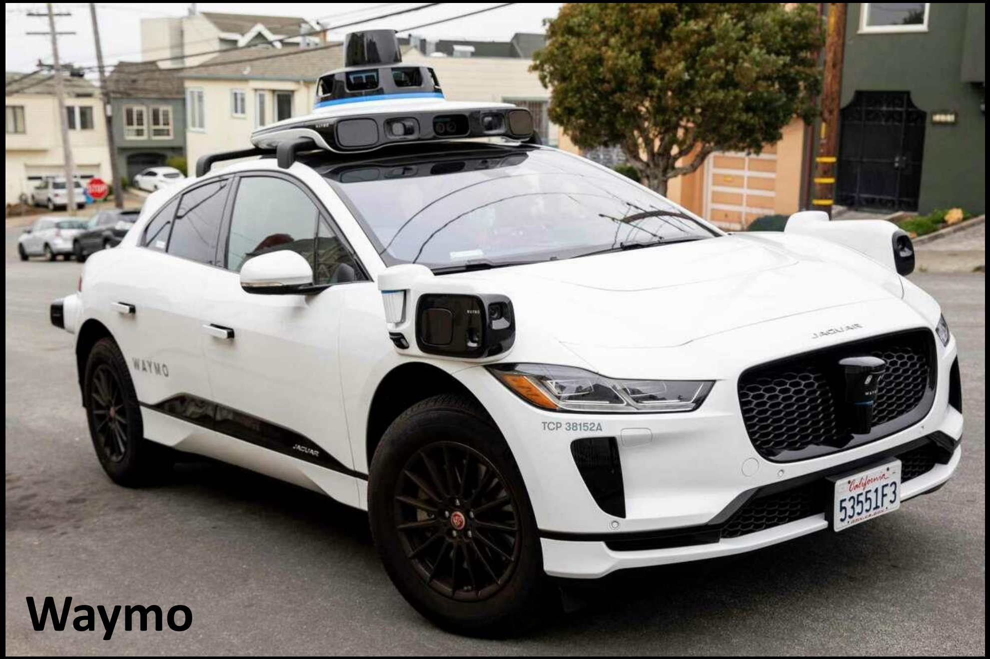
Stepping Towards Unsupervised Scene Understanding

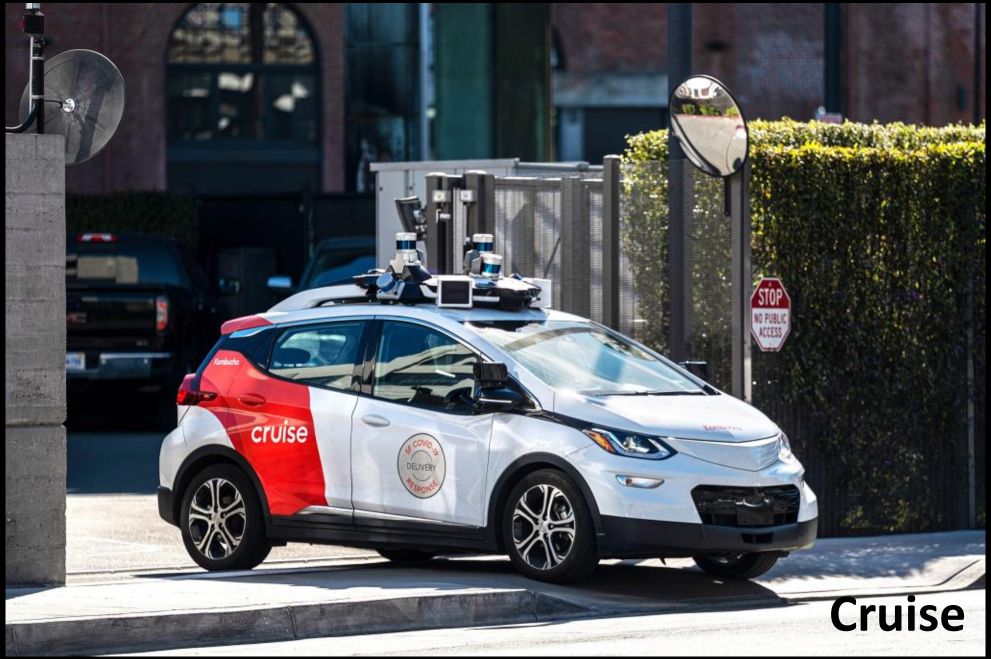
Raoul de Charette

rdecharette.github.io | raoul.de-charette@inria.fr

Scene understanding

AD - Level 4


Waymo
WAYMO
53551F3

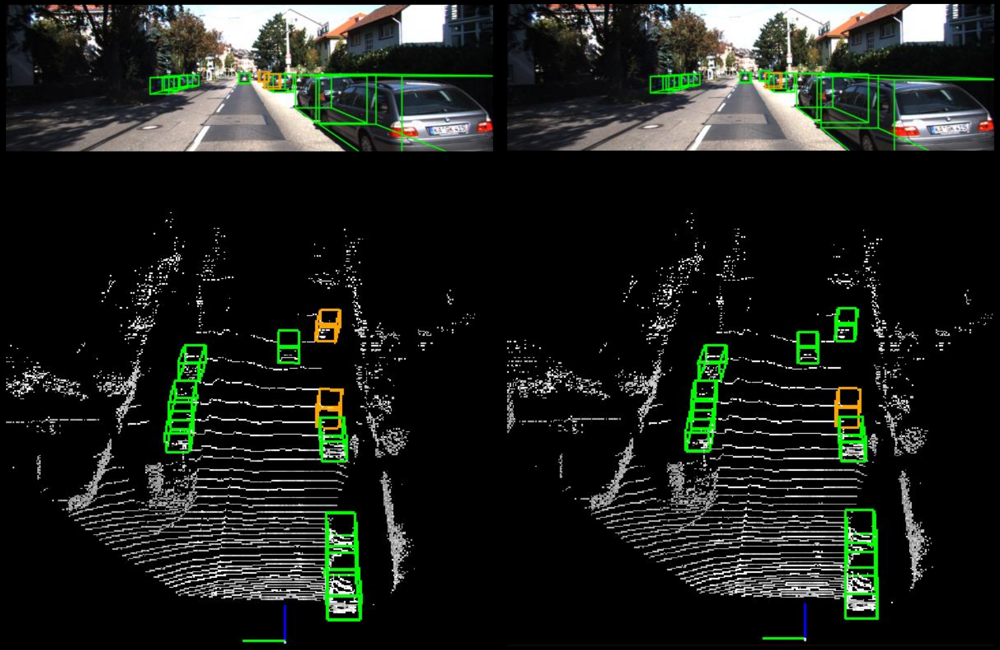

cruise
STOP
NO PUBLIC ACCESS
Cruise


apollo
Baidu

(a quick partial/biased overview of CV pillars for AD)

Semantics | Geometry | Motion

Object detection



(Chen et al., ECCV 2020)

Semantic Segmentation

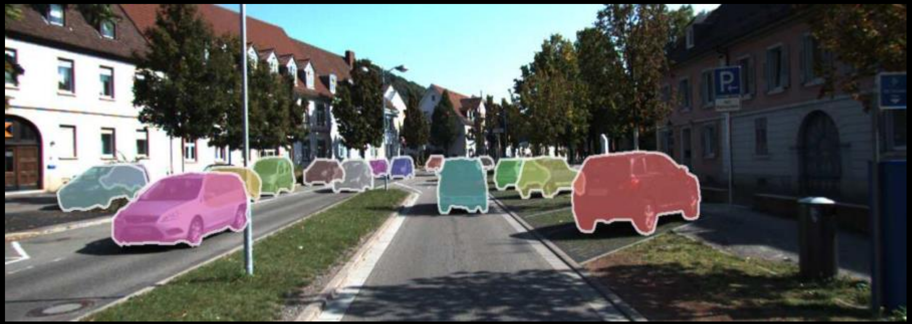
(Silberman et al., ECCV 2012)

1 color = 1 semantic class

Semantics | Geometry | Motion

Amodal Segmentation (2016)

(Li and Malik, ECCV 16)



KINS (Qi et al., CVPR 19)

Segmentation of visible and invisible pixels

"Things and Stuff"

Panoptic segmentation (2019)

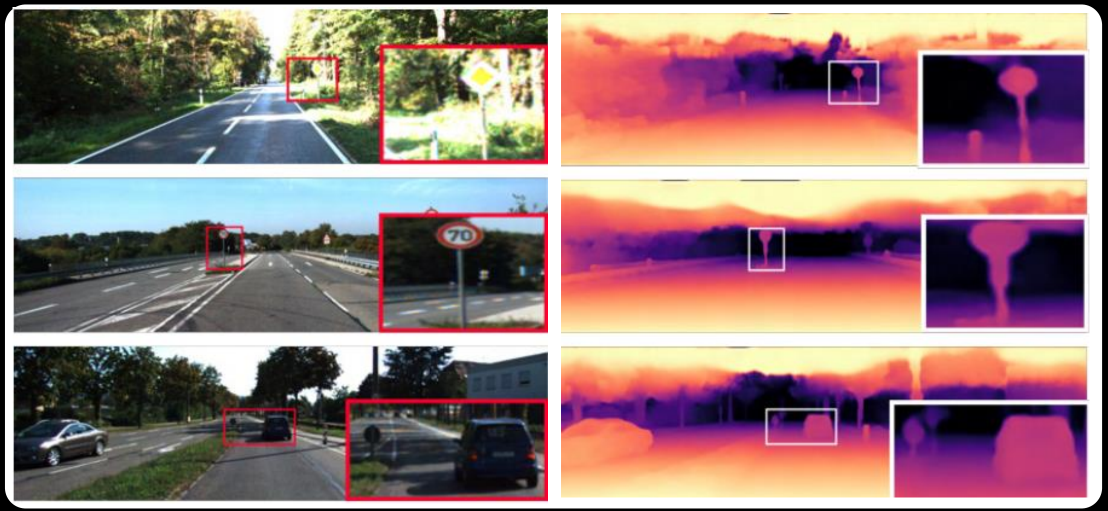
(Kirillov et al., CVPR 2019)

