Efficient Test-time Adaptive Object Detection via Sensitivity-Guided Pruning

Kunyu Wang, Xueyang Fu, Xin Lu, Chengjie Ge, Chengzhi Cao, Wei Zhai, Zheng-Jun Zha†

University of Science and Technology of China





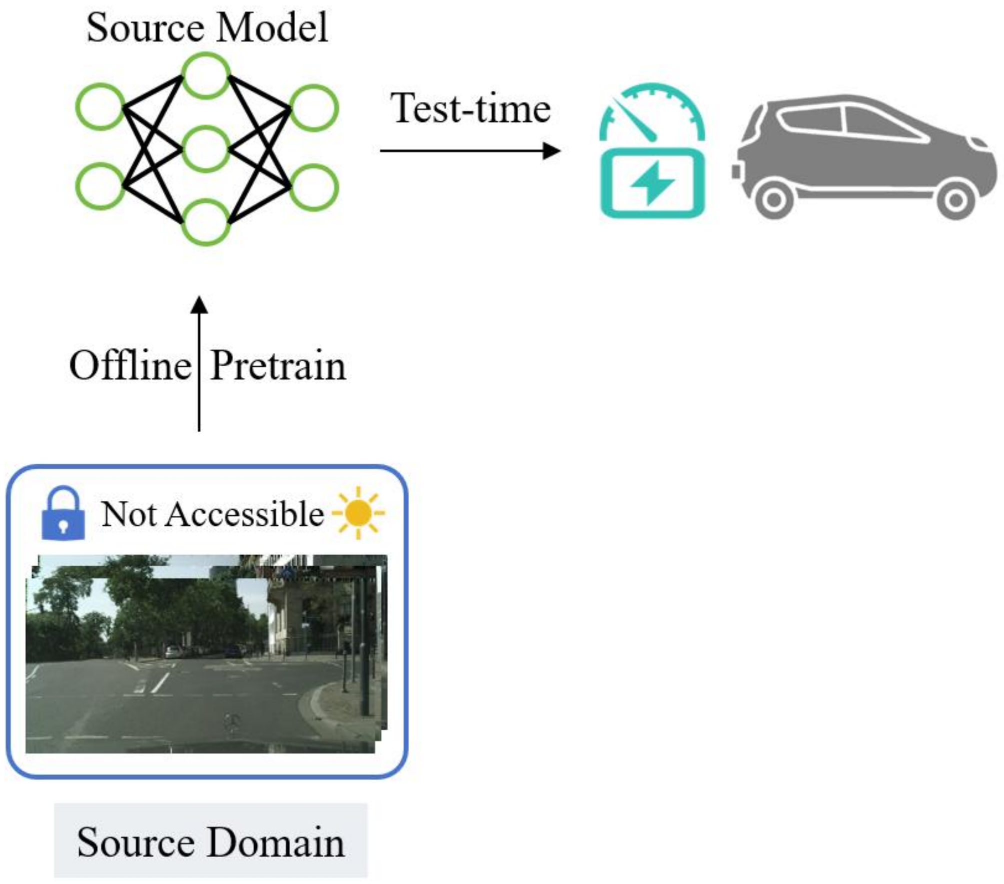


Test-Time Adaptation (TTA)

TTA aims to online adapt a pre-trained model to unlabeled and changing environments during inference, without accessing source data.

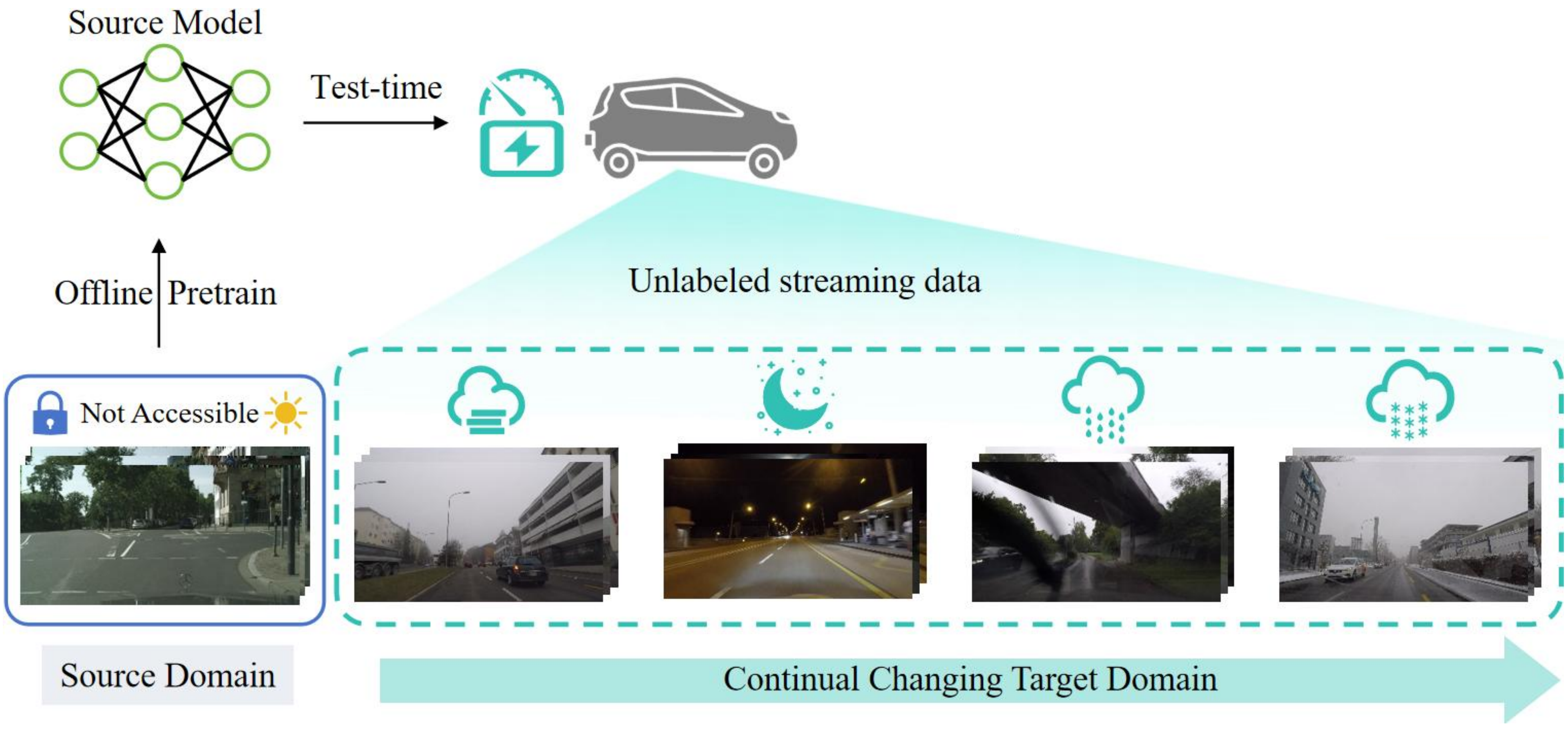
Test-Time Adaptation (TTA)

TTA aims to online adapt a pre-trained model to unlabeled and changing environments during inference, without accessing source data.



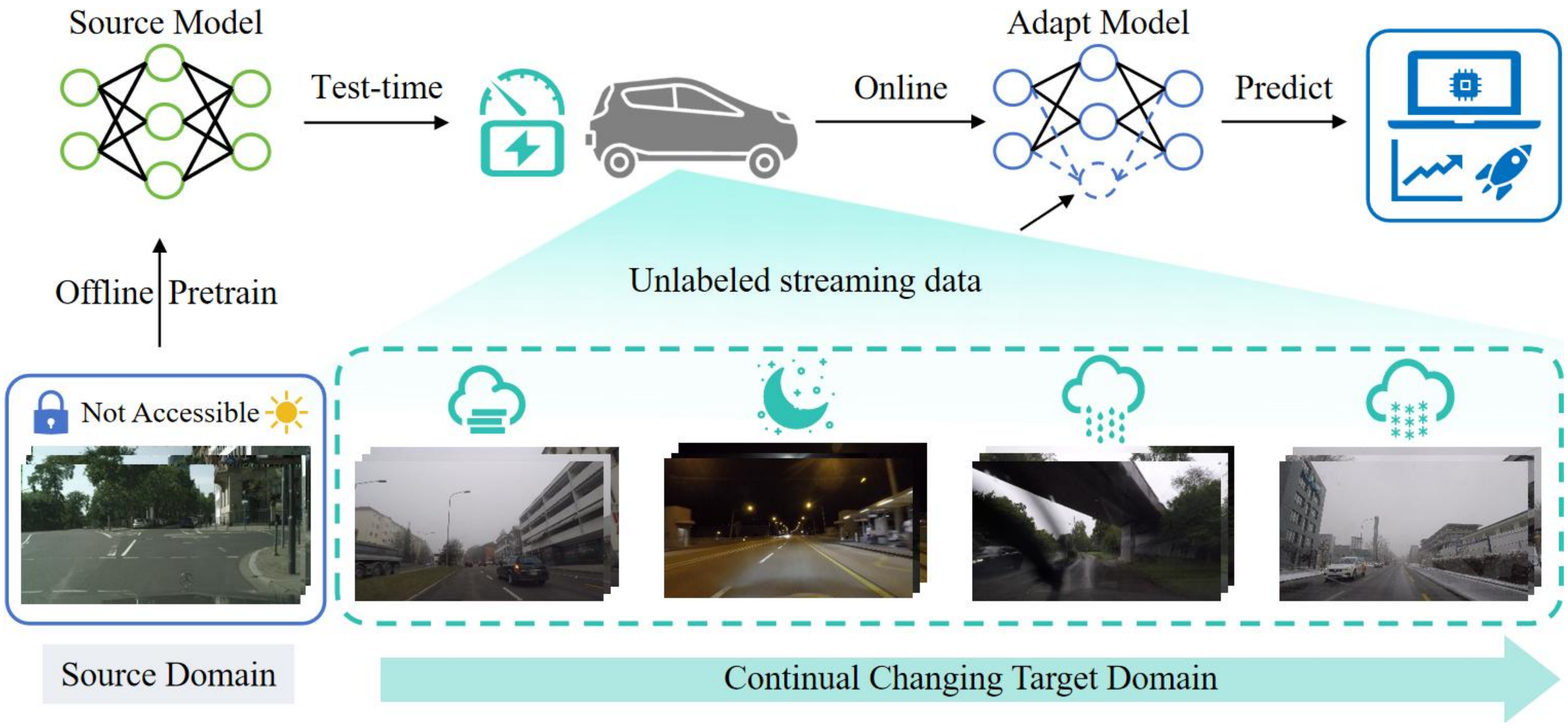
Test-Time Adaptation (TTA)

TTA aims to online adapt a pre-trained model to unlabeled and changing environments during inference, without accessing source data.



