Turbo: Opportunistic Enhancement for Edge Video Analytics

Yan Lu, Shiqi Jiang, Ting Cao, Yuanchao Shu

SenSys 2022

Outline

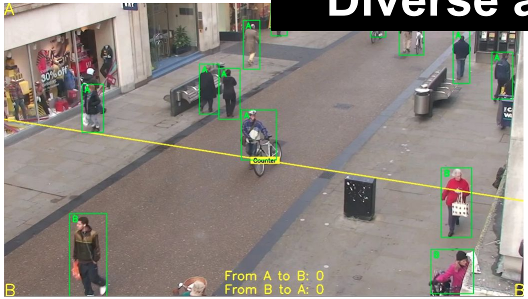
- **Background of edge video analytics**
- **Opportunities for existing VAPs**
  - Idle computing resources
  - Hard samples
  - Image (Data) enhancement
- **Turbo**
  - Detector-specific GAN & Model-aware adversarial training
  - Resource-aware scheduler
- **Experiments**
- **Summary**

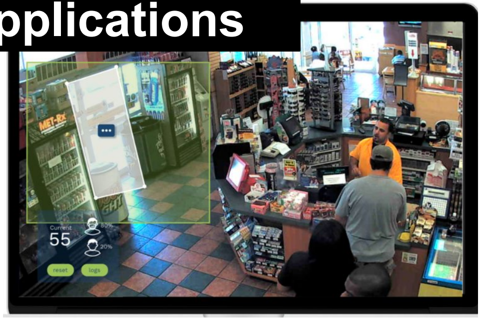
Video is everywhere

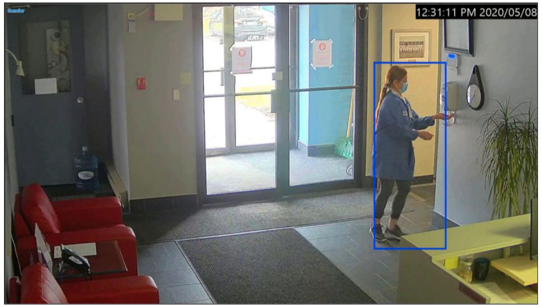