1 color = 1 instance

Segmentation of class and instance at once

Semantics | **Geometry** | Motion

Depth prediction

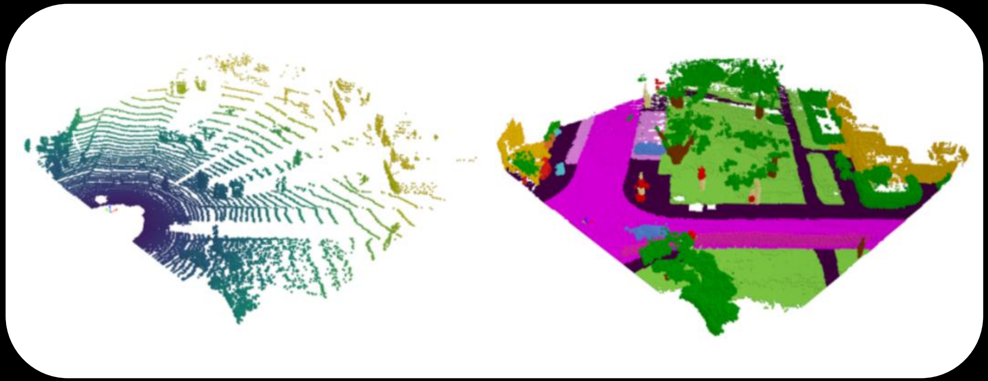


(Bhat et al., CVPR 21)

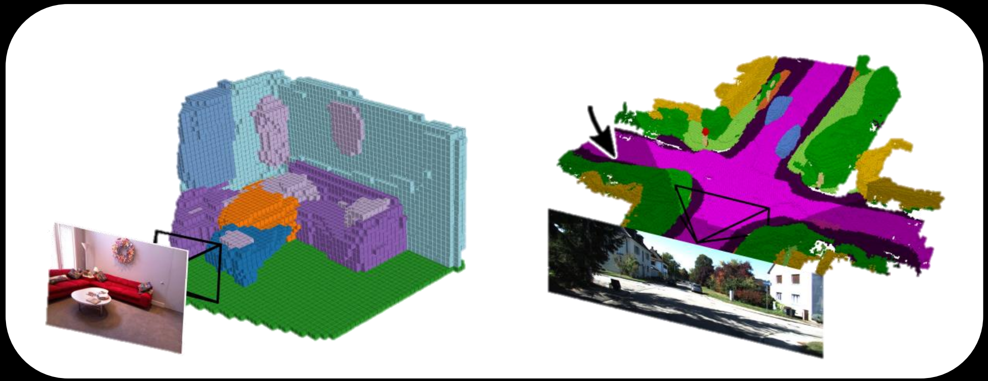
Reconstruction

(Cao and de Charette, ICCV 23)

Semantic Scene Completion



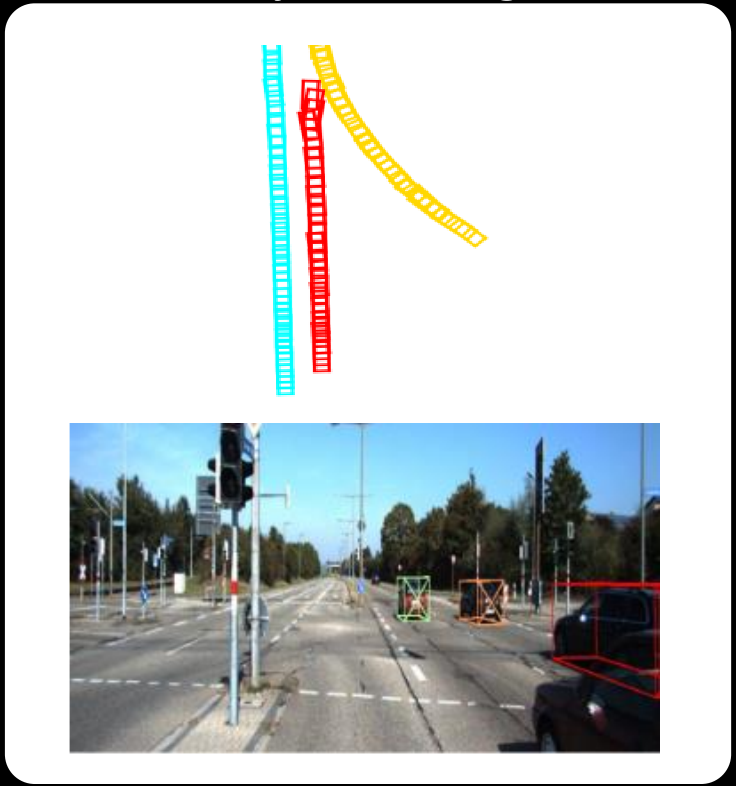
(Roldao et al., IJCV 21)



(Cao and de Charette, CVPR 22)

Semantics | Geometry | Motion

Object tracking



CAMO-MOT. (Wang et al., 22)

Pixel tracking

(Wang et al., ICCV 23)

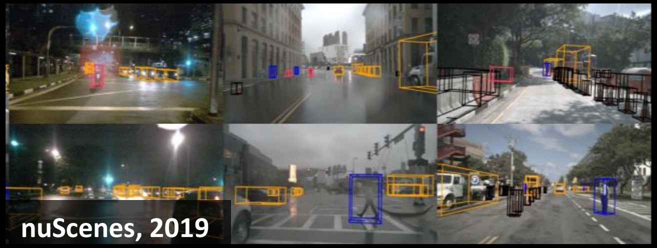
Semantics | Geometry | **Motion**

Forecasting

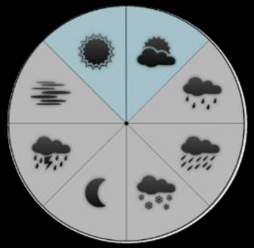
(Liu et al., CVPR 21)

And many more..

Data proficiency (10^7)



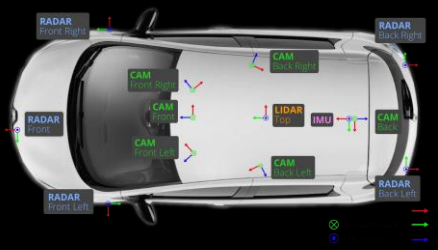
Biases



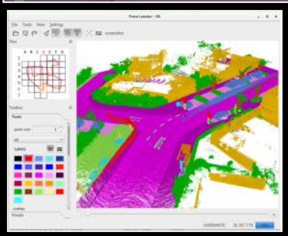
(Torralba and Efros, CVPR 11)

Cost

For 1 annotator 8h/365d

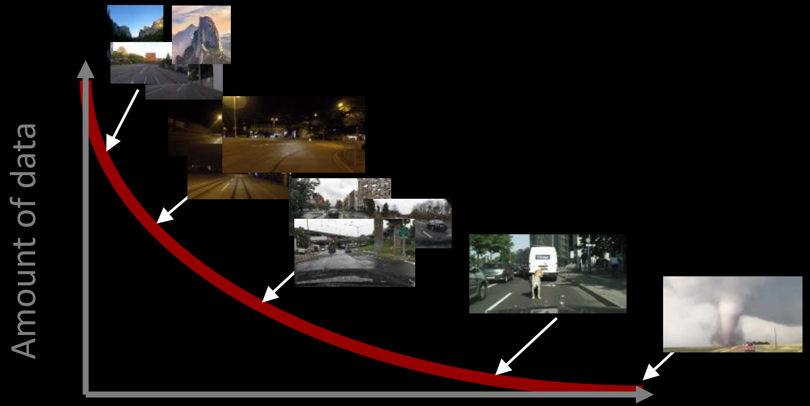


~8y



~1y

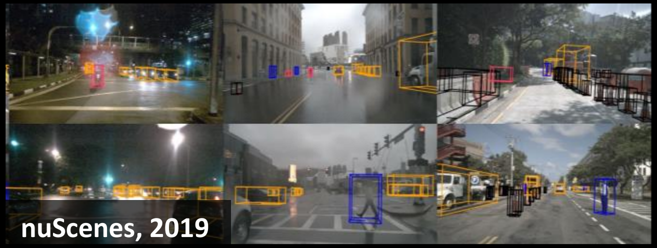
World is long-tail



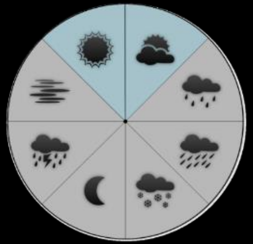
Data proficiency (10^7)