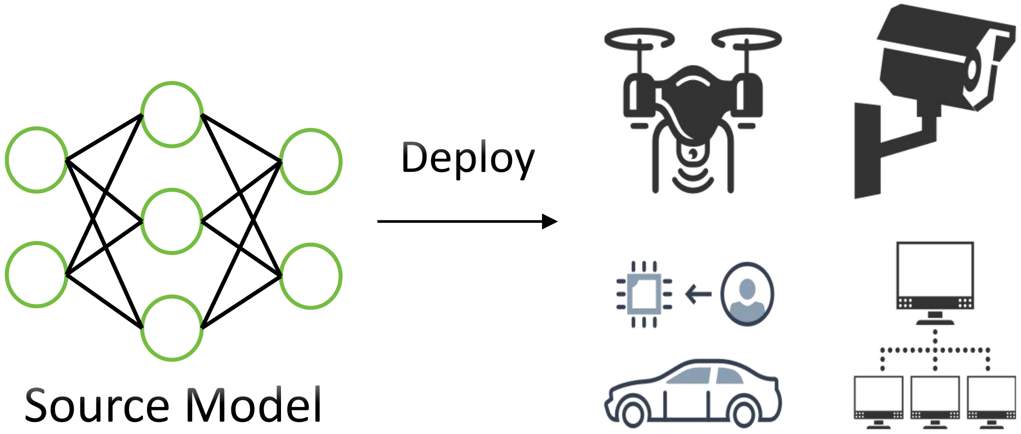
Test-Time Adaptation (TTA)

TTA aims to online adapt a pre-trained model to unlabeled and changing environments during inference, without accessing source data.



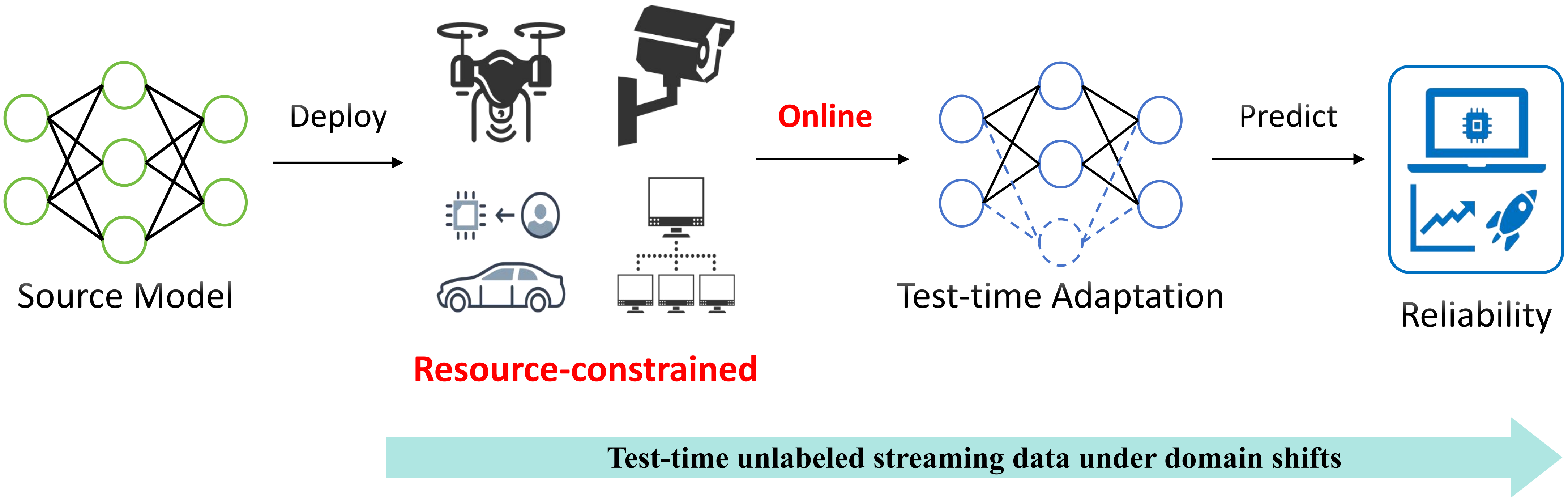
Key Challenge in TTA — Computational Efficiency

Key Challenge in TTA — Computational Efficiency

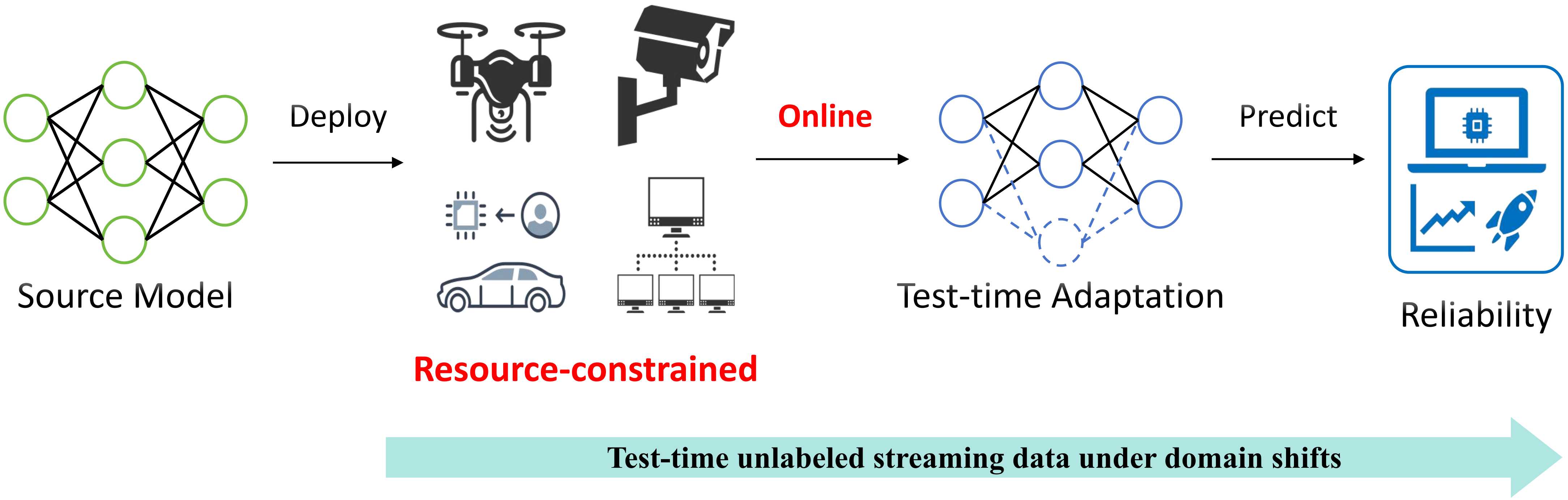


Resource-constrained

Key Challenge in TTA — Computational Efficiency



Key Challenge in TTA — Computational Efficiency



How can we adapt efficiently?

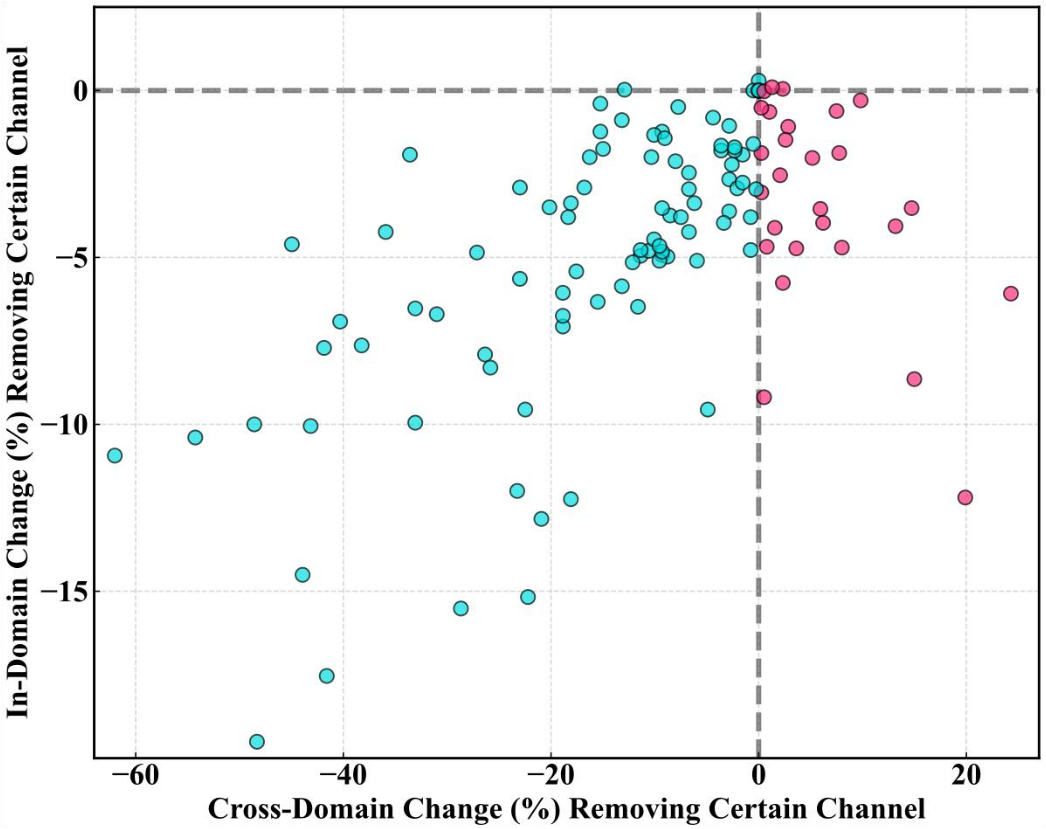
Key Insight – Selective Transfer of Source Knowledge

- During source-to-target transfer, **not all source-learned features are beneficial**, some can even degrade target performance.

Key Insight – Selective Transfer of Source Knowledge

- During source-to-target transfer, **not all source-learned features are beneficial**, some can even degrade target performance.

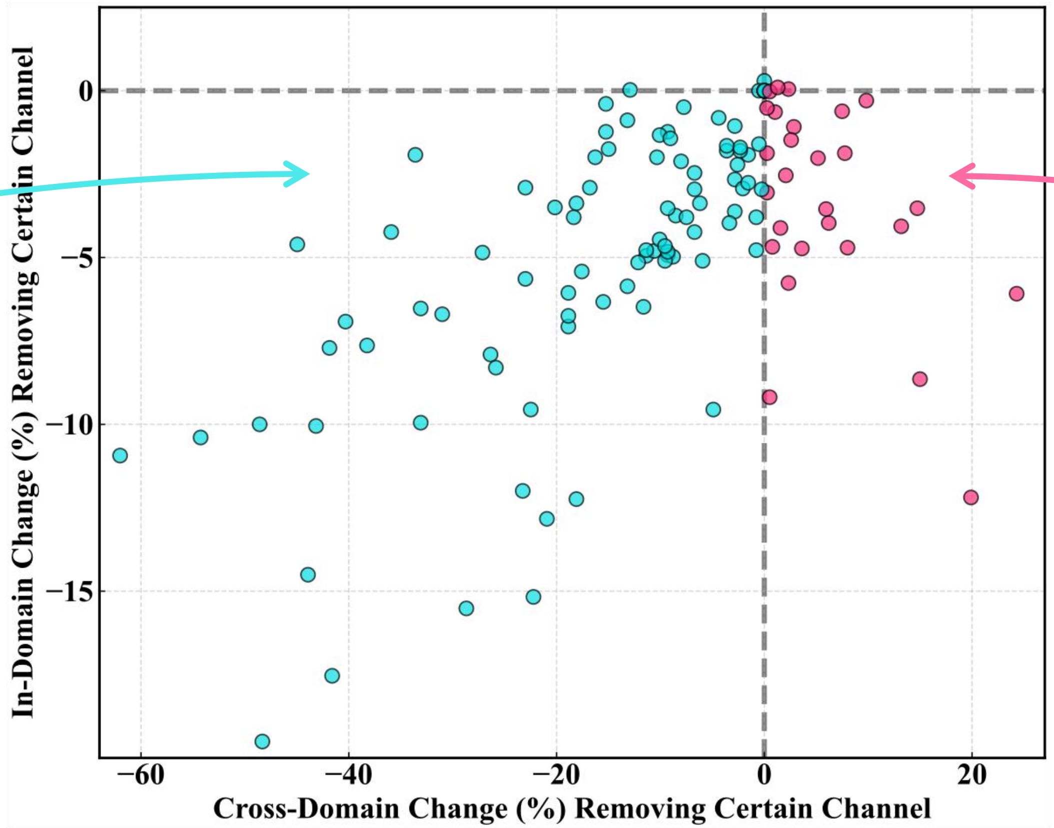
How removing certain source feature channels affects performance?