Sensors

Video Analytics

Diverse applications



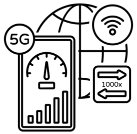




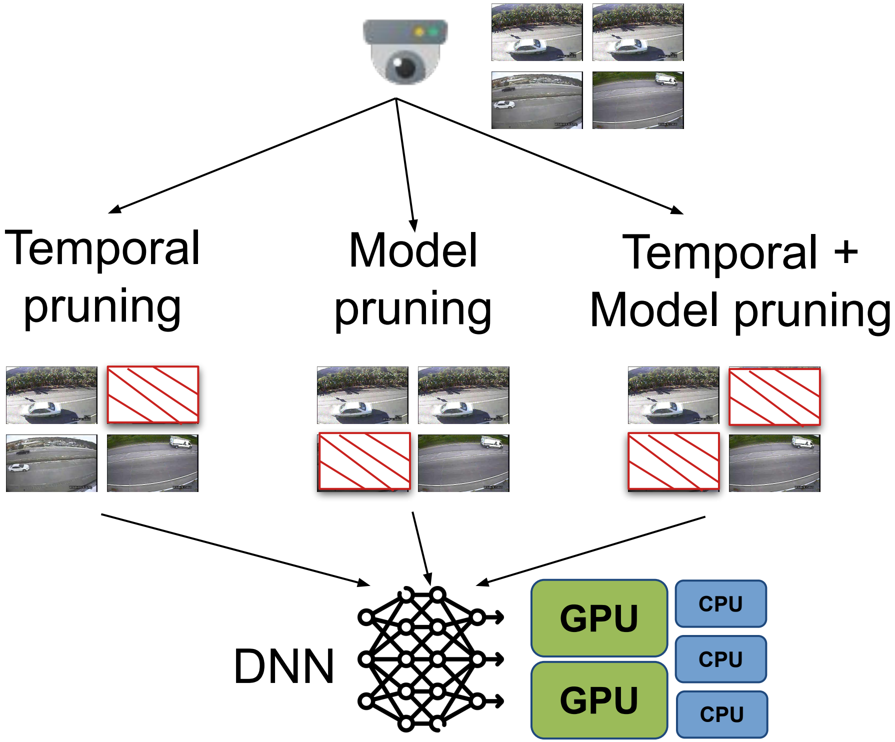
Move to edge

Privacy

Network

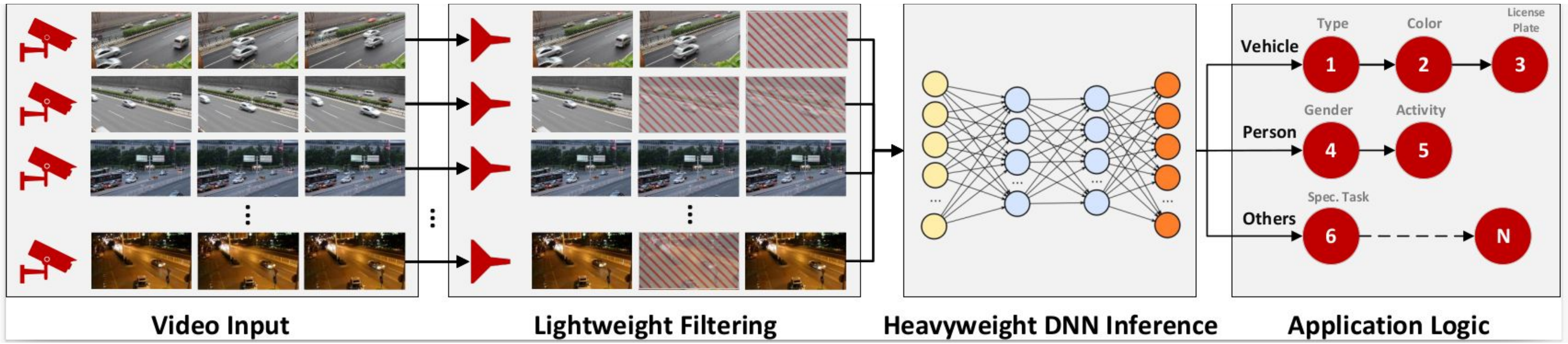


Edge Video
Analytics Pipelines
(2015~2020)



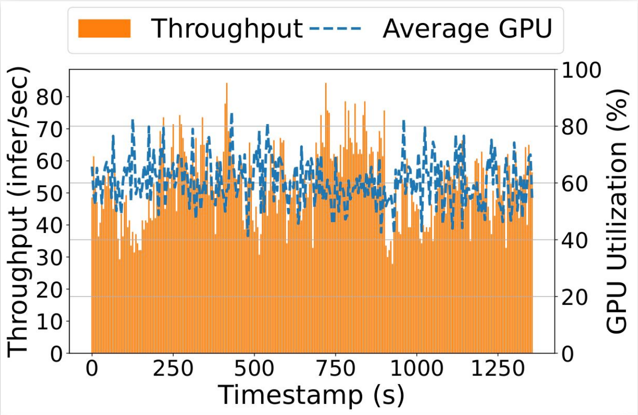
How many resources?
Usually, they are set to meet 4 fps instead of 2 or 3 fps!

Idle resources are common

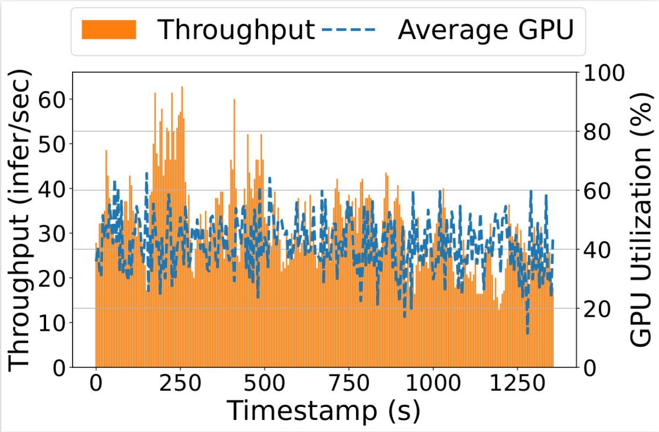


Can we **leverage** these idle resources to **improve** video analytics?