KITTI, 2012 / Sem.KITTI, 2019


Waymo Open Dataset, 2020


nuScenes, 2019

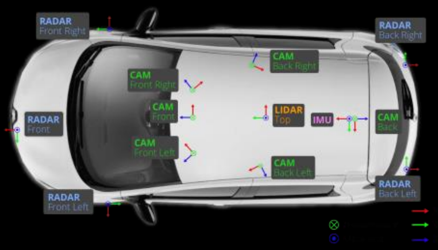
Biases



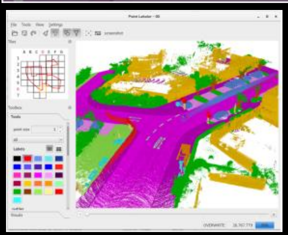
(Torralba and Efros, CVPR 11)

Cost

For 1 annotator 8h/365d

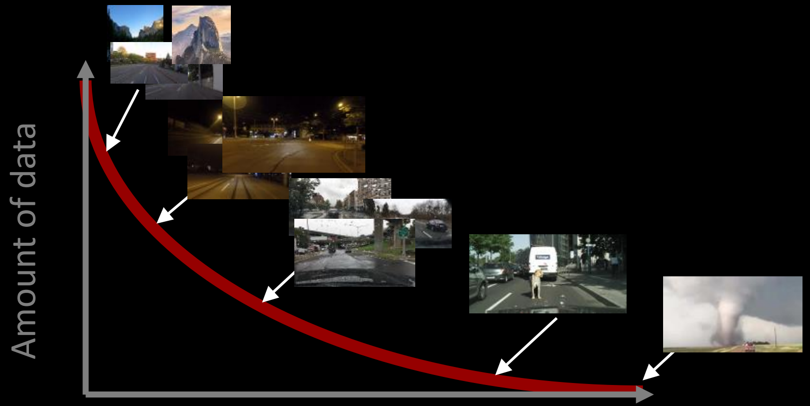


~8y

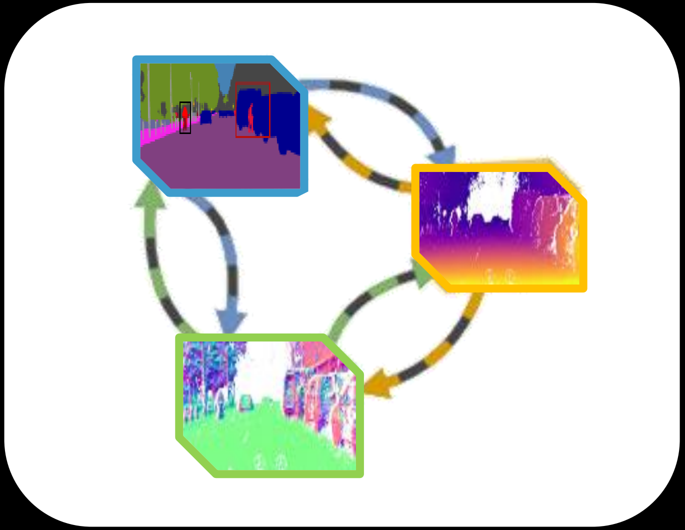


~1y

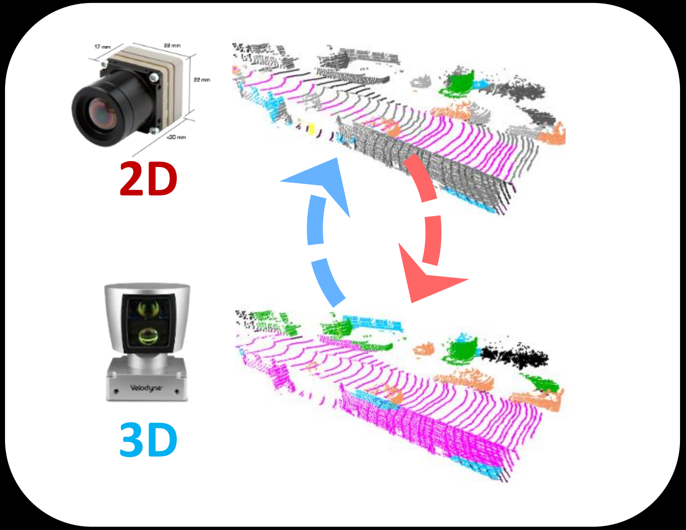
World is long-tail



Supervised learning is doomed to Out Of Distribution



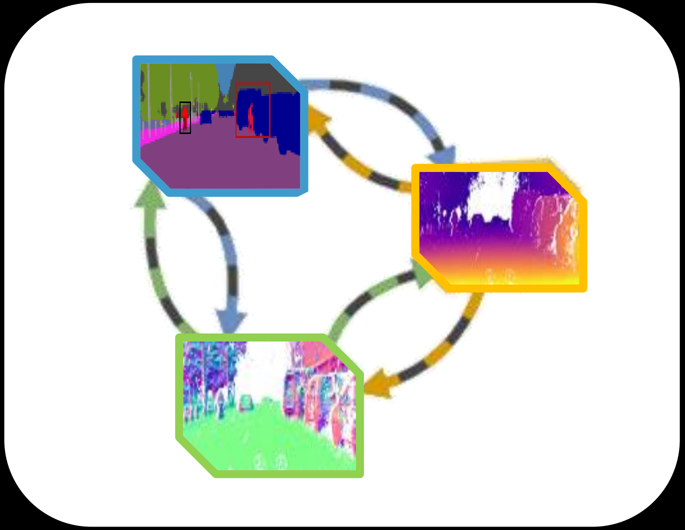
Multi-Task Learning



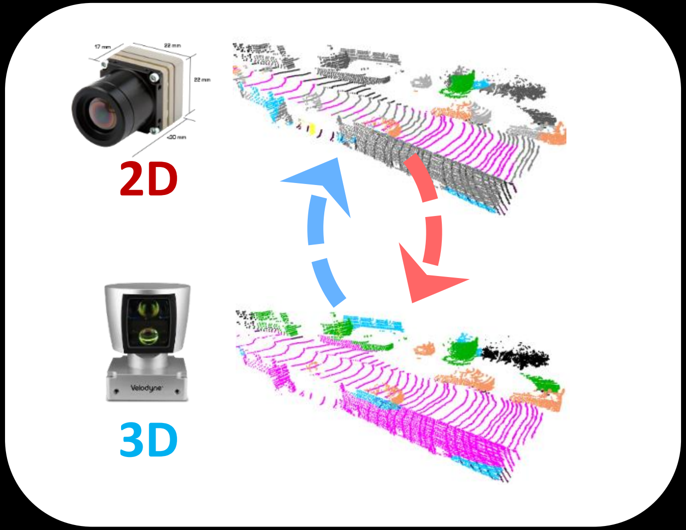
Cross-Modal Learning

Prompt-driven Learning

Knowledge Distillation



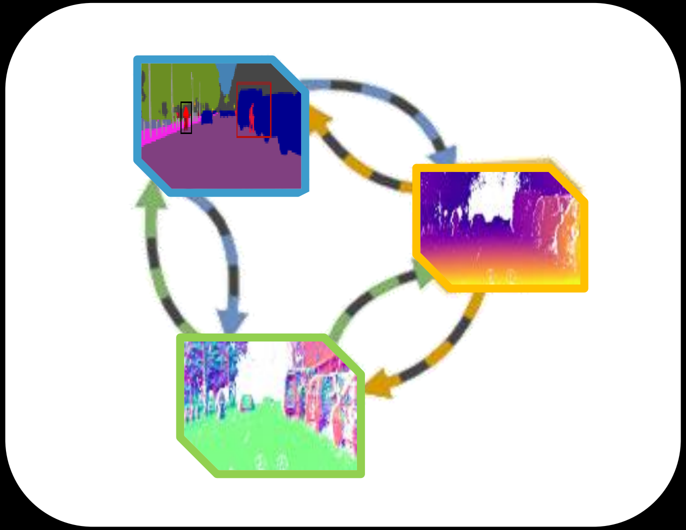
Multi-Task Learning



Cross-Modal Learning

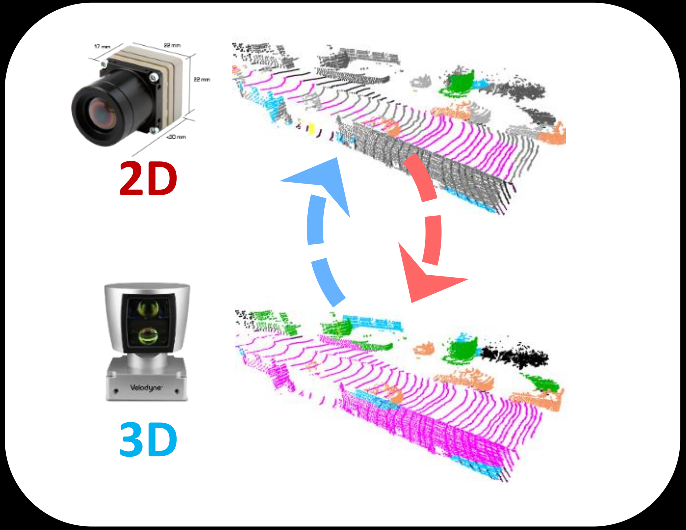
Prompt-driven Learning

Knowledge Distillation



Multi-Task Learning

(Lopes et al., WACV 2023)