Key Insight – Selective Transfer of Source Knowledge

- During source-to-target transfer, **not all source-learned features are beneficial**, some can even degrade target performance.

How removing certain source feature channels affects performance?



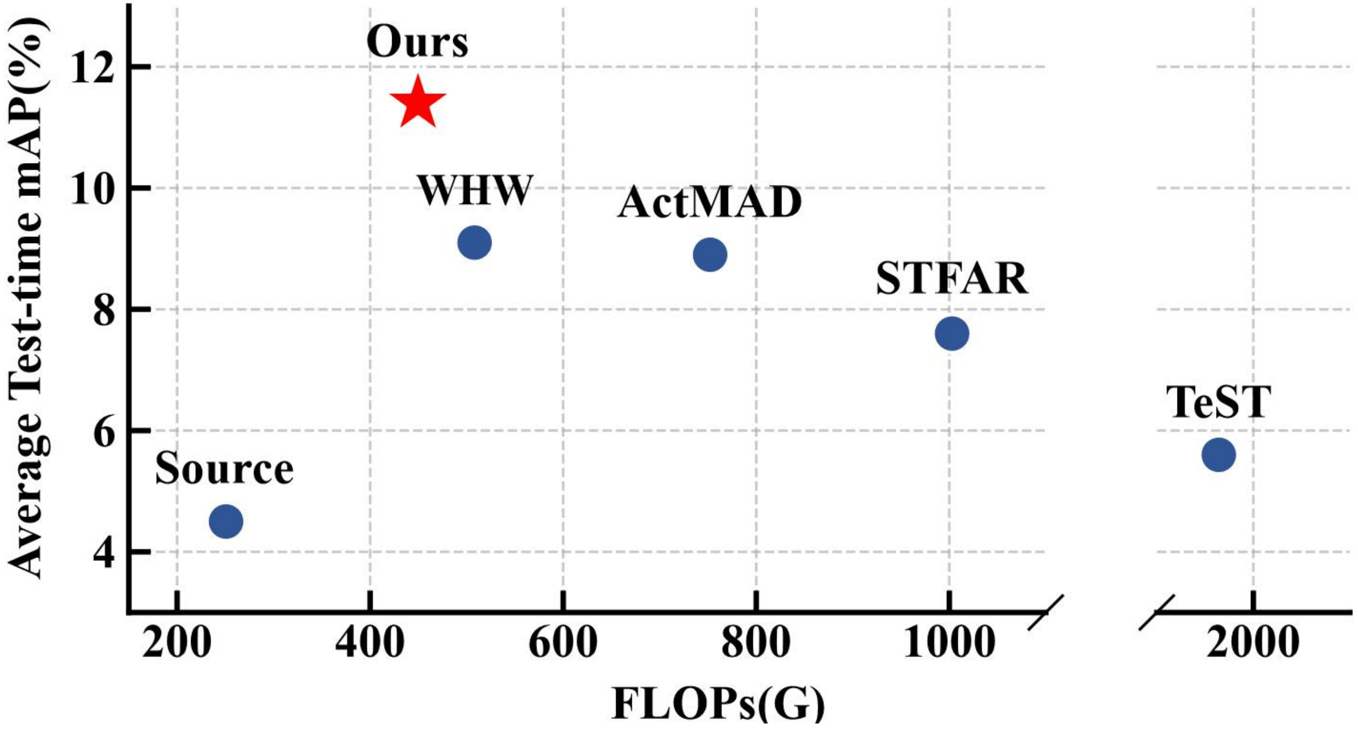
Their removal hurts both source and target performance.

Their removal hurts the source but helps the target.

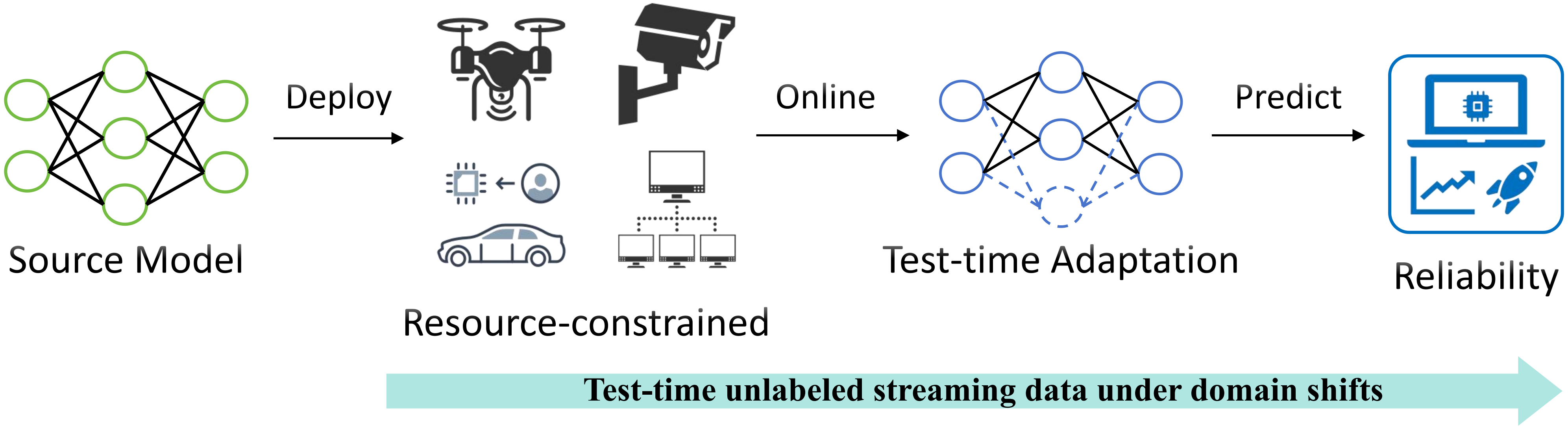
Key Contribution

- Inspired by this insight, we propose that TTA should focus on the helpful parts of the source model while ignoring the harmful ones.

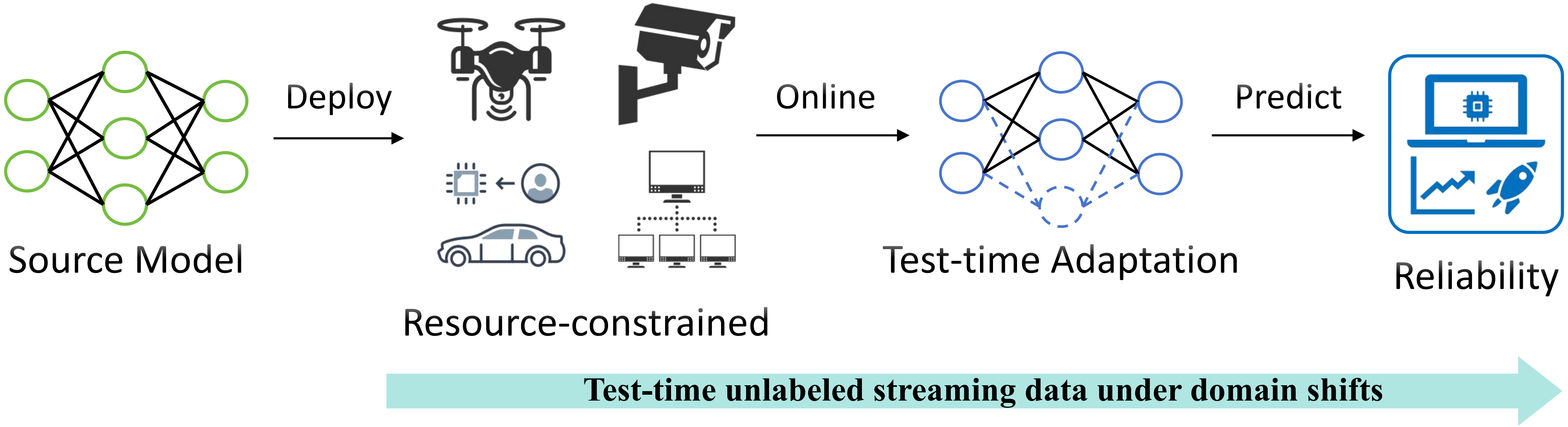
- This not only reduces adaptation difficulty, but also improves adaptation efficiency.



Related Works — Computational Efficiency



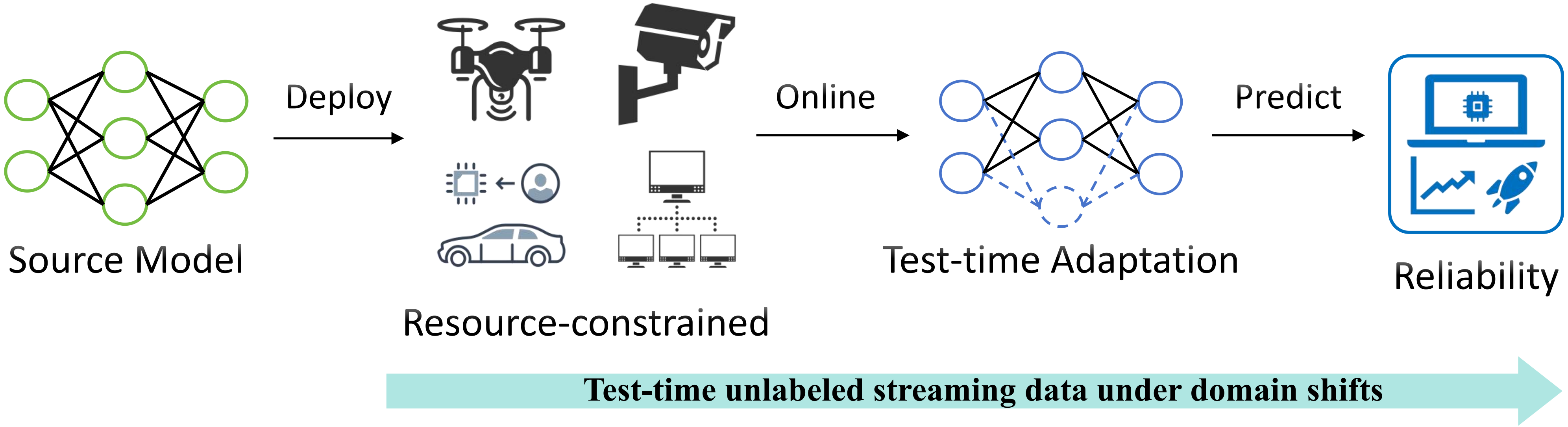
Related Works — Computational Efficiency



Data-level (e.g., EATA: Niu et al. 2022)