Idle resources are common



Vigil

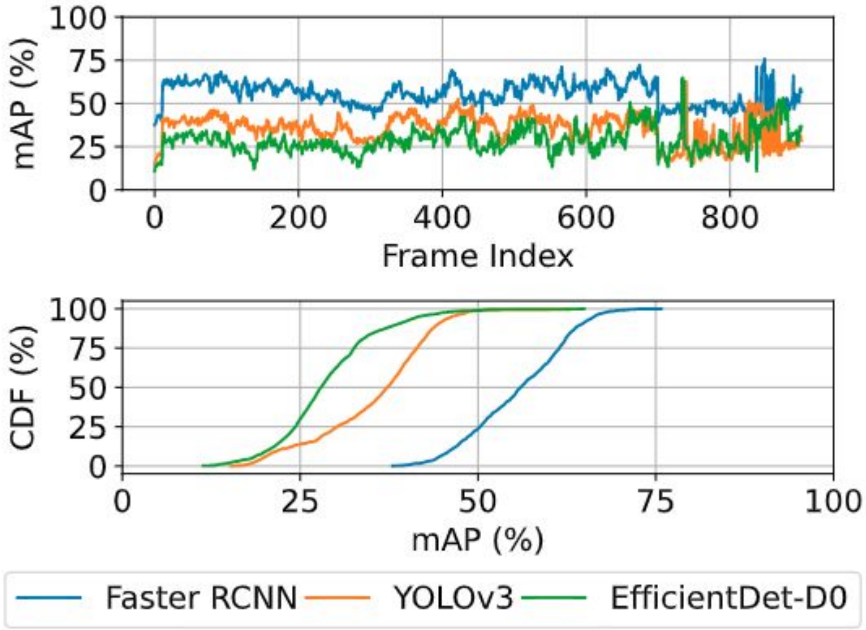


Glimpse

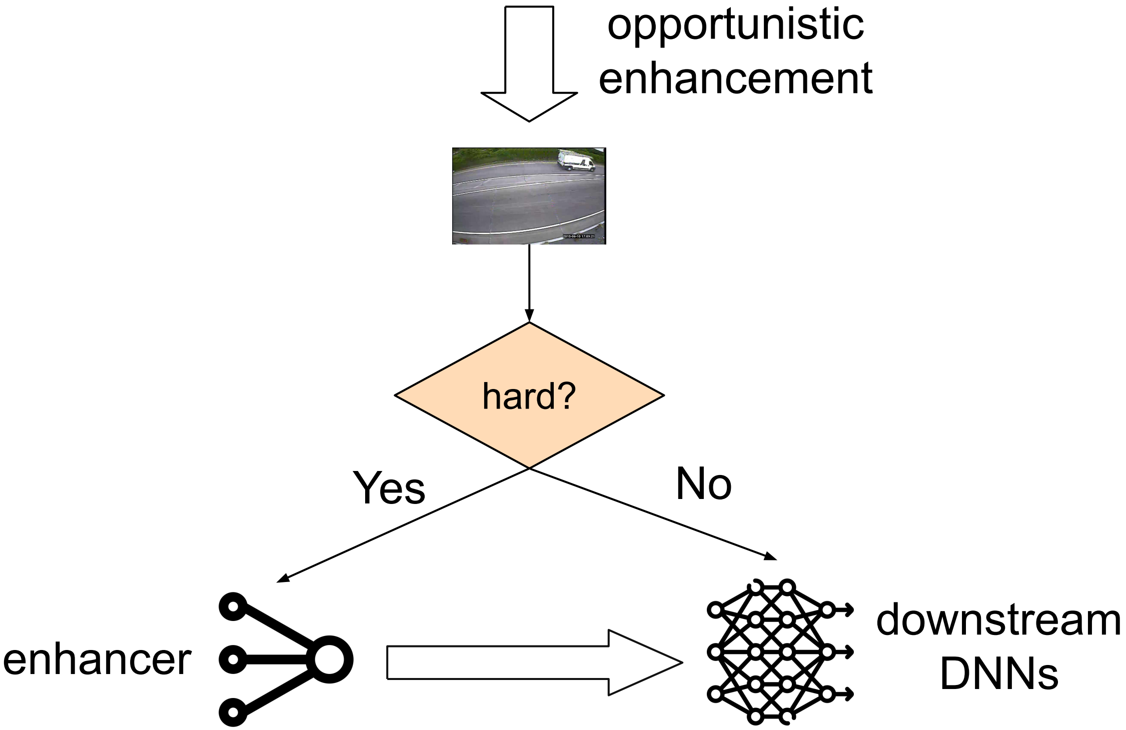
Vigil: 19.03% < 45 infer/sec
Glimpse: 7.26% > 50 infer/sec