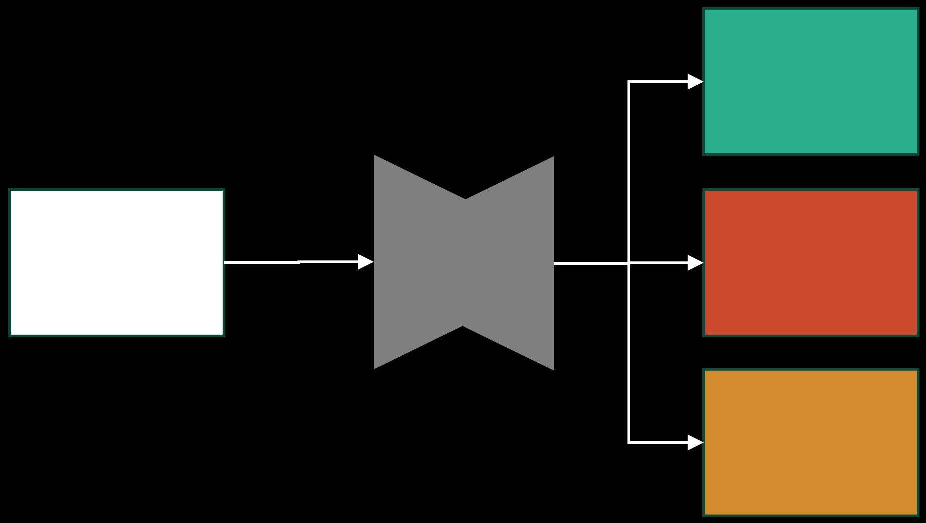


Cross-Modal Learning

(Jaritz et al., TPAMI 2022)

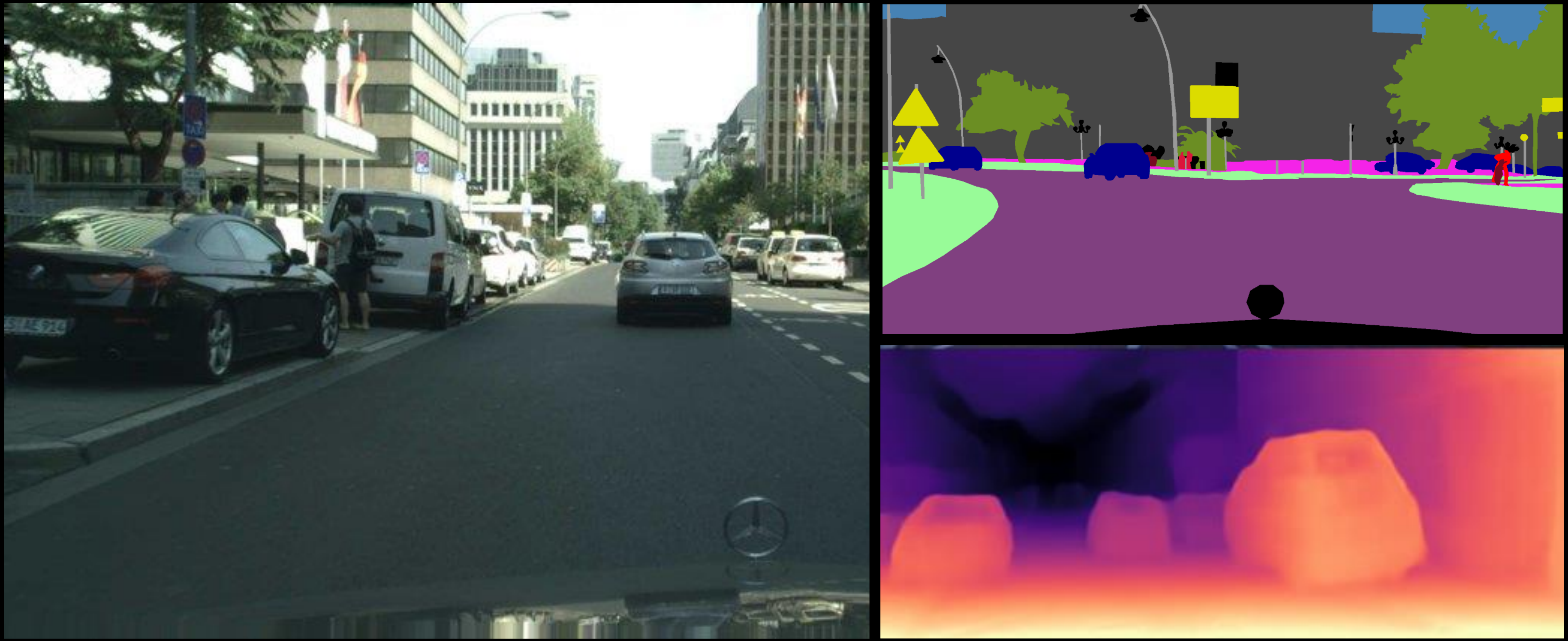
Prompt-driven Learning

(Fahes et al., ICCV 2023)