- Filter uncertain samples for selective updates

Related Works — Computational Efficiency



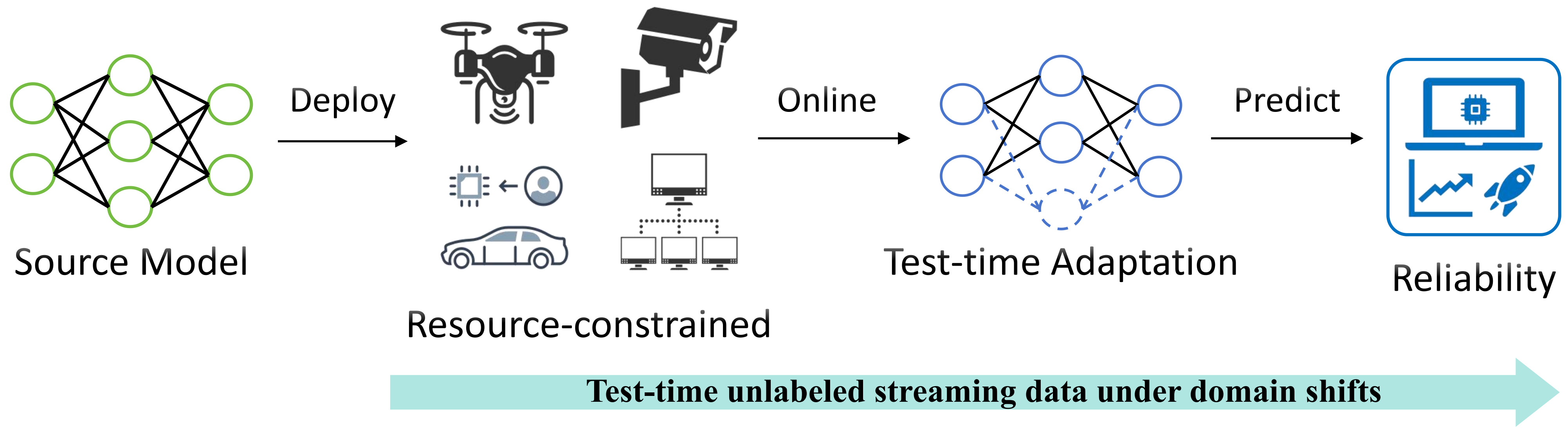
Data-level (e.g., EATA: Niu et al. 2022)

- Filter uncertain samples for selective updates

Parameter-level (e.g., EcoTTA: Song et al. 2023)

- Update minimal, lightweight parameters

Related Works — Computational Efficiency



Data-level (e.g., EATA: Niu et al. 2022)

- Filter uncertain samples for selective updates

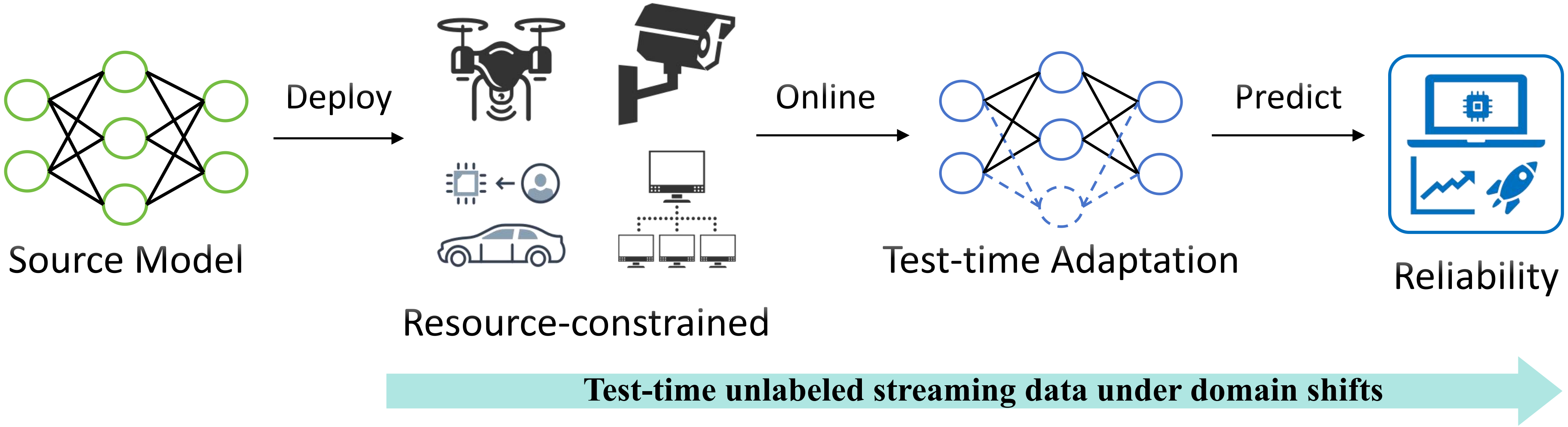
Parameter-level (e.g., EcoTTA: Song et al. 2023)

- Update minimal, lightweight parameters

Scheduling-level (e.g., TinyTTA: Jia et al. 2024)

- Early-exit inference routing

Related Works — Computational Efficiency



Data-level (e.g., EATA: Niu et al. 2022)

- Filter uncertain samples for selective updates

Parameter-level (e.g., EcoTTA: Song et al. 2023)

- Update minimal, lightweight parameters

Scheduling-level (e.g., TinyTTA: Jia et al. 2024)

- Early-exit inference routing

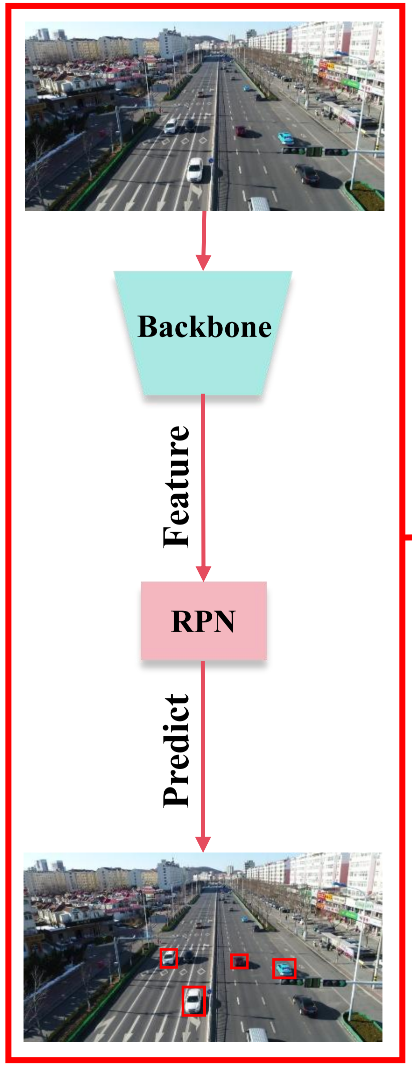
Model structure-level (This work)

- Prune domain-sensitive feature channels

Methodology

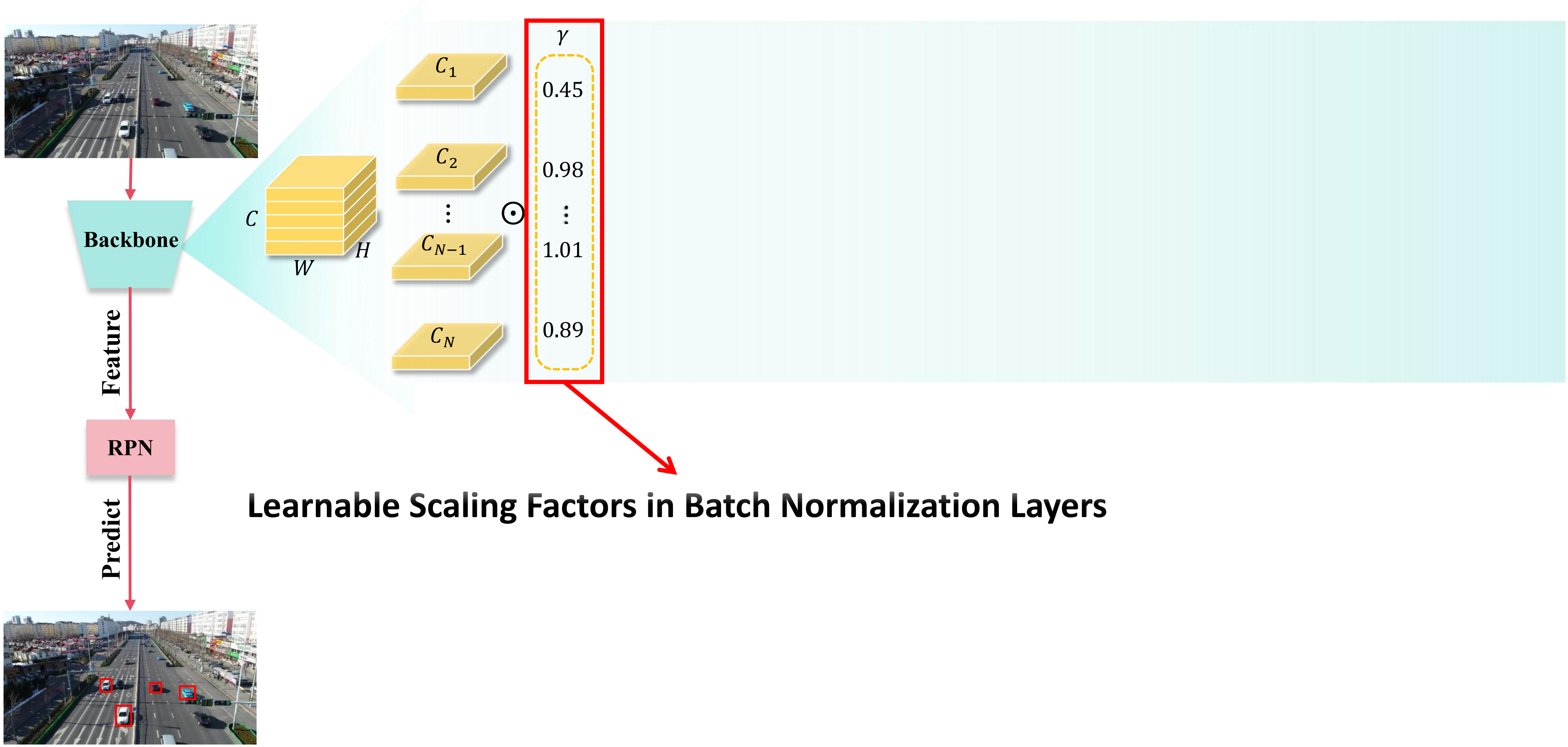
Make test-time adaptation more efficient by updating only the useful parts of the model