It is hard because they are non-deterministic and fragmented!

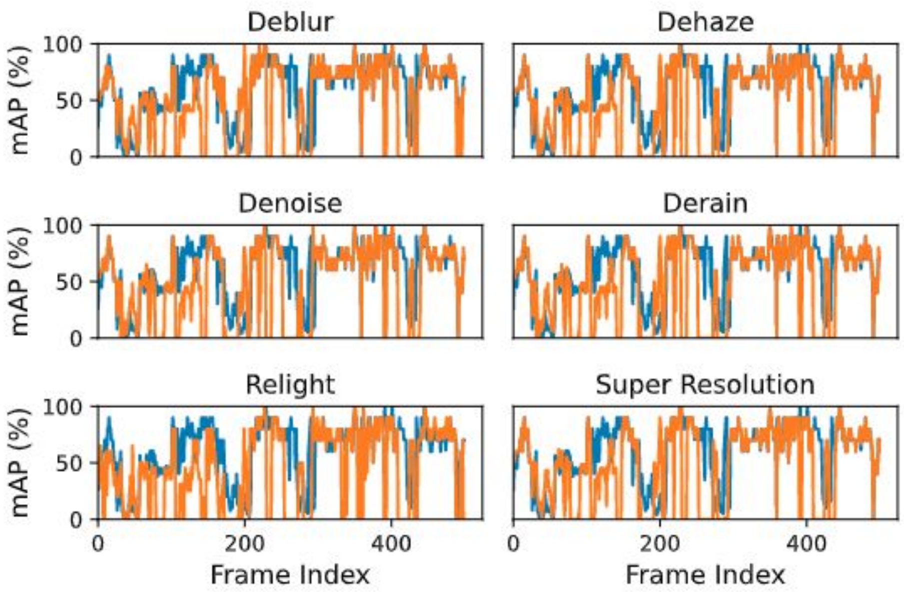
How to leverage idle resources?



A **small portion** of frames make a bad overall mAP for detectors!



How to improve hard samples?



Off-the-shelf
image enhancement may help?

Why they
fail?

