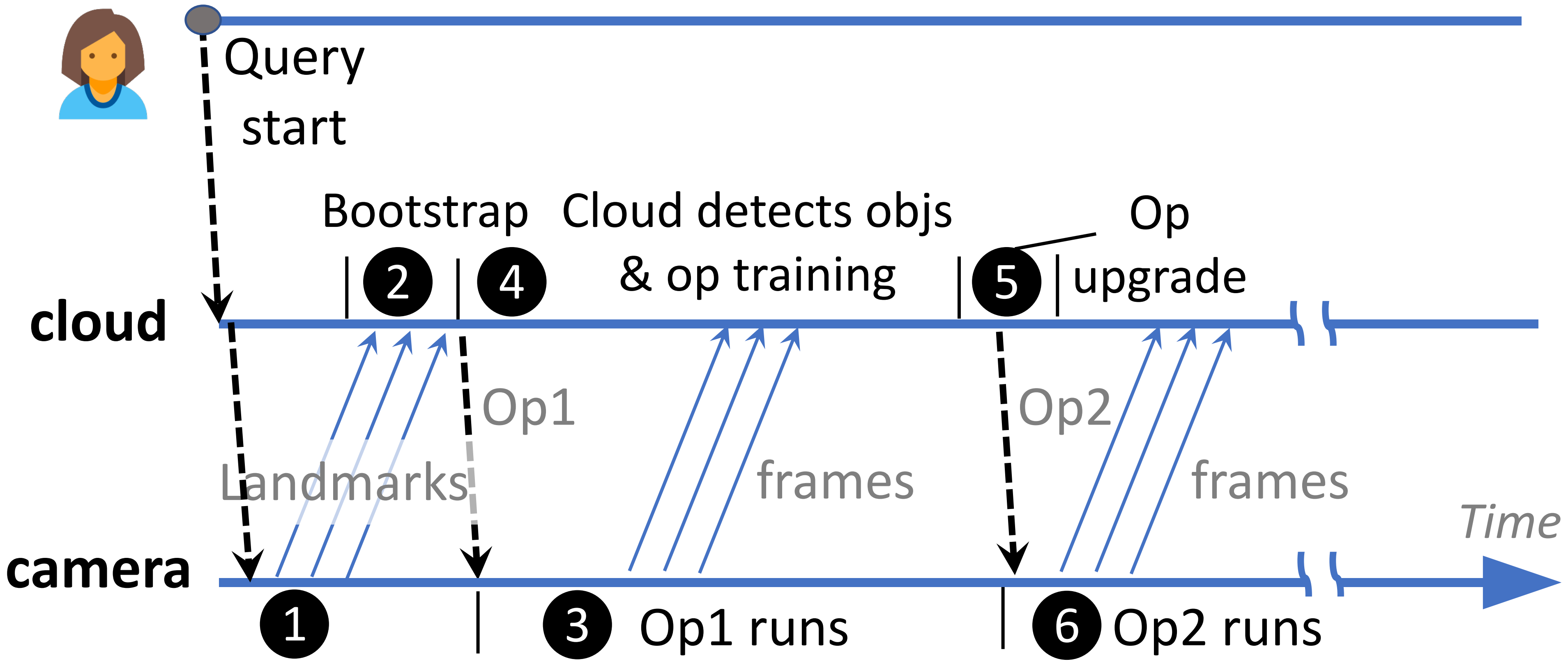
Concrete execution plan



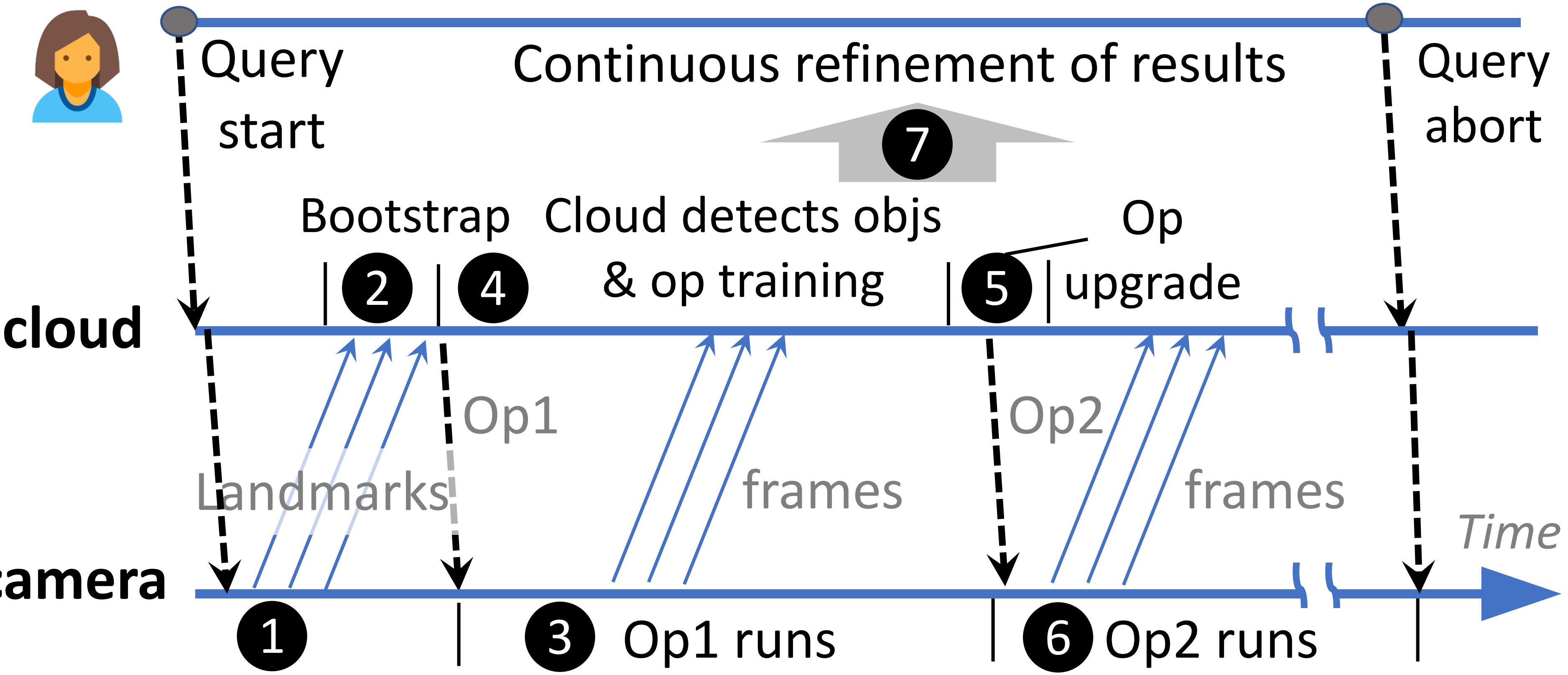
Concrete execution plan