Methodology



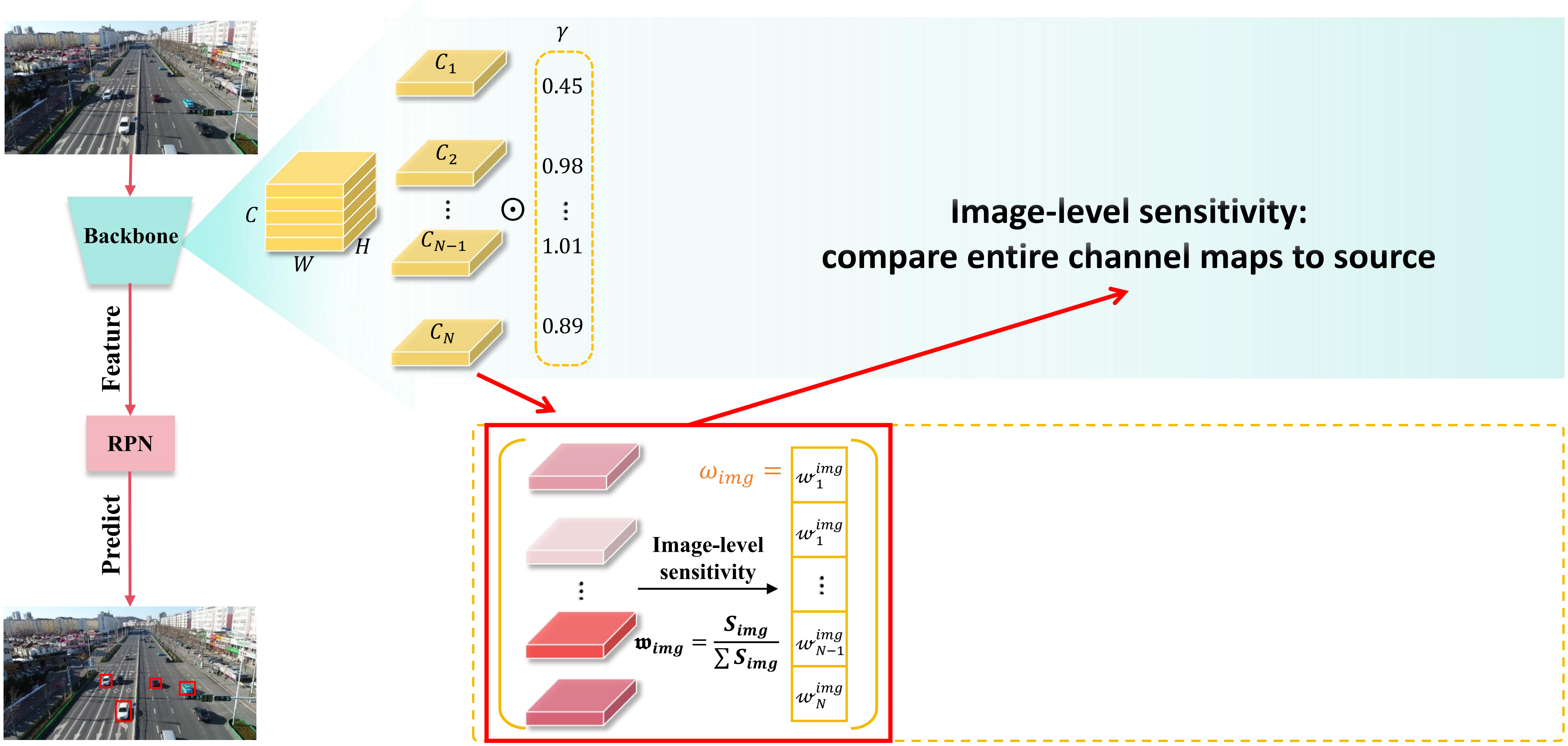
Detection Task
(Faster R-CNN)