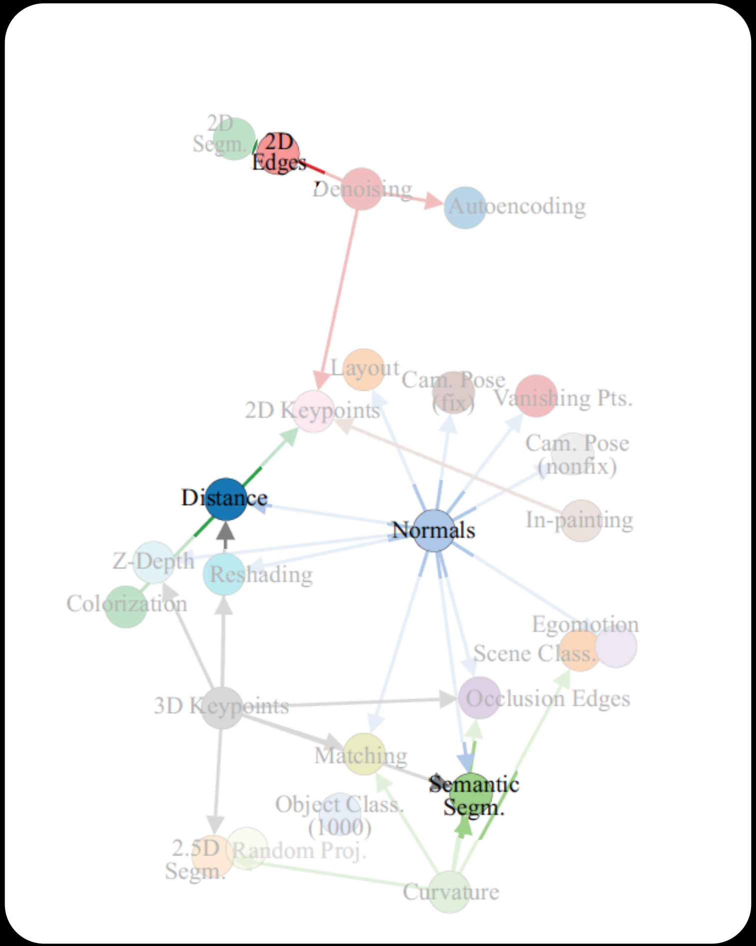


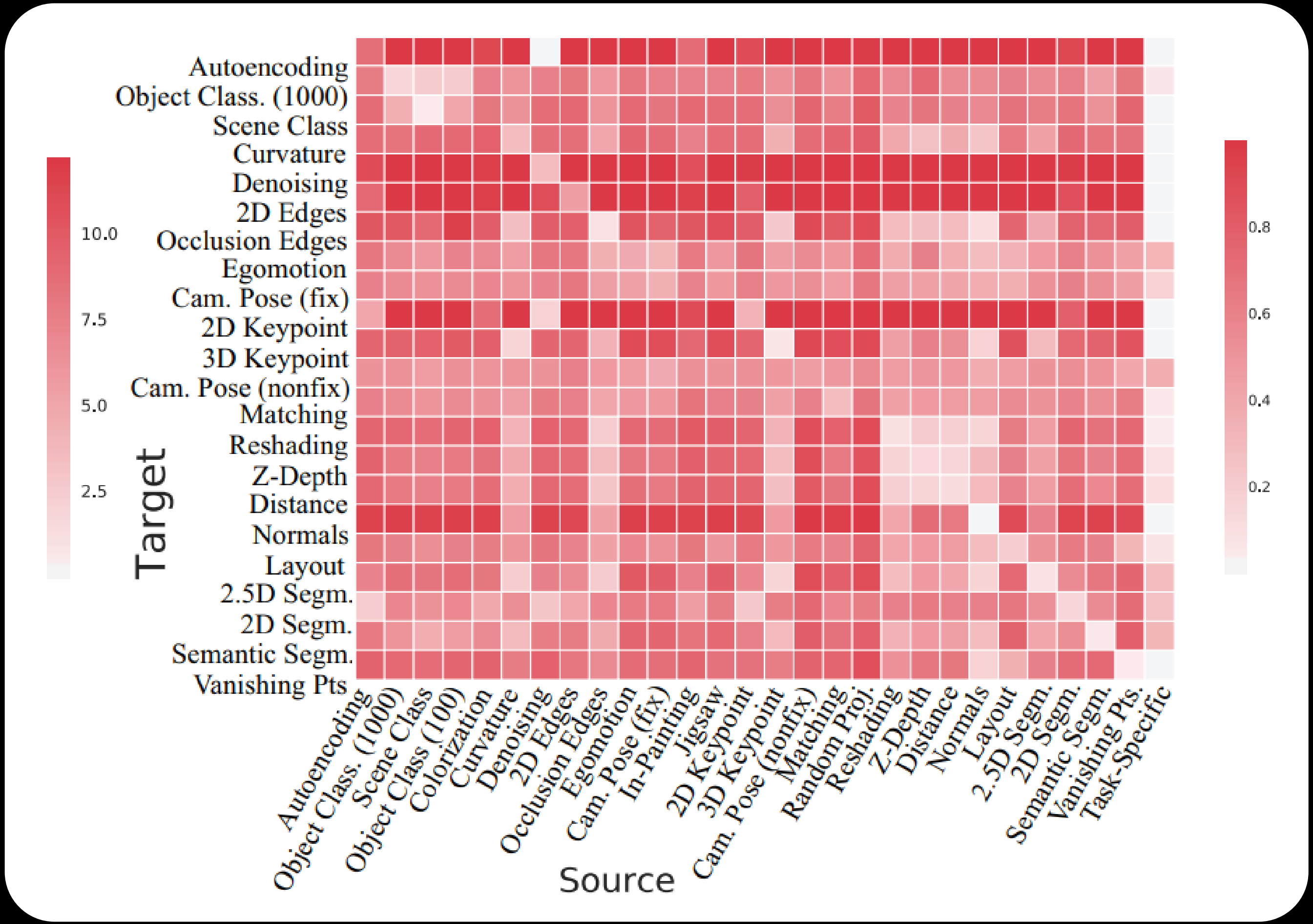
Multi-task Learning

$$\{T_1, \cdots, T_n\}$$



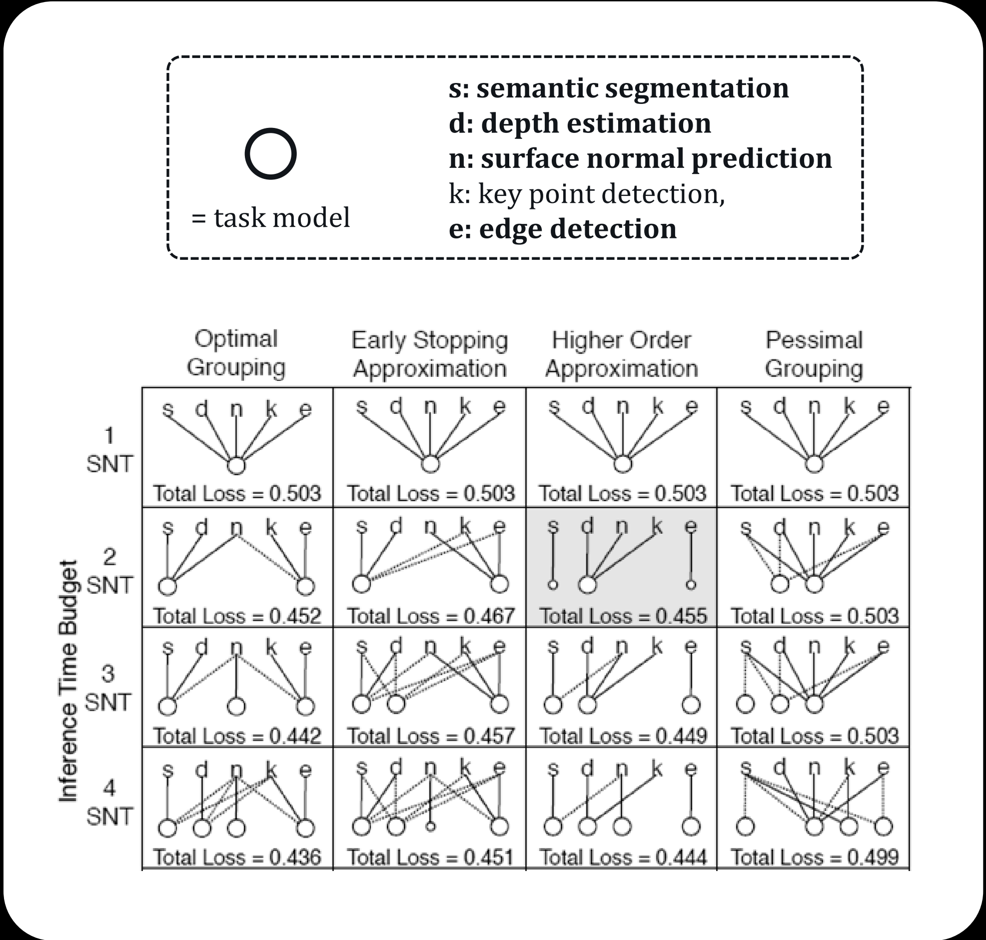
Semantics conveys geometry cues, and vice versa.





TRANSFER LEARNING

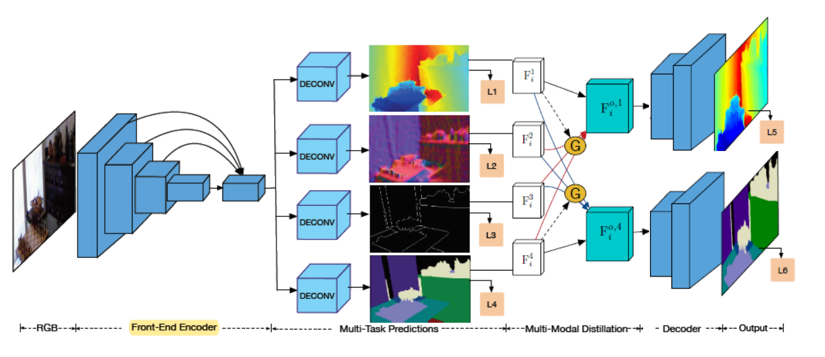
Taskonomy [Zamir *et al.*, CVPR'18]



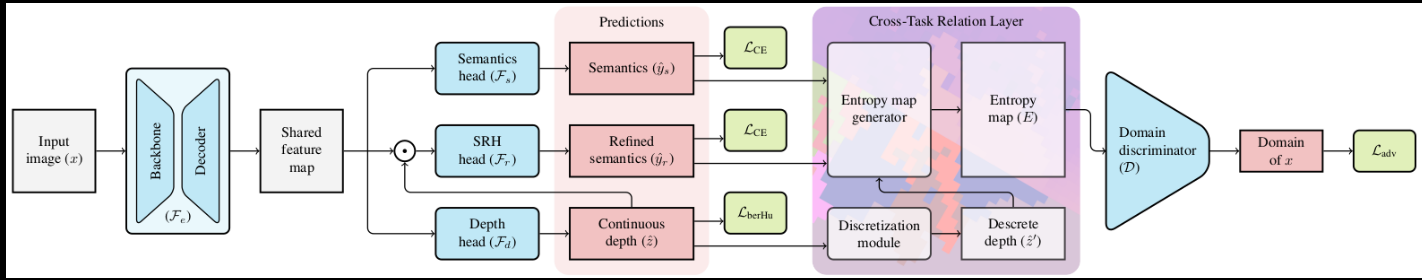
MULTI-TASK LEARNING

Which tasks to learn together in MTL [Standley *et al.*, ICML'20]

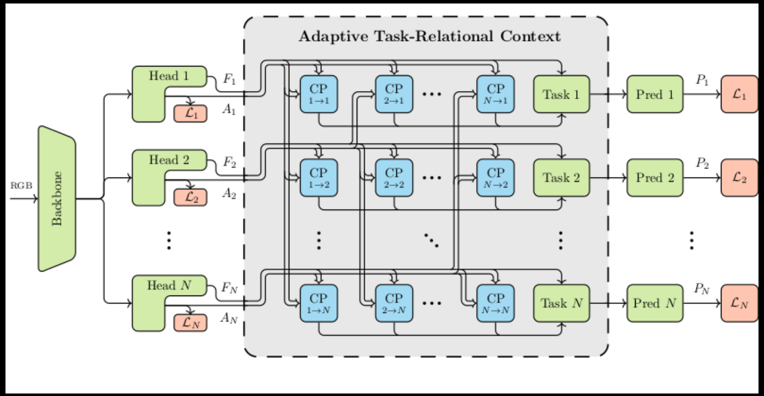
PAD-Net [Xu *et al.*, CVPR’18]