Concrete execution plan



Concrete execution plan



Concrete execution plan

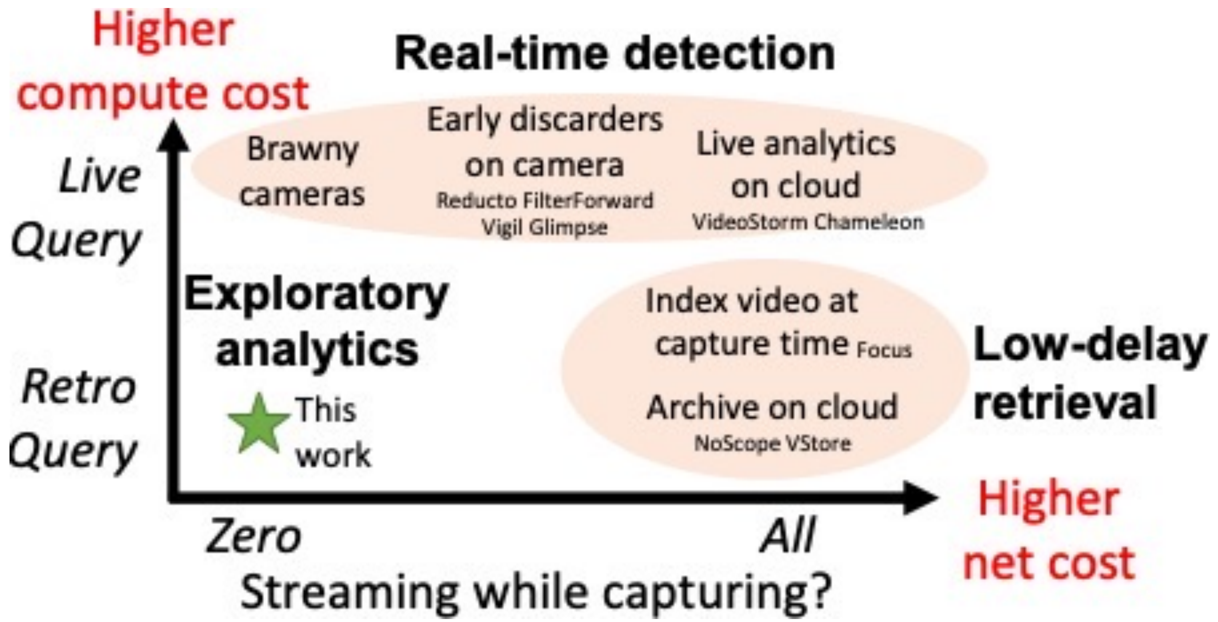
Detailed questions:

- When to upgrade an operator?
- What operator to be upgraded to?
- What frames to be processed first?

Please refer to our paper for details!

More about DIVA

- Scaled to more cameras? Just adding more GPUs.
- DIVA is complementary to real-time video analytics, which shall be deployed to critical regions, e.g., banks.



Experiment settings

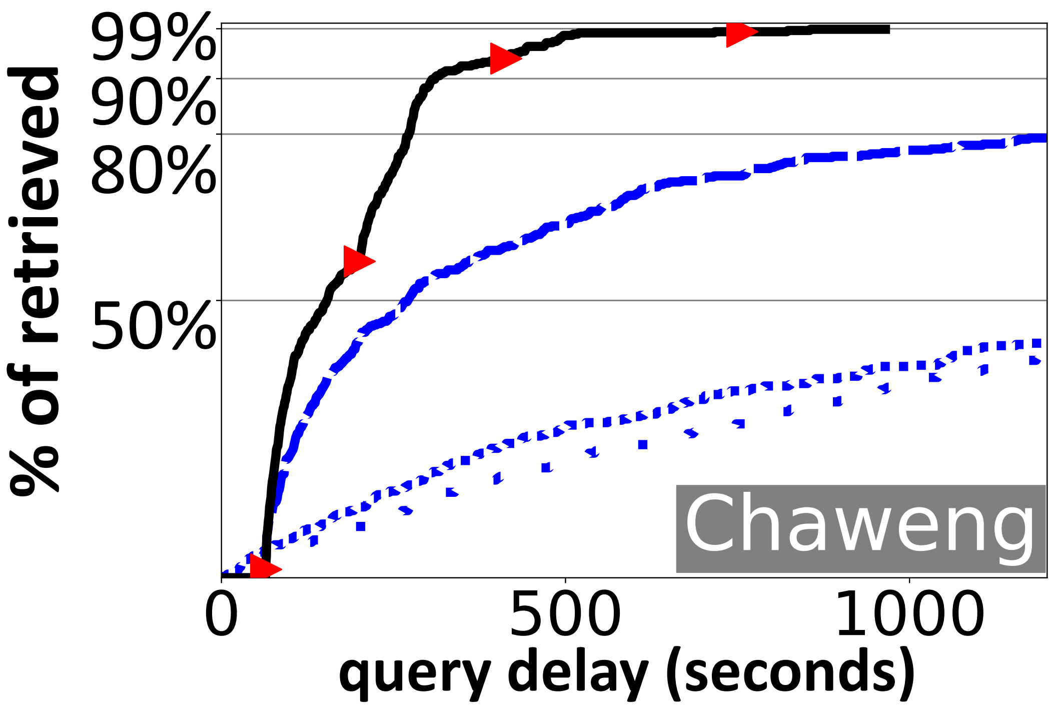
- 15 public video streams from YouTube
  - Per stream: 48 hours, 720P@1FPS

- Hardware: RPI 3B/Odroid XU4 (Camera) + Nvidia Titan V (Server)
- Network: 0.1-10MB/s (1MB/s by default)
- Baselines
  - CloudOnly, OptOp[1], PreIndexAll[2]

[1] Kang, Daniel, et al. "Noscope: optimizing neural network queries over video at scale." *VLDB* 2017.
[2] Hsieh, Kevin, et al. "Focus: Querying large video datasets with low latency and low cost." *OSDI* 2018.