Methodology

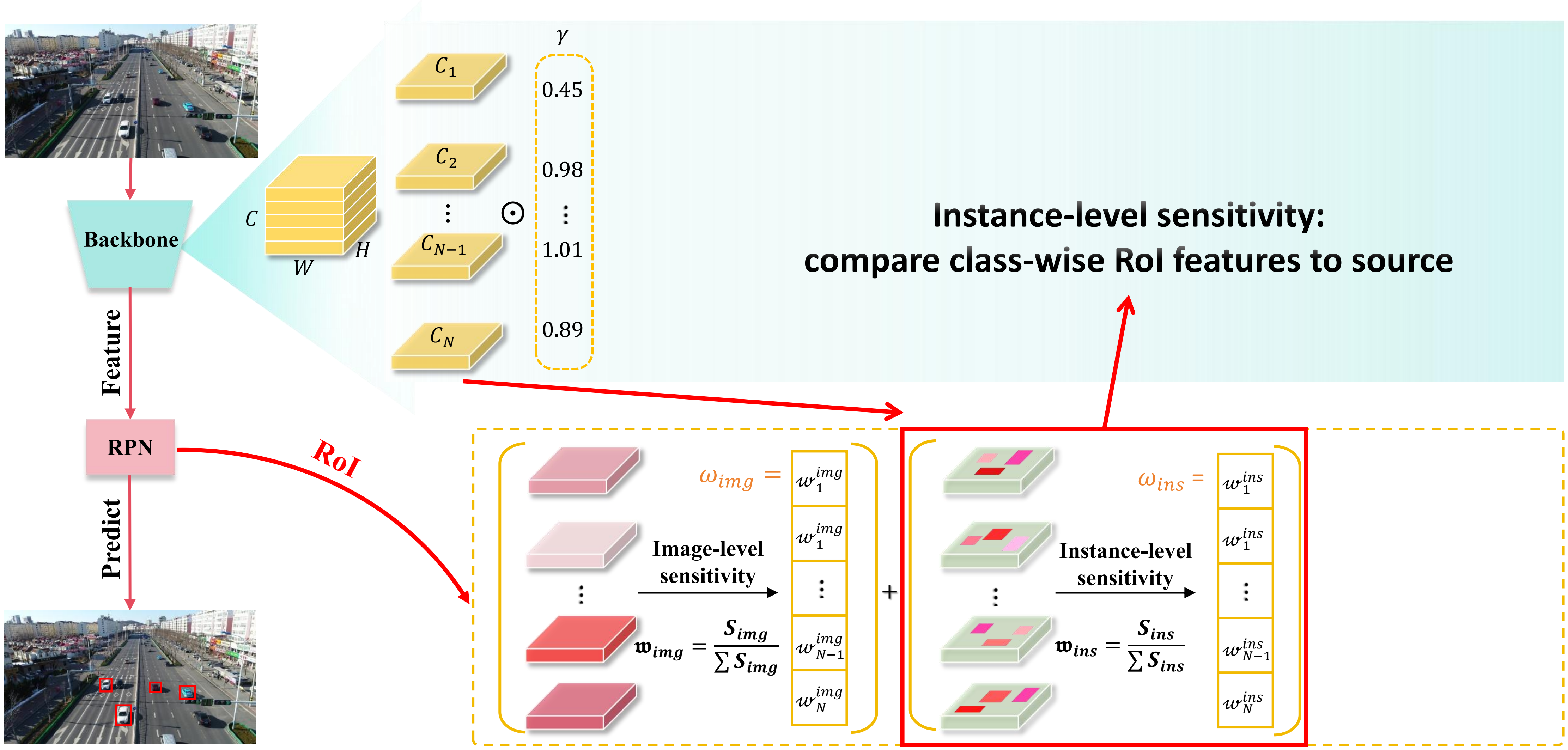


Learnable Scaling Factors in Batch Normalization Layers

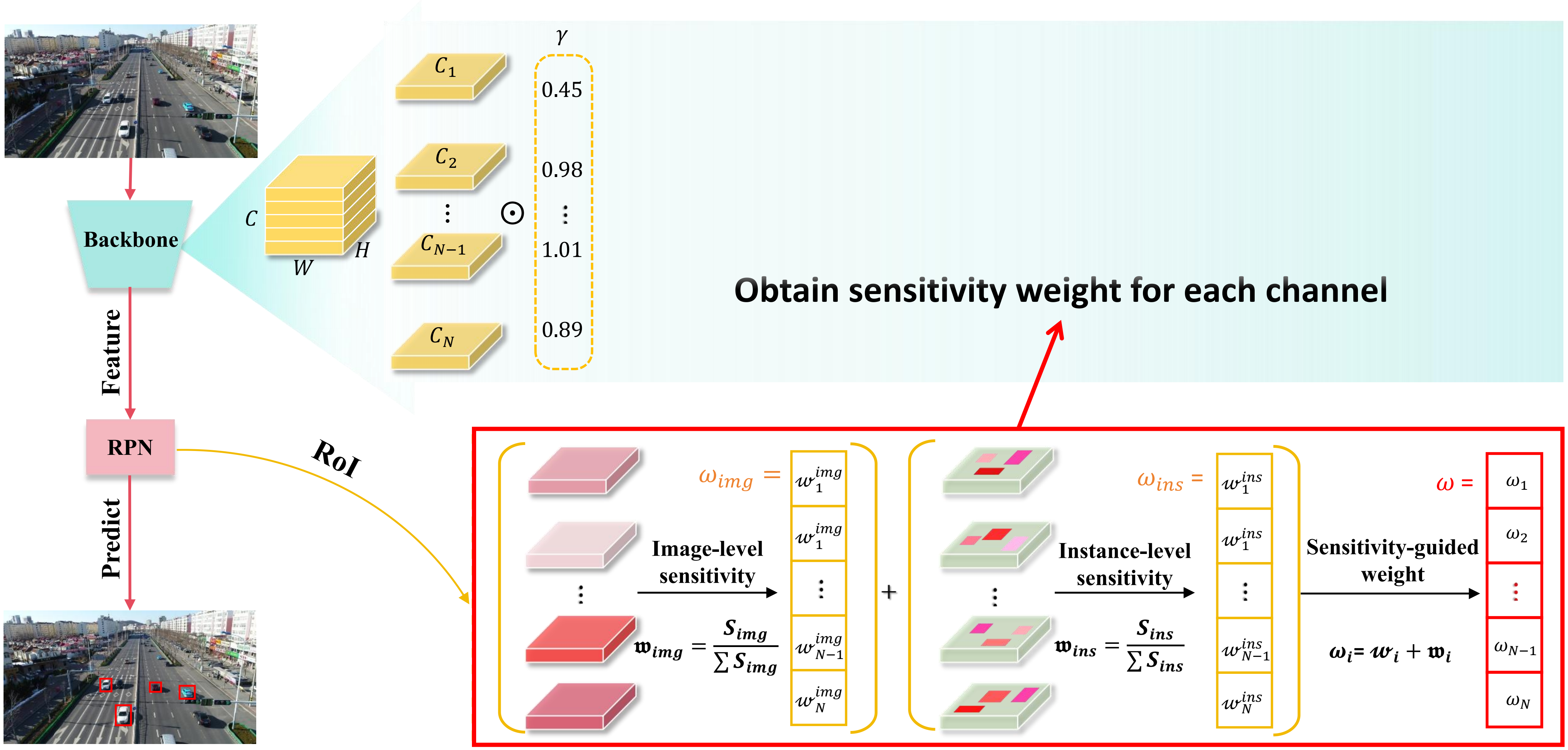
Methodology



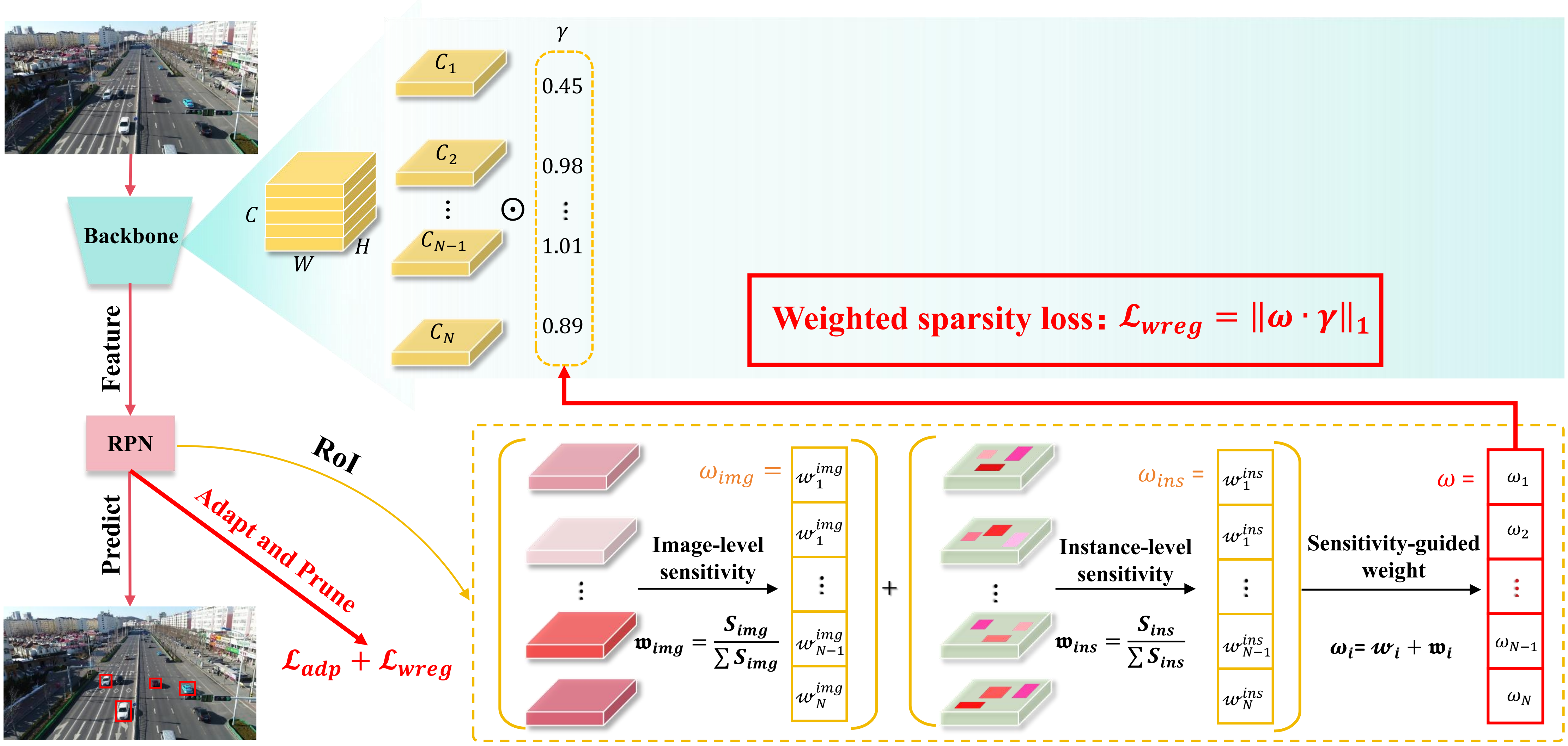
Methodology



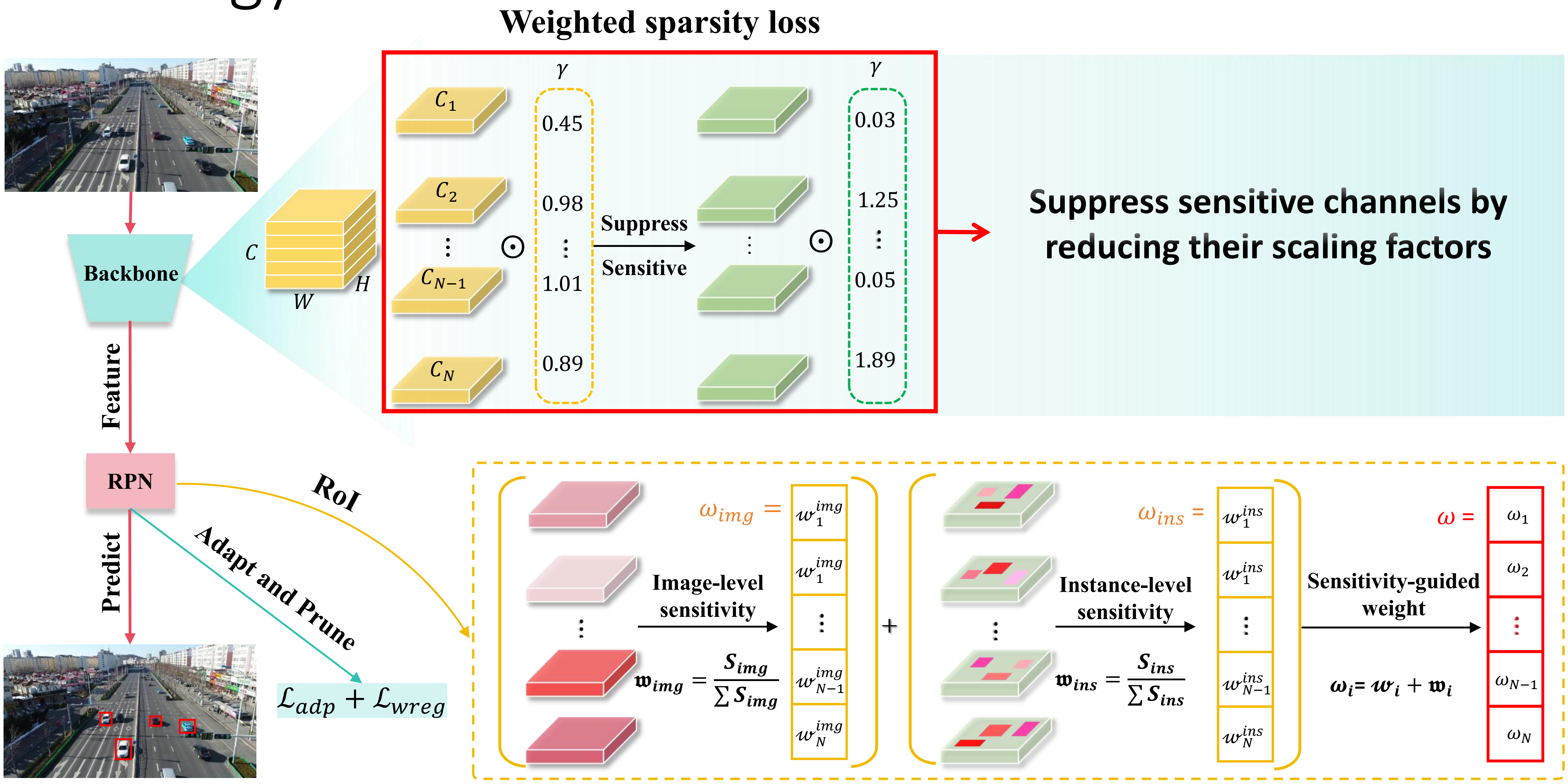
Methodology



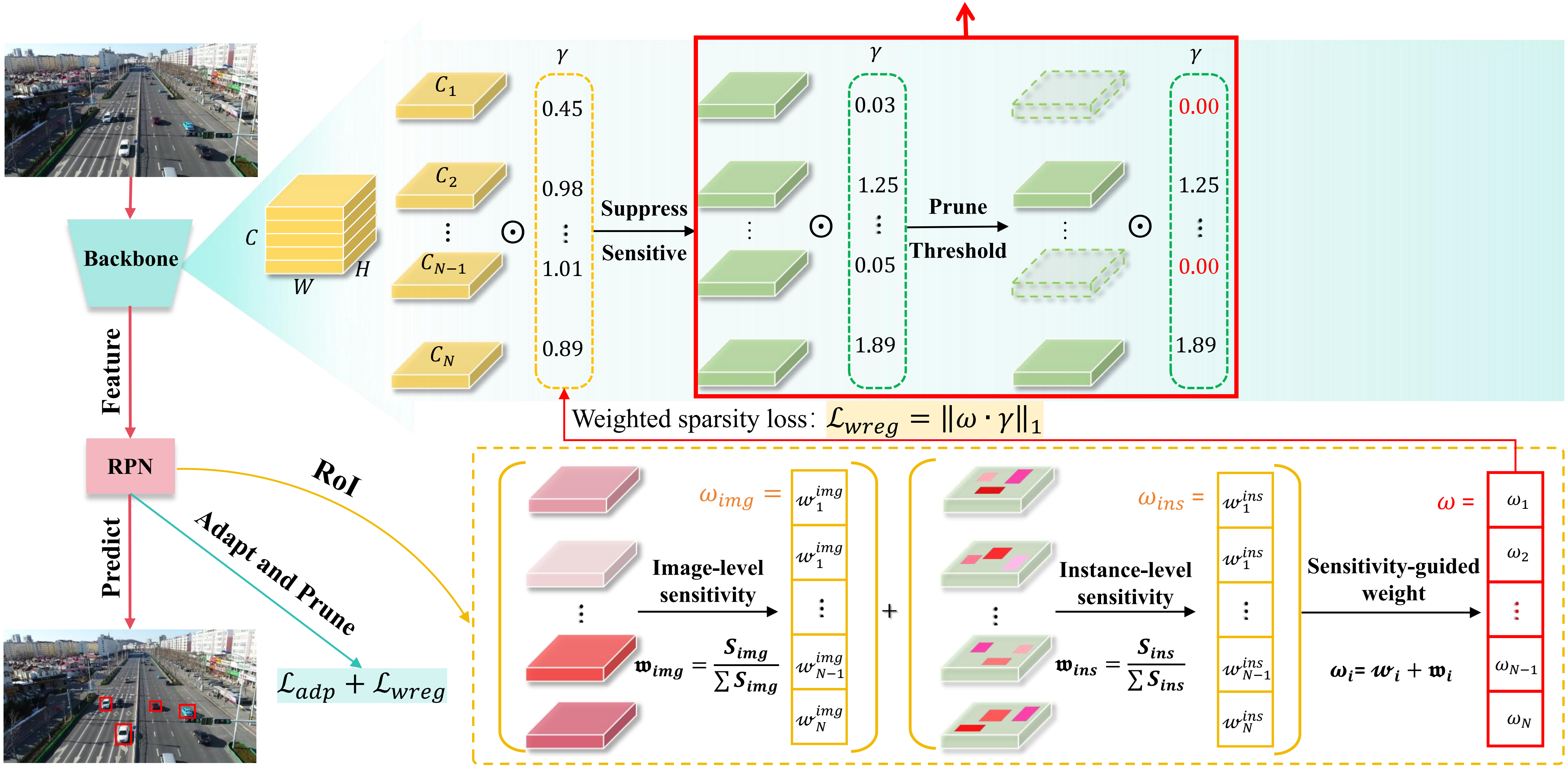
Methodology



Methodology

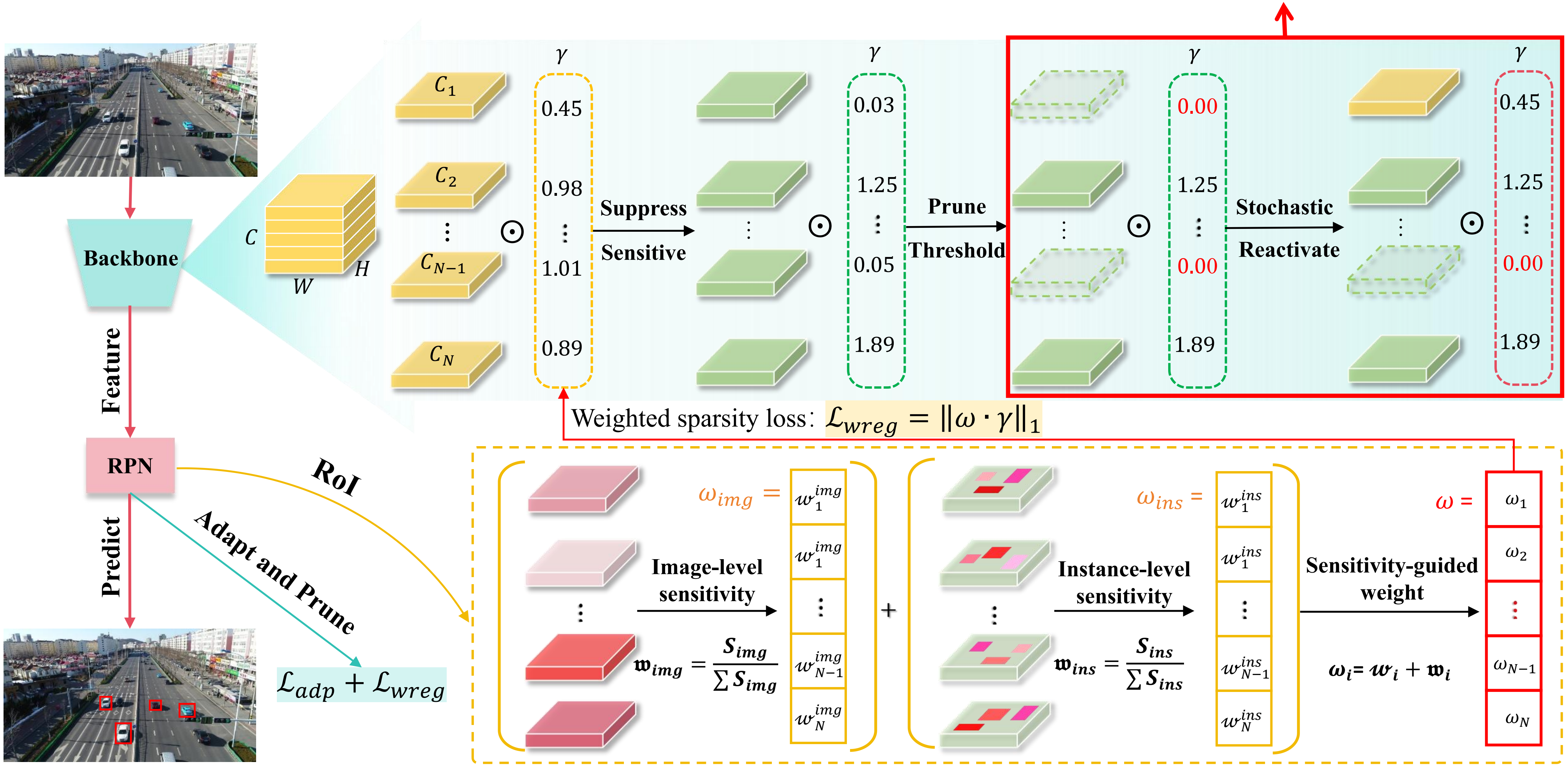


Methodology



Methodology

Reactivate channels to reassess their utility in different domains



Experiments — Settings

Synthetic Benchmark (i.e., Cityscapes → Cityscapes-C, UAVDT → UAVDT-C)

Fog

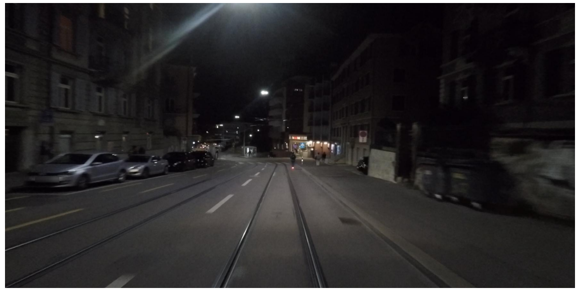
Motion blur

Contrast

Defocus blur

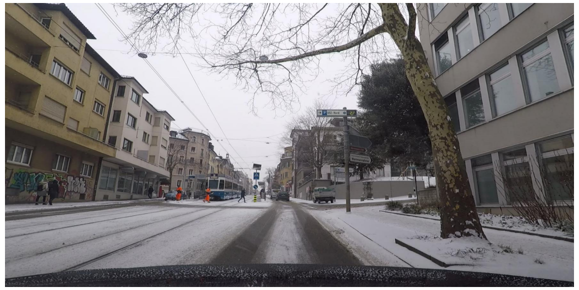
Real-world Benchmark (i.e., Cityscapes → ACDC)

Fog