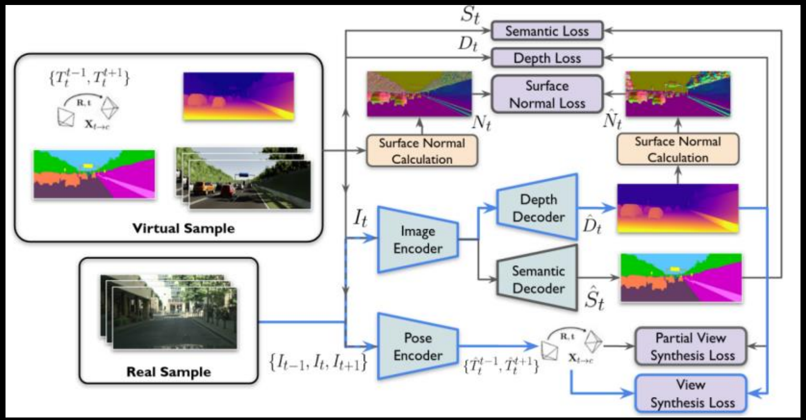
CTRL-UDA [Saha *et al.*, CVPR’21]



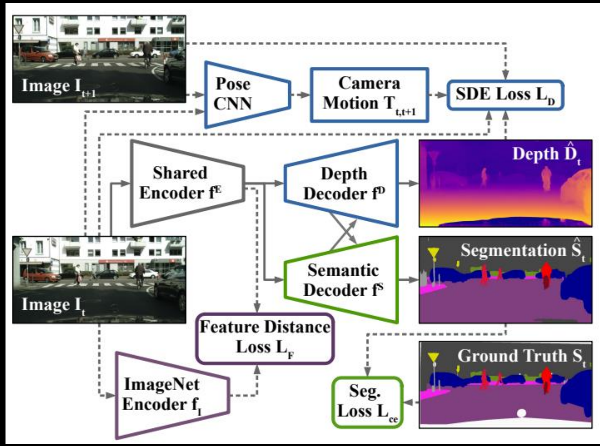
ATRC [Bruggemann *et al.*, ICCV’21]



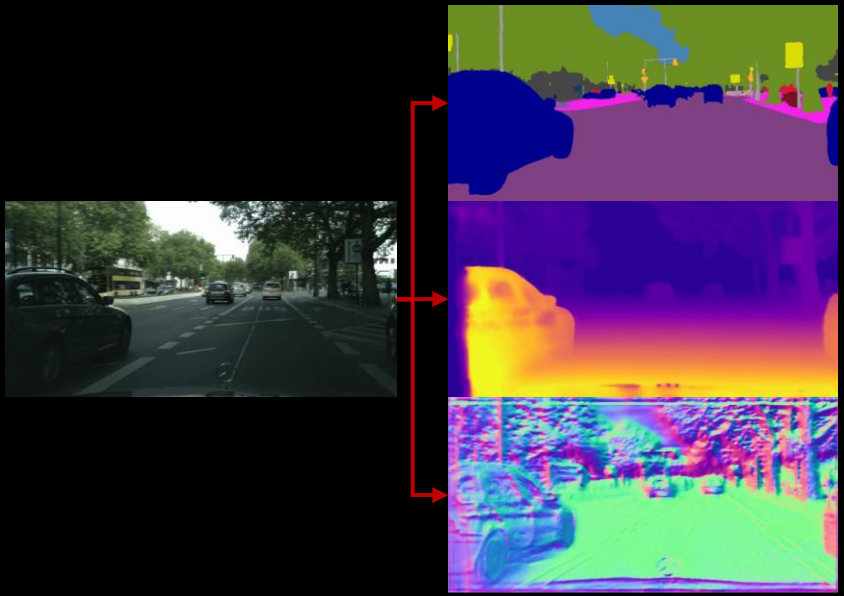
GUDA [Guizilini *et al.*, ICCV’21]



3-Ways [Hoyer *et al.*, CVPR’21]



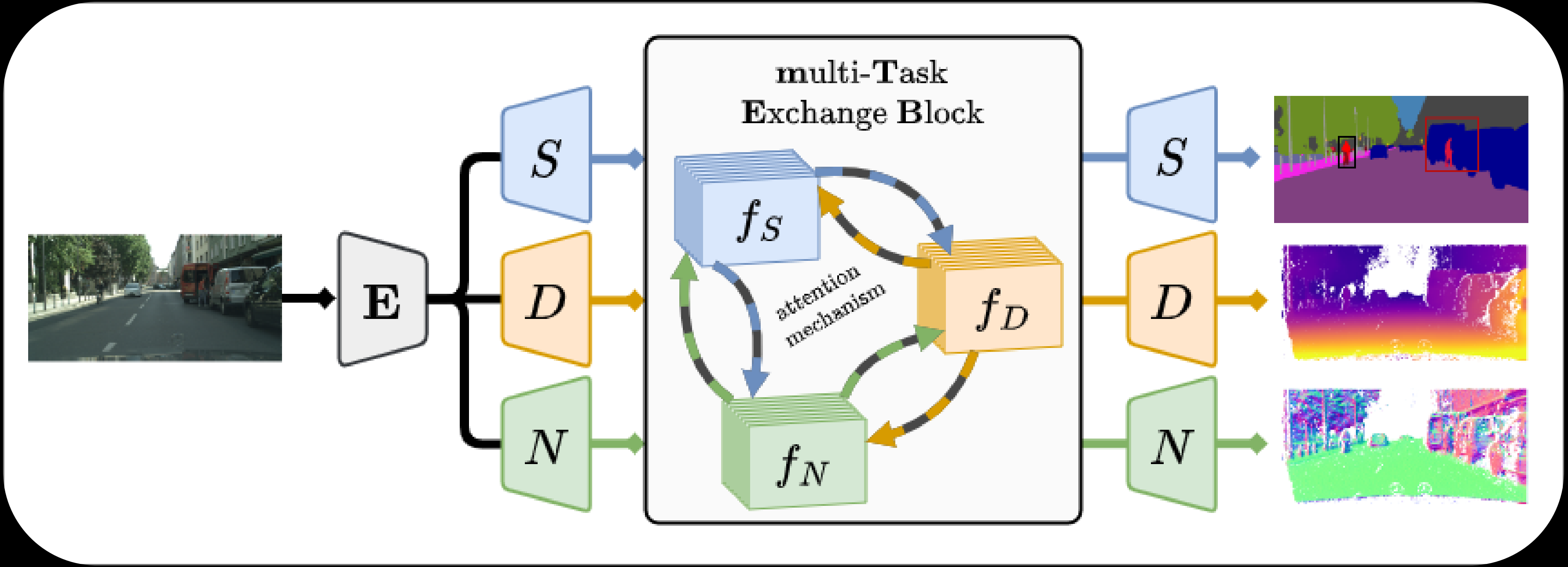
Survey: [Vandenhende et al., TPAMI 2020]