Highlighted results

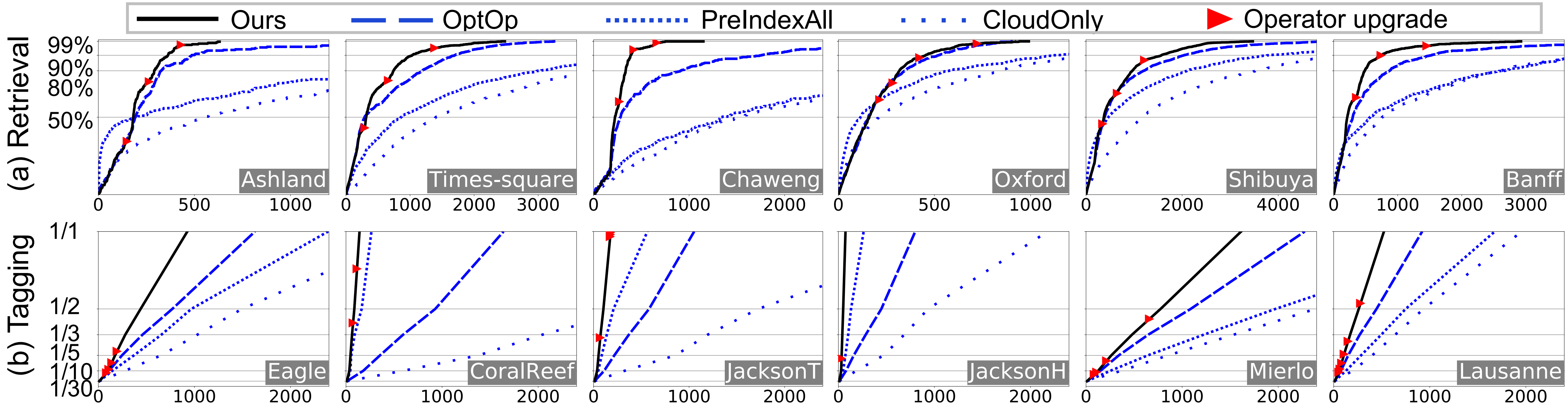
- DIVA outperforms the baselines throughout the query process