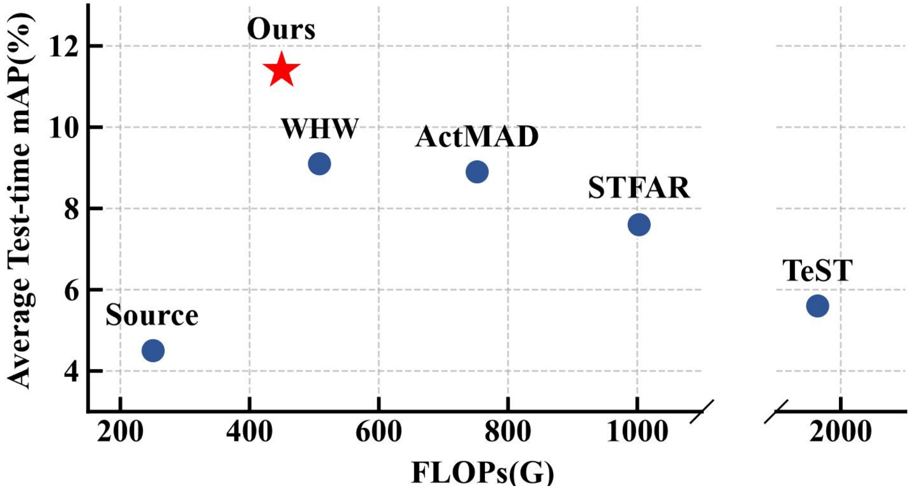
Night

Rain

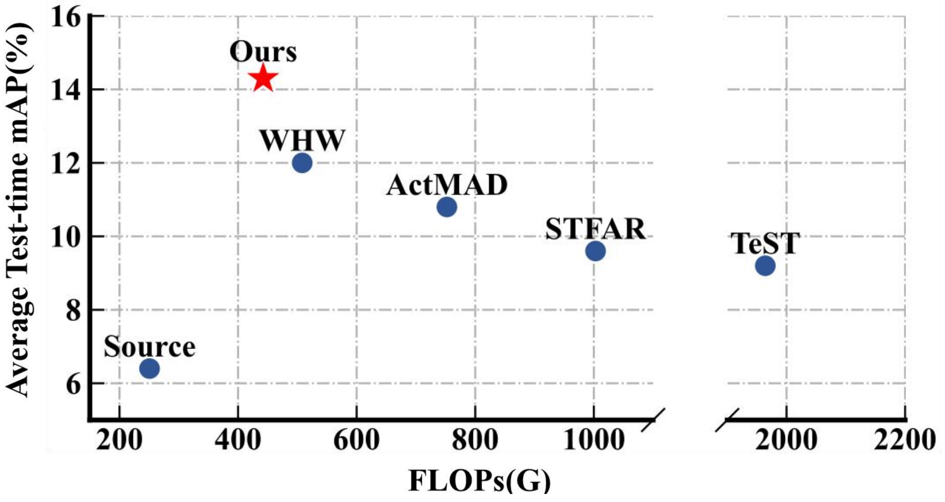


Snow

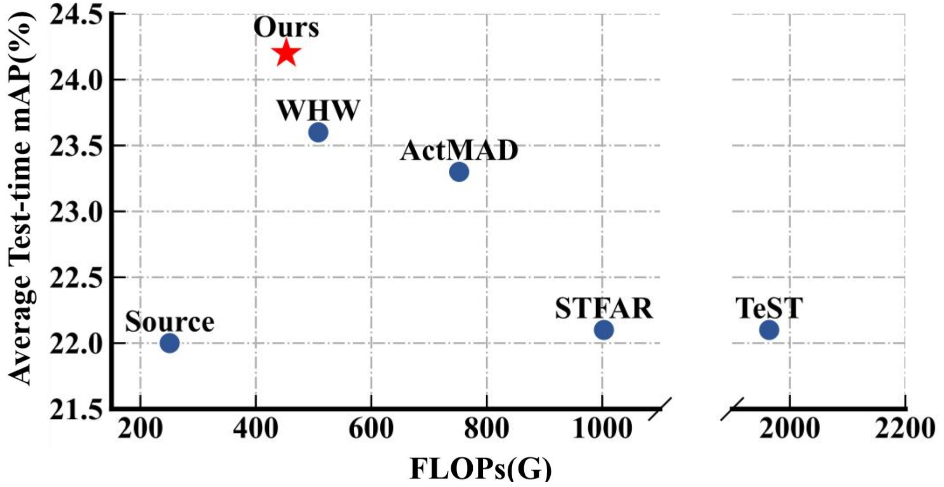
Experiments — Benchmark Results



Cityscapes → Cityscapes-C



UAVDT → UAVDT-C



Cityscapse → ACDC

10-Round Average

mAP@50 (%) and FLOPs (G)

Experiments — Ablation and Visualization

$\mathcal{L}_{adp}$	$\mathcal{L}_{reg}$	ω_{img}	ω_{ins}	SCR	Avg	ΔFLOPs↓
					4.5	0.0%
✓					10.9	0.0%
✓	✓				8.8	10.2%
✓	✓	✓			10.5	10.5%
✓	✓		✓		9.9	10.9%
✓	✓	✓	✓		11.2	10.7%
✓	✓	✓	✓	✓	11.4	10.4%

Framework Component Ablation

Experiments — Ablation and Visualization

$\mathcal{L}_{adp}$	$\mathcal{L}_{reg}$	ω_{img}	ω_{ins}	SCR	Avg	ΔFLOPs↓
					4.5	0.0%
✓					10.9	0.0%
✓	✓				8.8	10.2%
✓	✓	✓			10.5	10.5%
✓	✓		✓		9.9	10.9%
✓	✓	✓	✓		11.2	10.7%
✓	✓	✓	✓	✓	11.4	10.4%