Human Visual Perception

≠

Downstream DNNs Accuracy

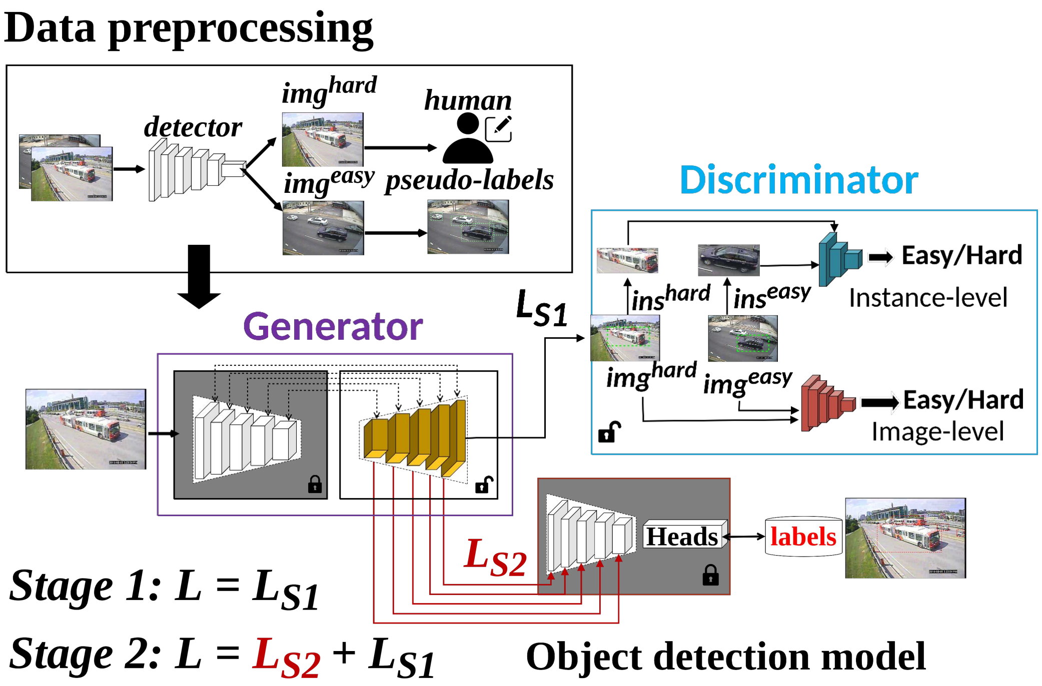
Key takeaways

Idle computing resources are common but highly dynamic and fragmented.

A small portion of hard frames lead to a bad overall accuracy.

Running off-the-shelf opportunistic enhancement methods is inappropriate.

Model-aware Adversarial Training



Stage 0: find easy/hard samples for a downstream detector.

Model-aware easy/hard

Stage 1: learning a G (x) and D (x) for a specific downstream object detection.

Hard -> Generator -> Easy

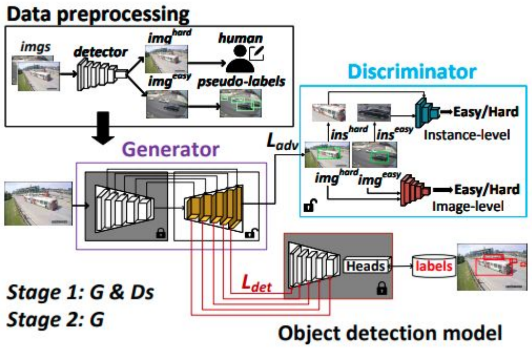
Stage 2: a multi-exit mechanism

Efficient Generator

Pre-training and fast adaptation

BDD100K
(100K driving videos)

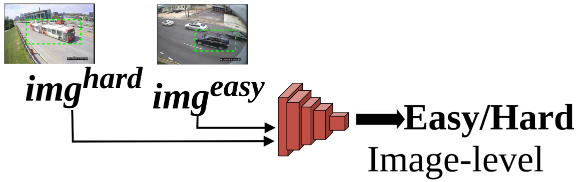
Pre-training



P(Hard) -> Generator -> P(Easy)