Q: Retrieving frames containing a bicycle

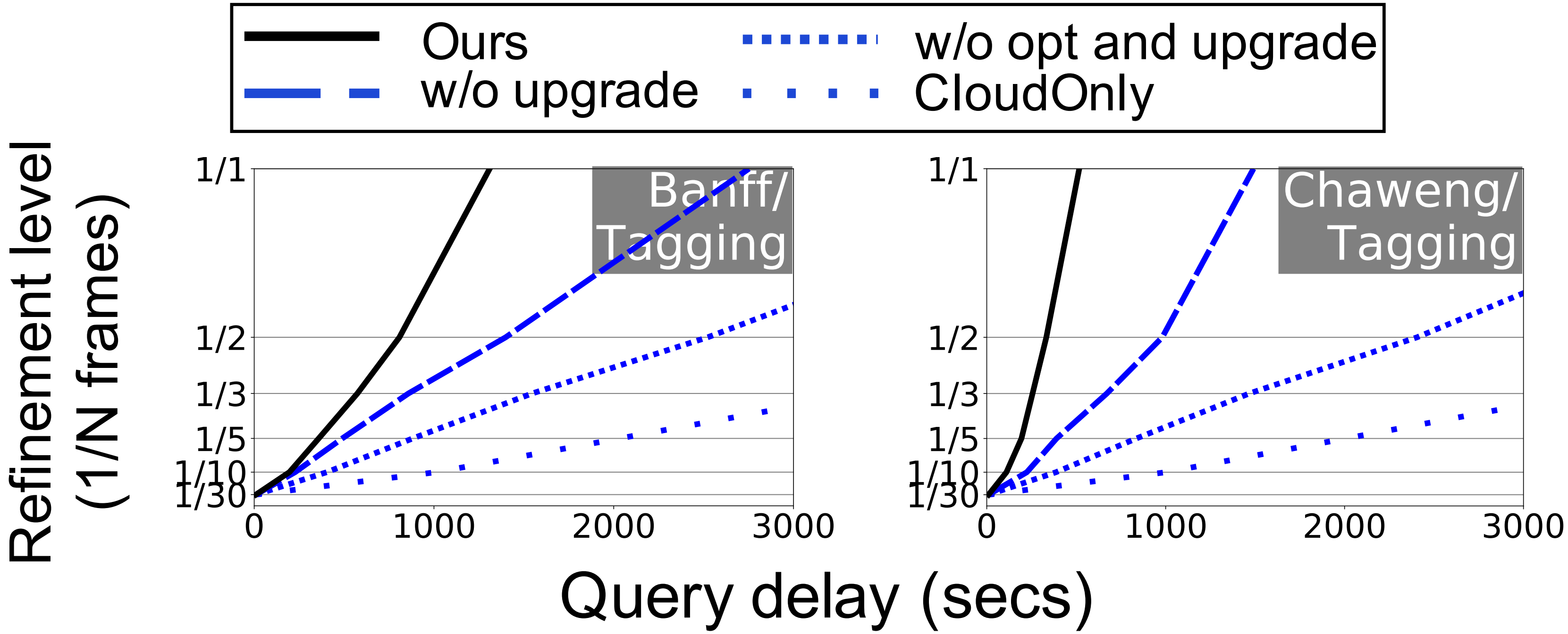
Highlighted results

- DIVA improves end-to-end latency by 4X – 30X to baselines



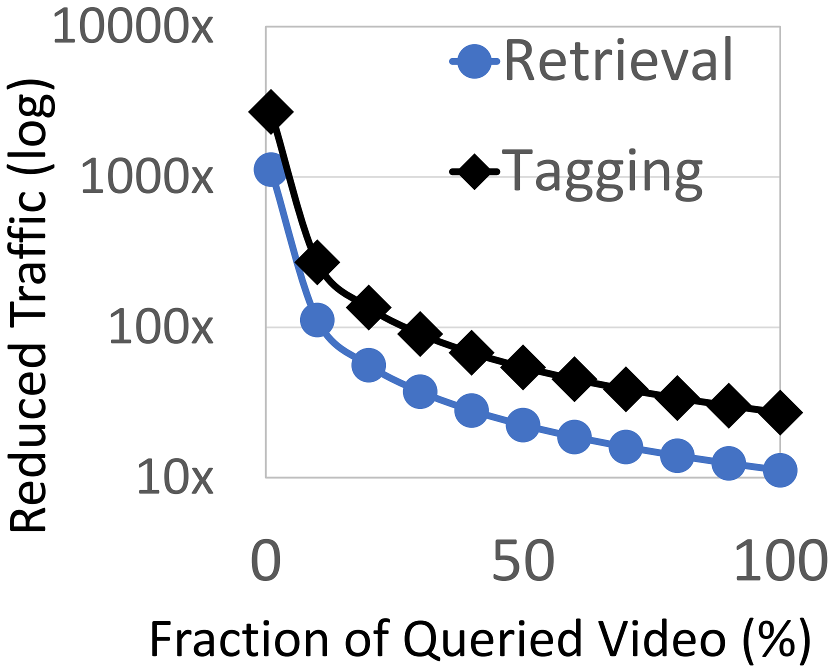
Highlighted results

- Both two key designs of DIVA are critical, e.g., in Tagging:
  - Operator Upgrade brings 2.0X – 3.0X speedup
  - Landmarks bring 1.6X – 3.1X speedup



Highlighted results

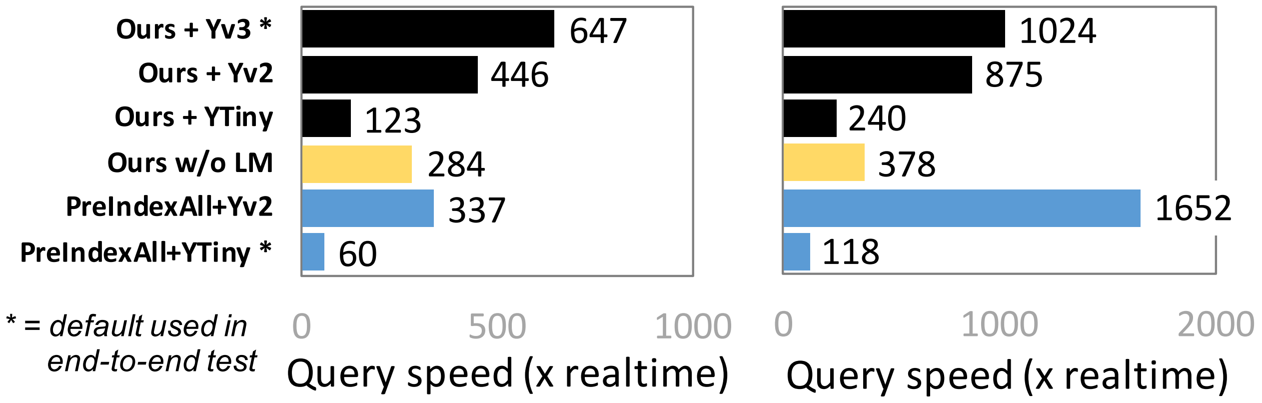
- DIVA saves network bandwidth over “all streaming” by >1,000X as in our campus case study.