DenseMTL:
Multitask Learning for UDA

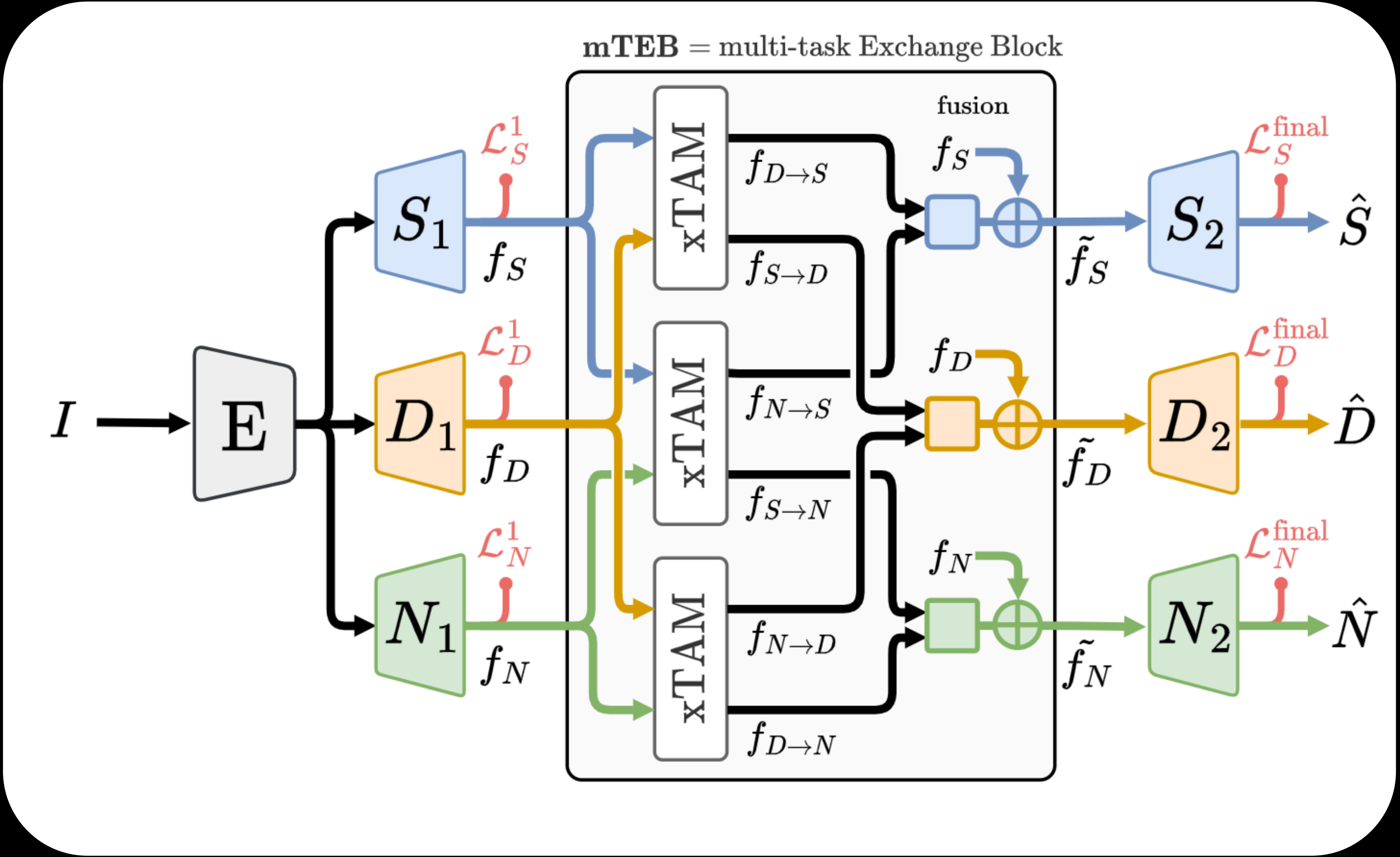
github.com/astra-vision/DenseMTL

I. Lopes, T-H. Vu, R. de Charette, WACV 2023

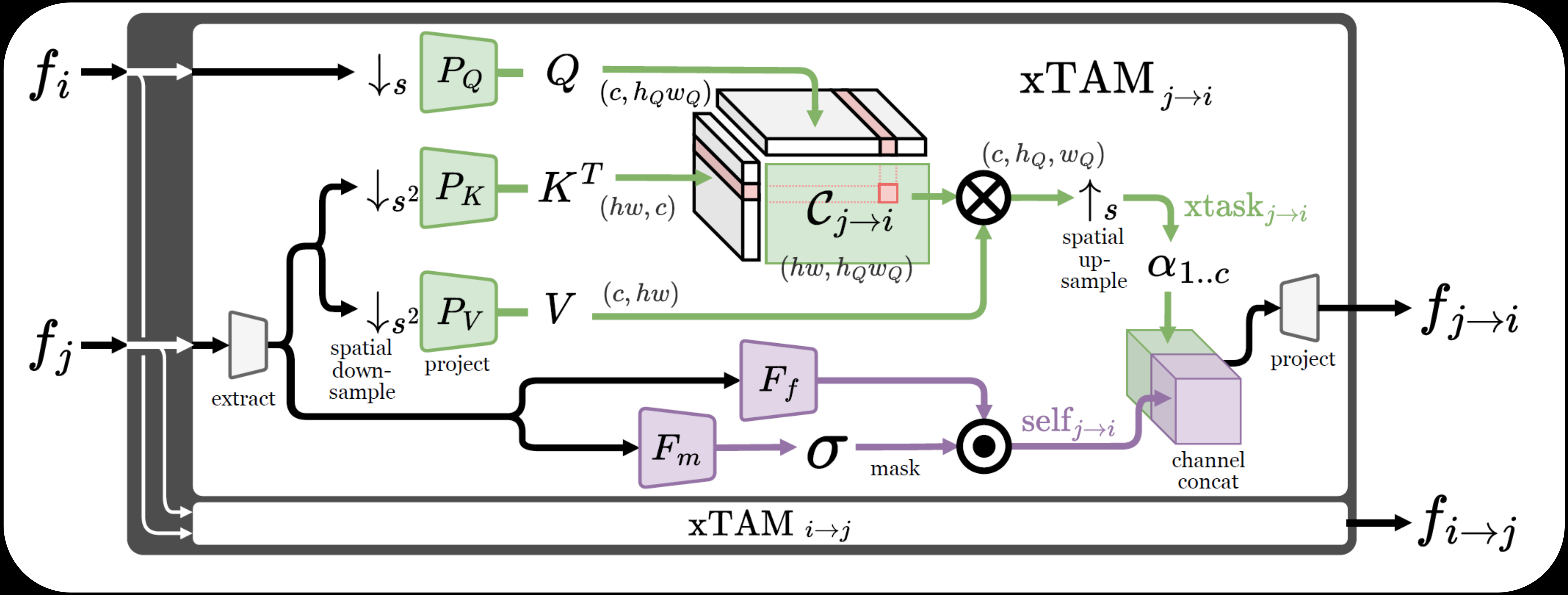


3 set of tasks: `S-D’, `S-D-N’, `S-D-N-E’

multi-Task Exchange Block (mTEB)

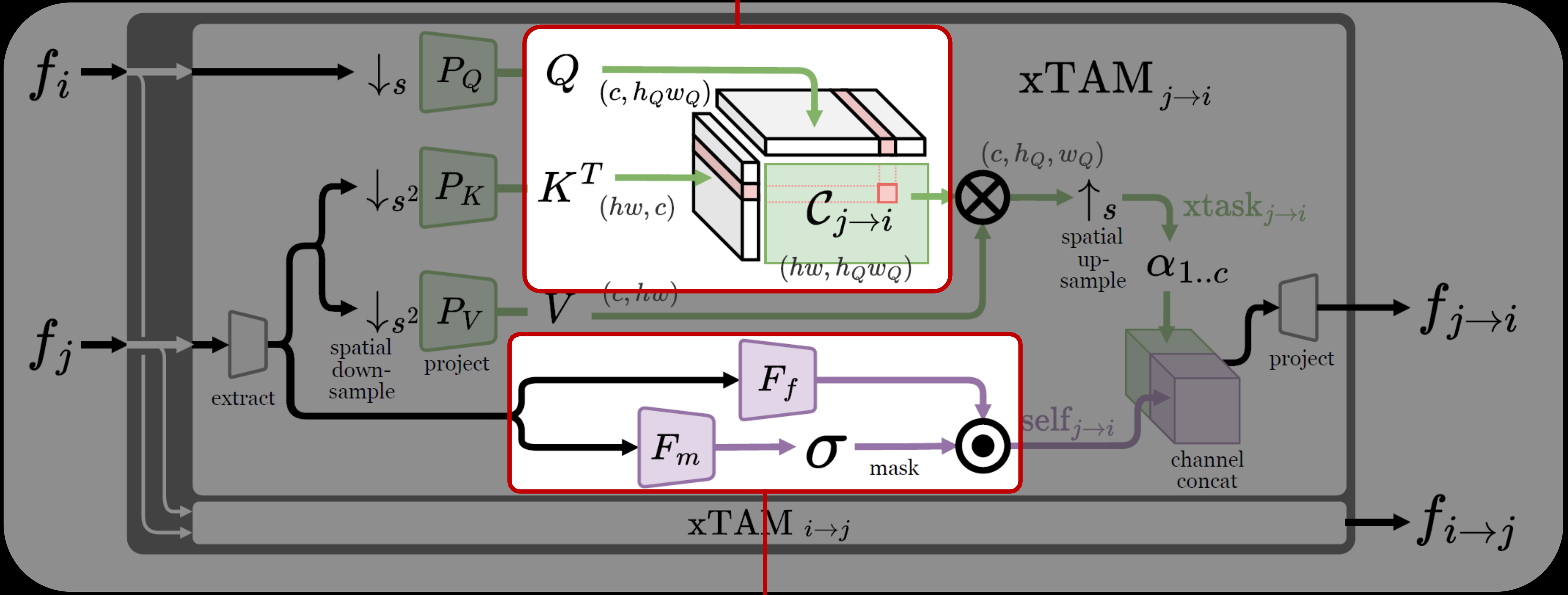


Guide feature inclusion from task ***j*** to task ***i*** and vice-versa



What tasks ***j*** and ***i*** have in common, to help task ***i*** ?

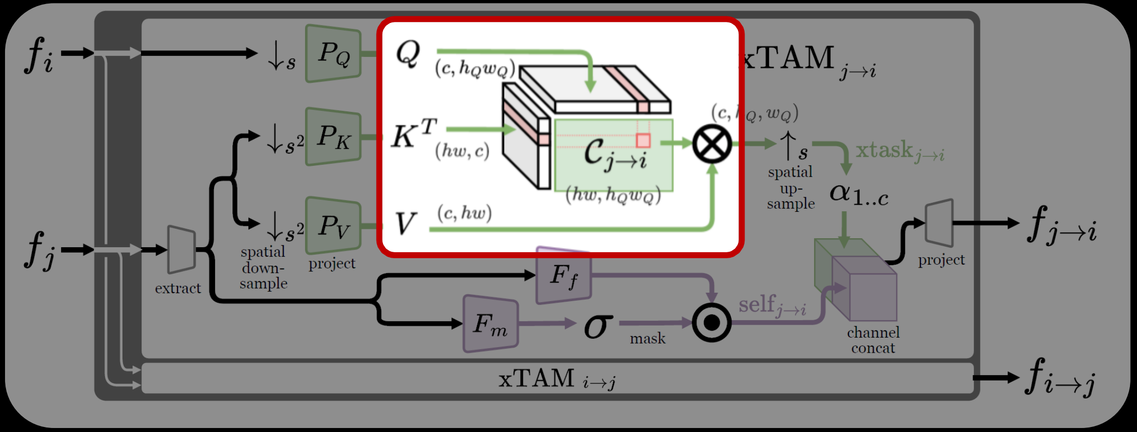
Cross-task attention



Self-attention

What task ***j*** can do alone, to help task ***i*** ?

Correlation-guided attention $\mathcal{C}_{j \to i}$



The intuition: injecting features from *j* that will contribute to better solving task *i*.