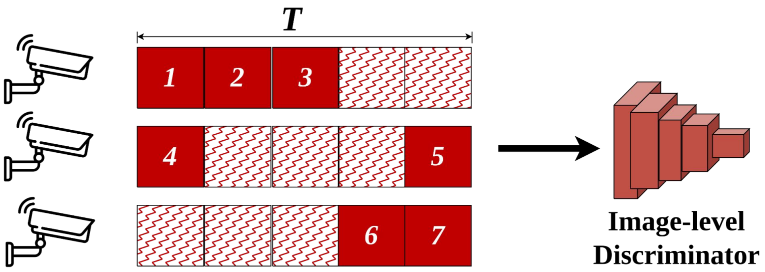
Unlabeled target videos

Unsupervised adaptation

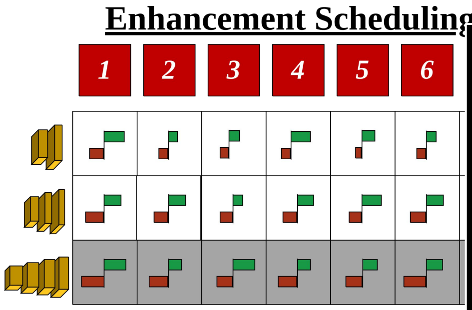


x -> Discriminator -> Easy/Hard

Resource-aware scheduling

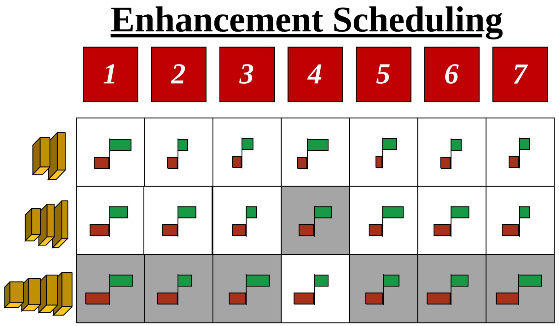


All frames are required to be processed within T.



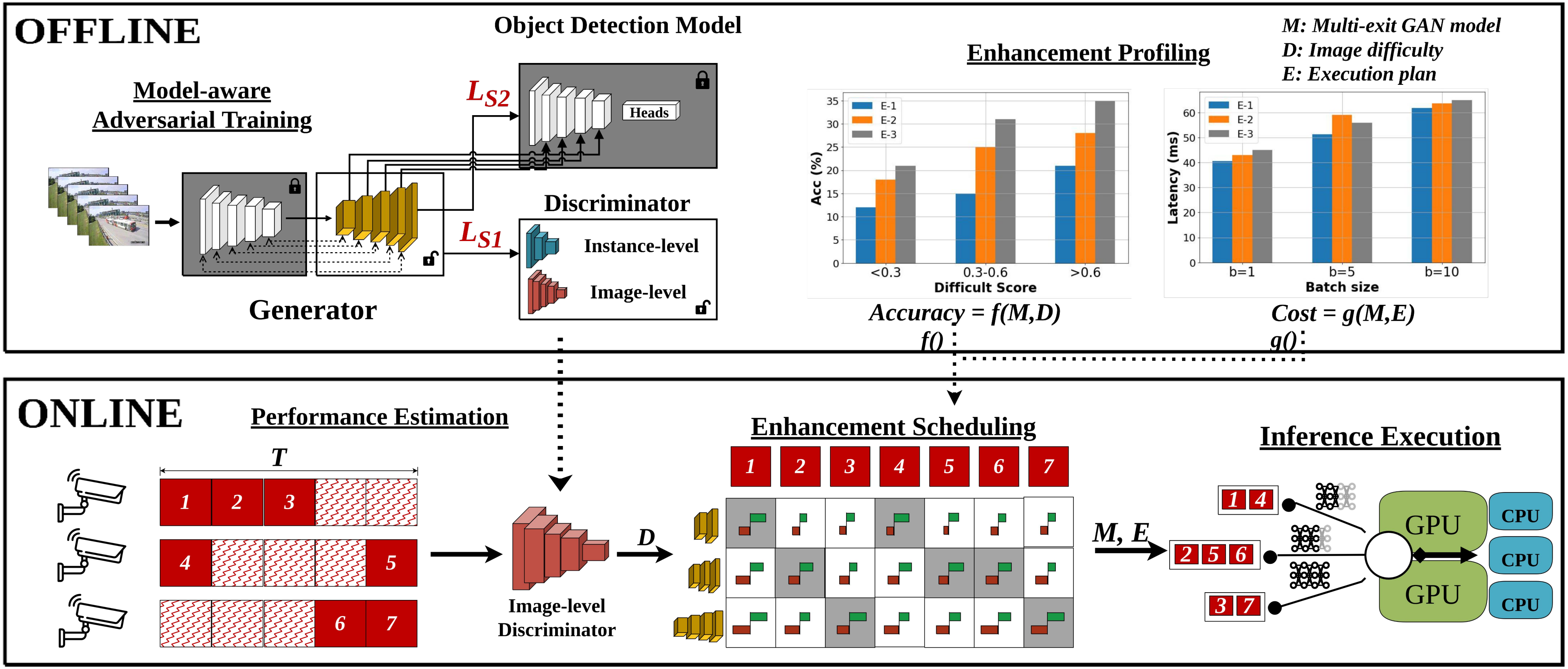
Repeat the last step until the total latency < T