Framework Component Ablation

Experiments — Ablation and Visualization

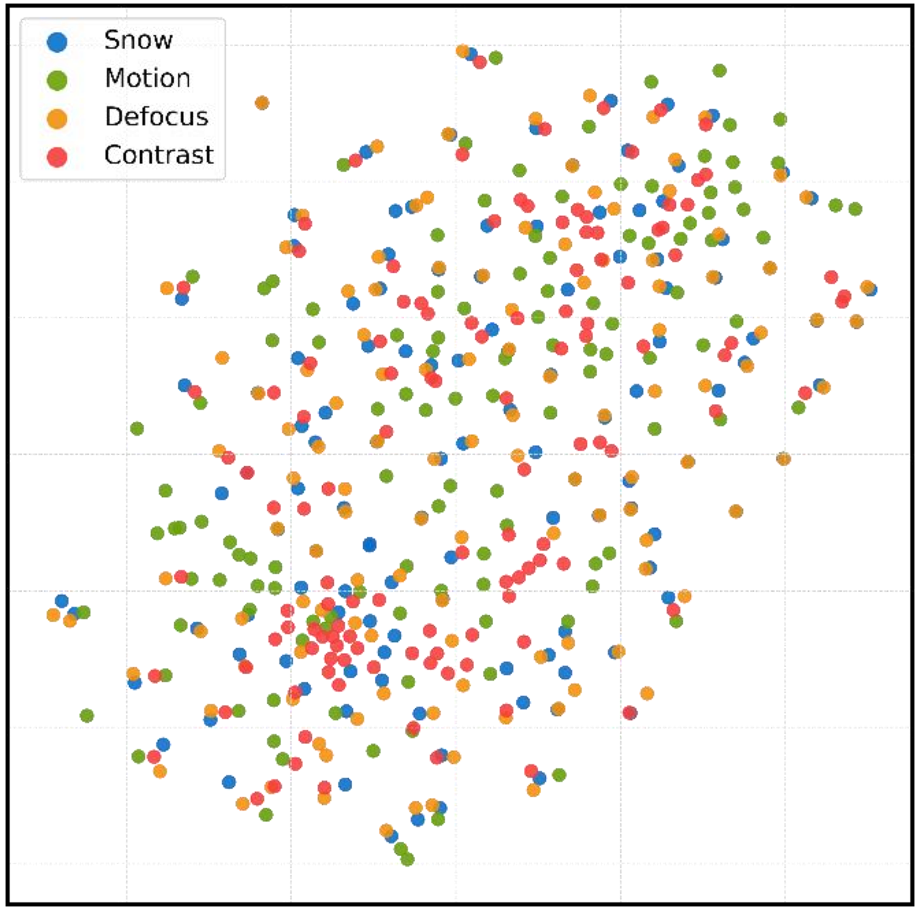
$\mathcal{L}_{adp}$	$\mathcal{L}_{reg}$	ω_{img}	ω_{ins}	SCR	Avg	ΔFLOPs↓
					4.5	0.0%
✓					10.9	0.0%
✓	✓				8.8	10.2%
✓	✓	✓			10.5	10.5%
✓	✓		✓		9.9	10.9%
✓	✓	✓	✓		11.2	10.7%
✓	✓	✓	✓	✓	11.4	10.4%

Framework Component Ablation

Experiments — Ablation and Visualization

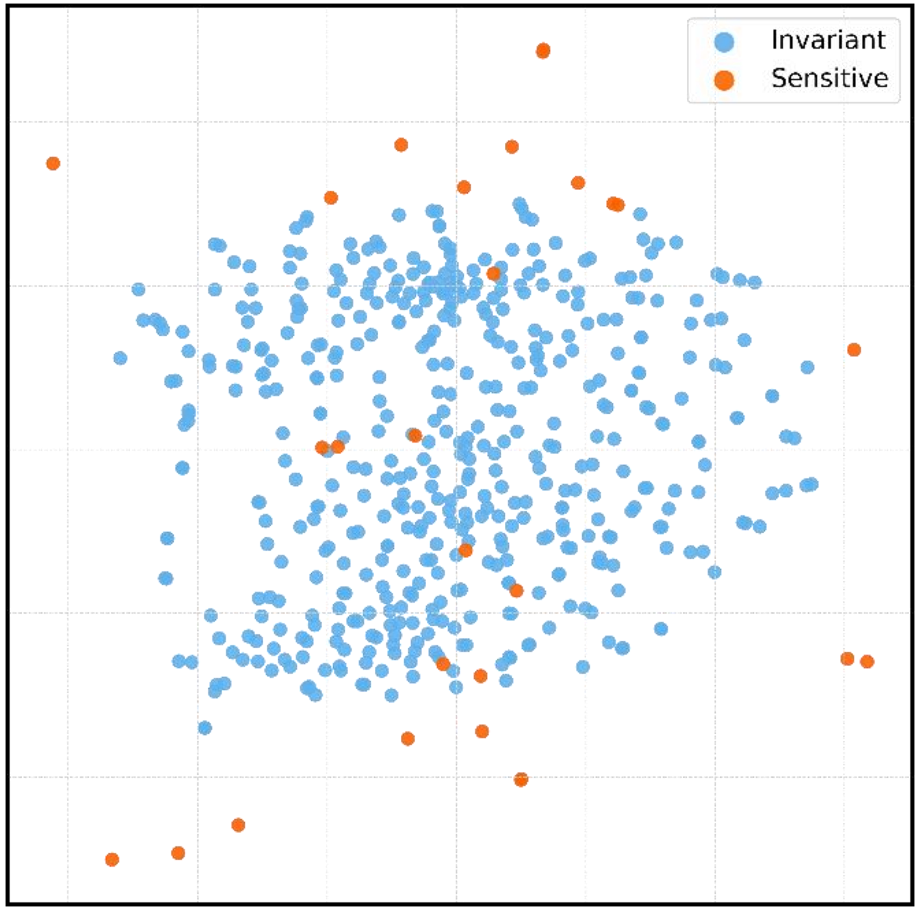
$\mathcal{L}_{adp}$	$\mathcal{L}_{reg}$	ω_{img}	ω_{ins}	SCR	Avg	ΔFLOPs↓
					4.5	0.0%
✓					10.9	0.0%
✓	✓				8.8	10.2%
✓	✓	✓			10.5	10.5%
✓	✓		✓		9.9	10.9%
✓	✓	✓	✓		11.2	10.7%
✓	✓	✓	✓	✓	11.4	10.4%

Framework Component Ablation



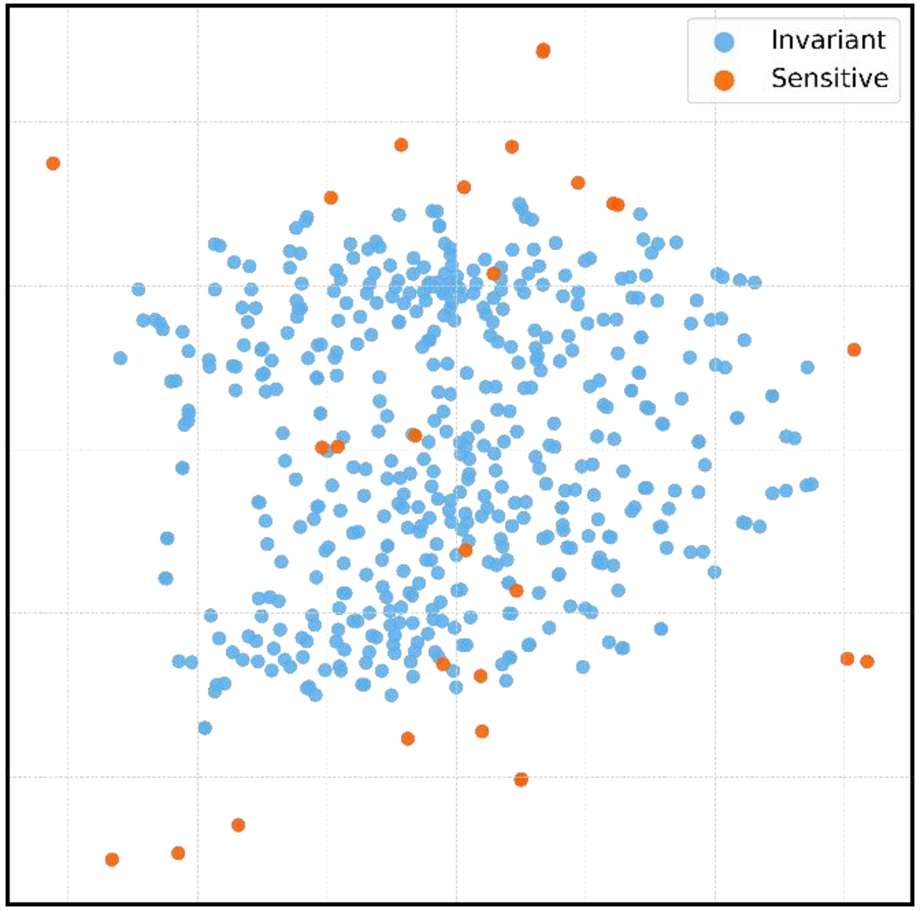
T-SNE of Invariant Channels
During Final Round of Domain-wise Adaptation

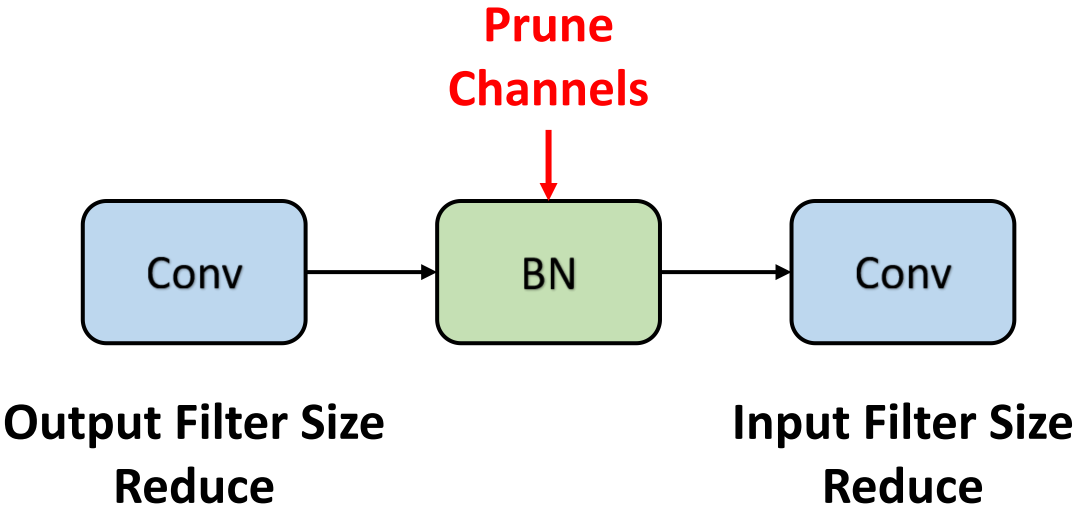
Experiments — Ablation and Visualization



T-SNE of Feature Channels on a Certain Target Domain
with Full Channel Update, No pruning
during Final Round of Adaptation
Differentiated by the Learned Channel Mask

Experiments — Ablation and Visualization





Thanks for your attention

Poster Session: 14 Jun (today), 10:30 a.m. CDT — 12:30 p.m,
ExHall D Poster #419

Contact me at kunyuwang@mail.ustc.edu.cn.

Q&A Session

1. Why did you choose to operate on BN layers?
2. How exactly are the channels pruned? Why does setting them to zero work?
3. Would lightweight detectors like YOLO still have redundant channels?
4. What about detectors without BN layers, like ViT or DETR?
5. How did you choose the hyper-parameters?
6. In your exploratory experiment, how did you identify “sensitive” channels without labels?
7. Are you assuming that the pre-trained model works reasonably well in the target domain?
8. How is your method different from standard test-time pruning or dynamic sparsity methods?
9. How often do you prune during test-time adaptation? Is it done every step?

Q&A Session

10、Could the method be unstable if the sensitivity scores are noisy or unreliable?

11、Is your pruning strategy task-agnostic? Could it be applied to classification or segmentation?

12、Does pruning ever hurt the performance on easy or clean target samples?

13、Could you incorporate uncertainty or confidence scores into the pruning decision?

14、Why do you not retrain or fine-tune the pruned model? Would that help?