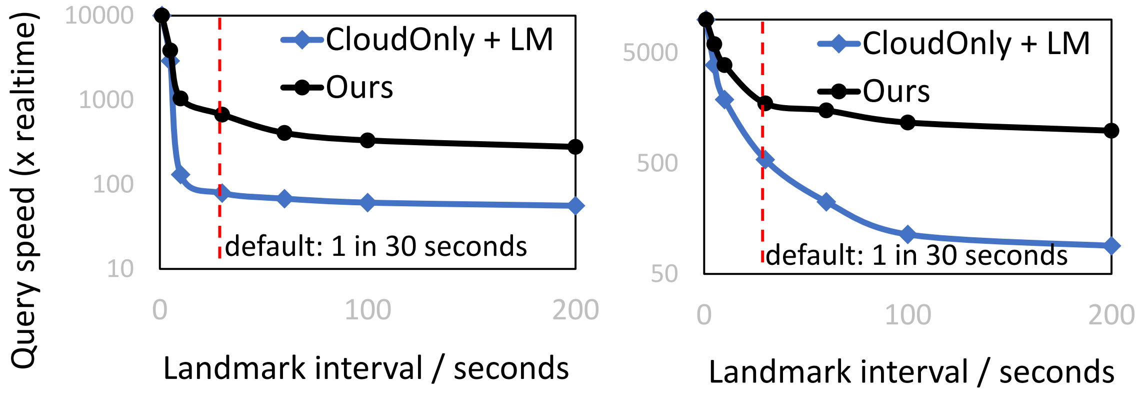
- ➢ *Most frames will not be queried*
- ➢ *Even for queried frames, most of them don’t have to be uploaded*

Highlighted results

- With sparser landmarks, DIVA's performance degrades slowly



- With inaccurate landmarks, DIVA's performance degrades significantly

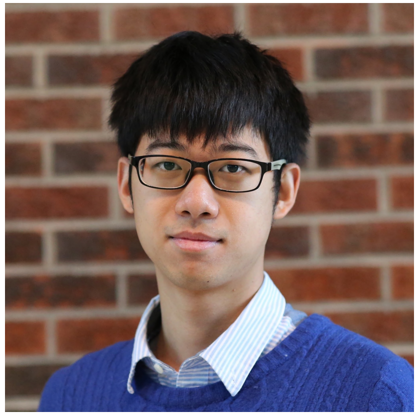


Summary

- Zero-streaming cameras towards high resource efficiency
  - A complement to cloud-centric approach
- DIVA: the first runtime for zero-streaming cameras
  - Key techniques: landmarks and operator upgrade
- Beyond cameras: cold data is pervasive (IoT, smartphones, etc)!
  - How to query them efficiently?
  - A new research direction?

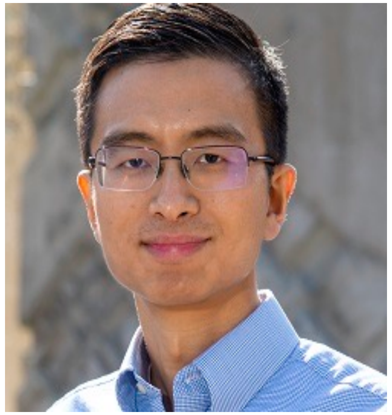
Thanks for your listenting!

Mengwei Xu
PKU & BUPT



Tiantu Xu
Purdue ECE

Yunxin Liu
Tsinghua



Felix Xiaozhu Lin
UVA

Contact: Mengwei Xu (mwx@bupt.edu.cn)