o the deepest enhancer.



Based on enhancement profiling results, we can select a frame with the minimal marginal accuracy gain and assign it to a weaker enhancer.

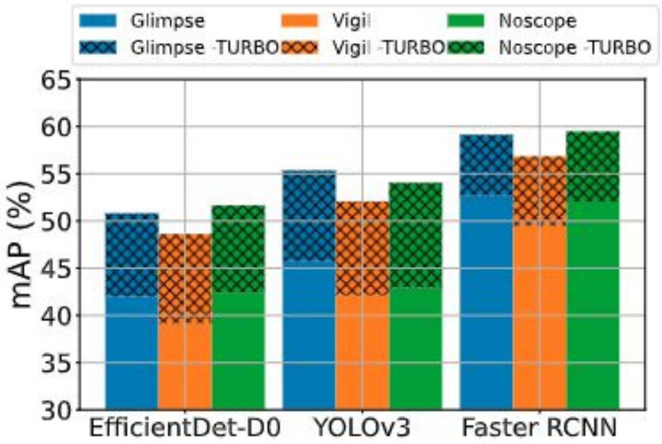
Overview


OFFLINE
Object Detection Model
Model-aware
Adversarial Training
L_{S2}
Heads
L_{S1}
Discriminator
Instance-level
Image-level
Generator
Enhancement Profiling
M: Multi-exit GAN model
D: Image difficulty
E: Execution plan
E-1
E-2
E-3
Acc (%)
<0.3
0.3-0.6
>0.6
Difficult Score
Latency (ms)
b=1
b=5
b=10
Batch size
Accuracy = f(M,D)
f()
Cost = g(M,E)
g()
ONLINE
Performance Estimation
T
1 2 3
4 5
6 7
Image-level
Discriminator
D
Enhancement Scheduling
1 2 3 4 5 6 7
M, E
Inference Execution
1 4
2 5 6
3 7
GPU
GPU
CPU
CPU
CPU

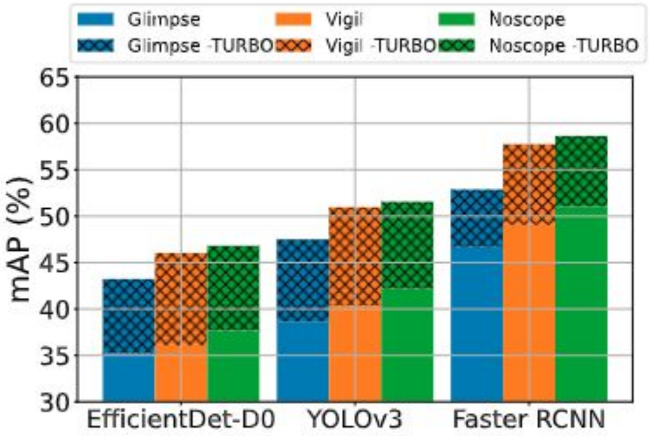
Experiments

- Detectors: YOLOv3, Faster RCNN, EfficientDet-D0.
- Test platforms: Nvidia Tesla V100 and Tesla T4.
- Testing Dataset: UA-DETRAC and AICity.
- Video analytics pipeline:
  - Glimpse: temporal pruning
  - Vigil: model pruning
  - NoScope: temporal pruning + model pruning

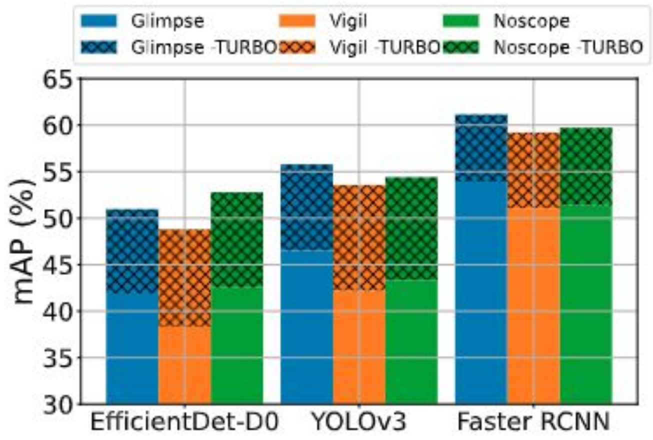
End-to-end results (Accuracy)



UA-DETRAC & T4



AICity & T4

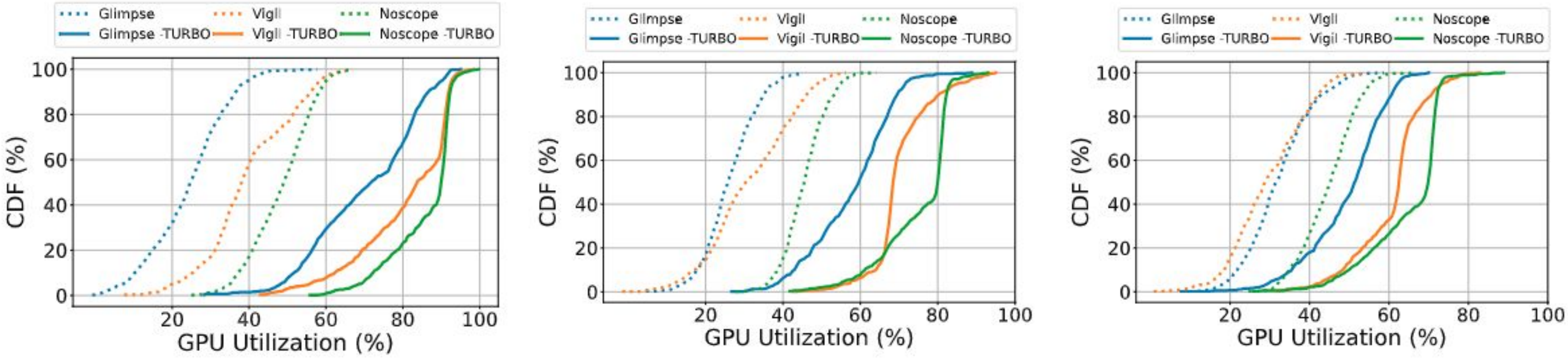


UA-DETRAC & V100

On UA-DETRAC, Turbo achieves 9.35%, 11.34%, 7.27% mAP improvement on average for 3 models.

Usually, we can achieve the maximum mAP improvements on Vigil. It is because model pruning groups most hard frames for Turbo.

End-to-end results (Idle GPU)



(a) EfficientDet-D0

(b) YOLOv3

(c) Faster RCNN

UA-DETRAC & T4

Summary

- Even on advanced video analytics pipelines, idle computing resources are common but ignored.
- Turbo **selectively** enhances incoming frames based GPU resource availability via a **detector-specific GAN** and **resource-aware scheduling** algorithm.
- Turbo achieves 7.27-11.34% mAP improvements by judiciously allocating 15.81-37.67% GPU idle resources.