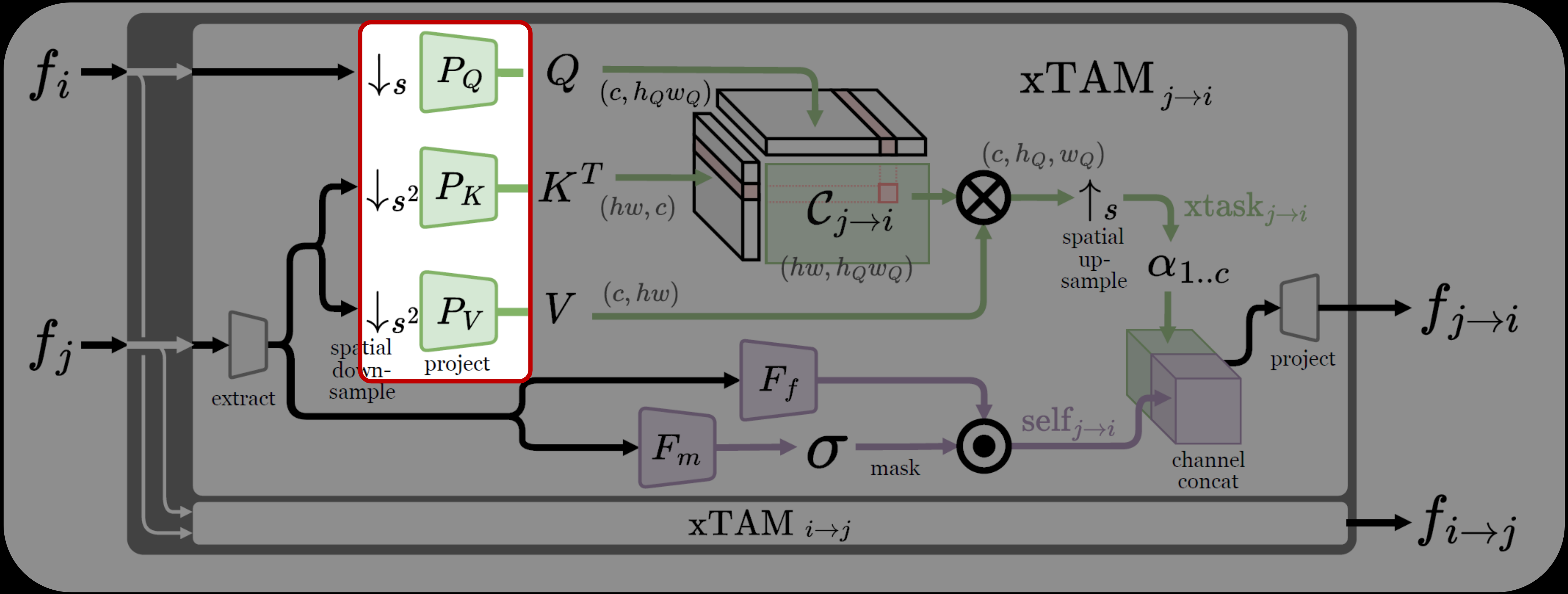
$$Q = \text{transform}(f_i)$$ Features from task *i*

$$\left.\begin{aligned} K^{\mathrm{T}} &= \text{transform}(f_j) \\ V &= \text{transform}(f_j) \end{aligned}\right\}$$ Features from task *j*

$$\mathcal{C}_{j \to i} = \text{softmax}\left(\frac{K^T \times Q}{\sqrt{d}}\right)$$

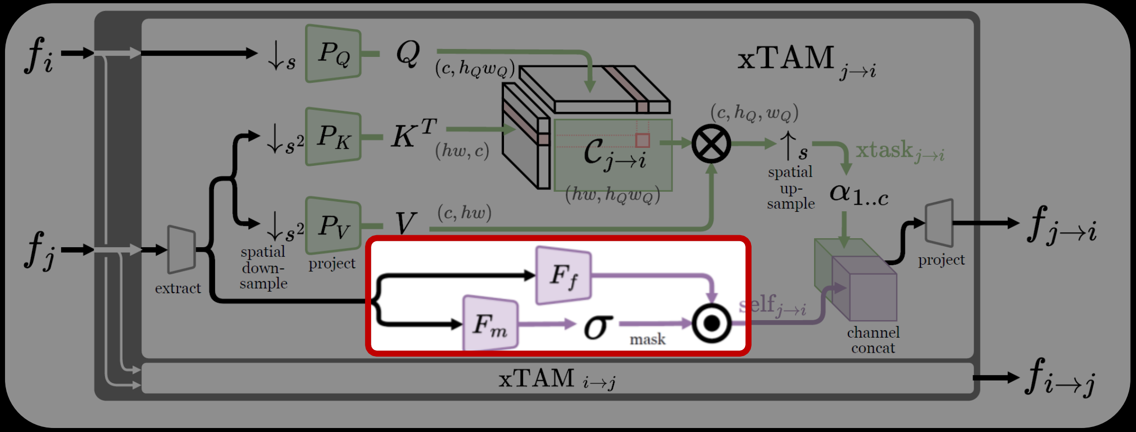
We account more for features of *j* which are highly spatially-correlated with features from *i*

$$\mathbf{xtask}_{j \to i} = V \times \mathcal{C}_{j \to i}$$



f_i
f_j
$\downarrow_s$
P_Q
$\downarrow_{s^2}$
P_K
$\downarrow_{s^2}$
P_V
spatial down-sample
project
extract
Q
$(c, h_Q w_Q)$
K^T
(hw, c)
V
(c, hw)
$\mathcal{C}_{j\to i}$
$(hw, h_Q w_Q)$
xTAM $_{j\to i}$
(c, h_Q, w_Q)
$\uparrow_s$
spatial up-sample
xtask$_{j\to i}$
$\alpha_{1..c}$
F_f
F_m
σ
mask
self$_{j\to i}$
channel concat
project
$f_{j\to i}$
xTAM $_{i\to j}$
$f_{i\to j}$

Self-attention



The intuition: discover private features from *j* that help task *i*

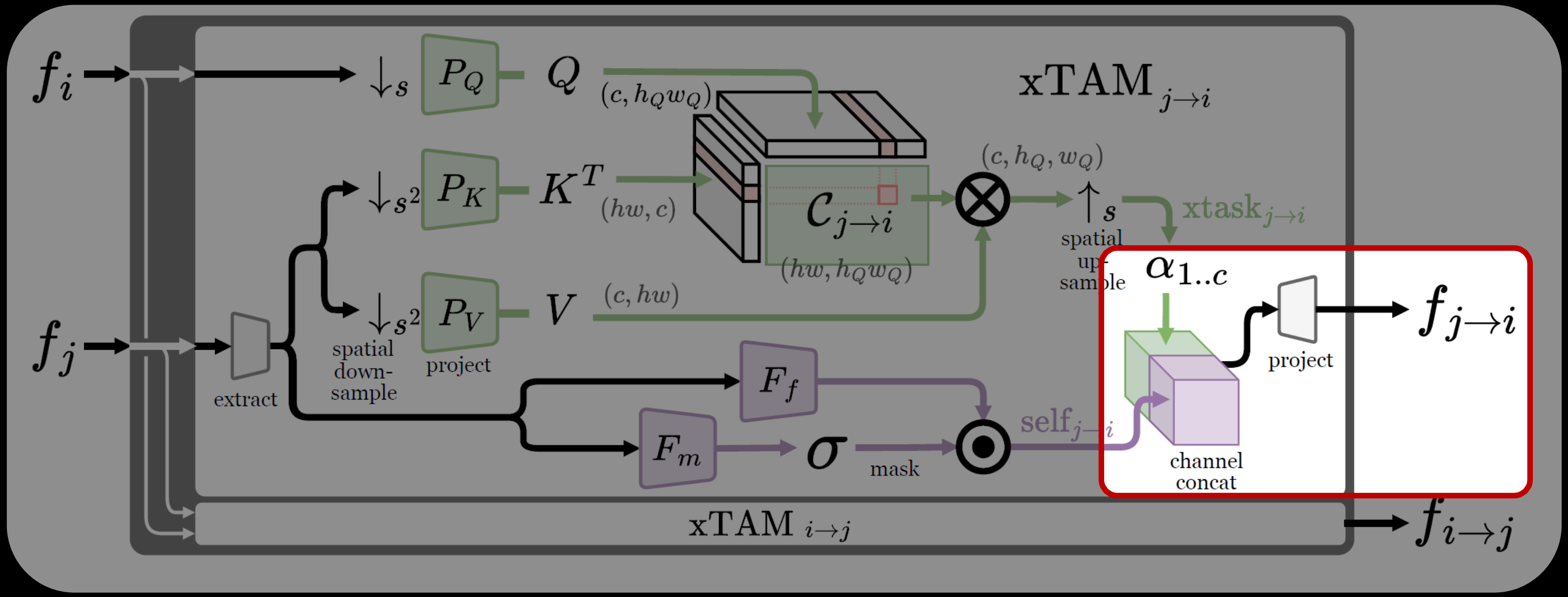
$F_f(f_j)$ Convolution block

$F_m(f_j)$ Convolution block

We let gradient flow optimize *F_f, F_m* to discover relevant features in *j* for task *i*

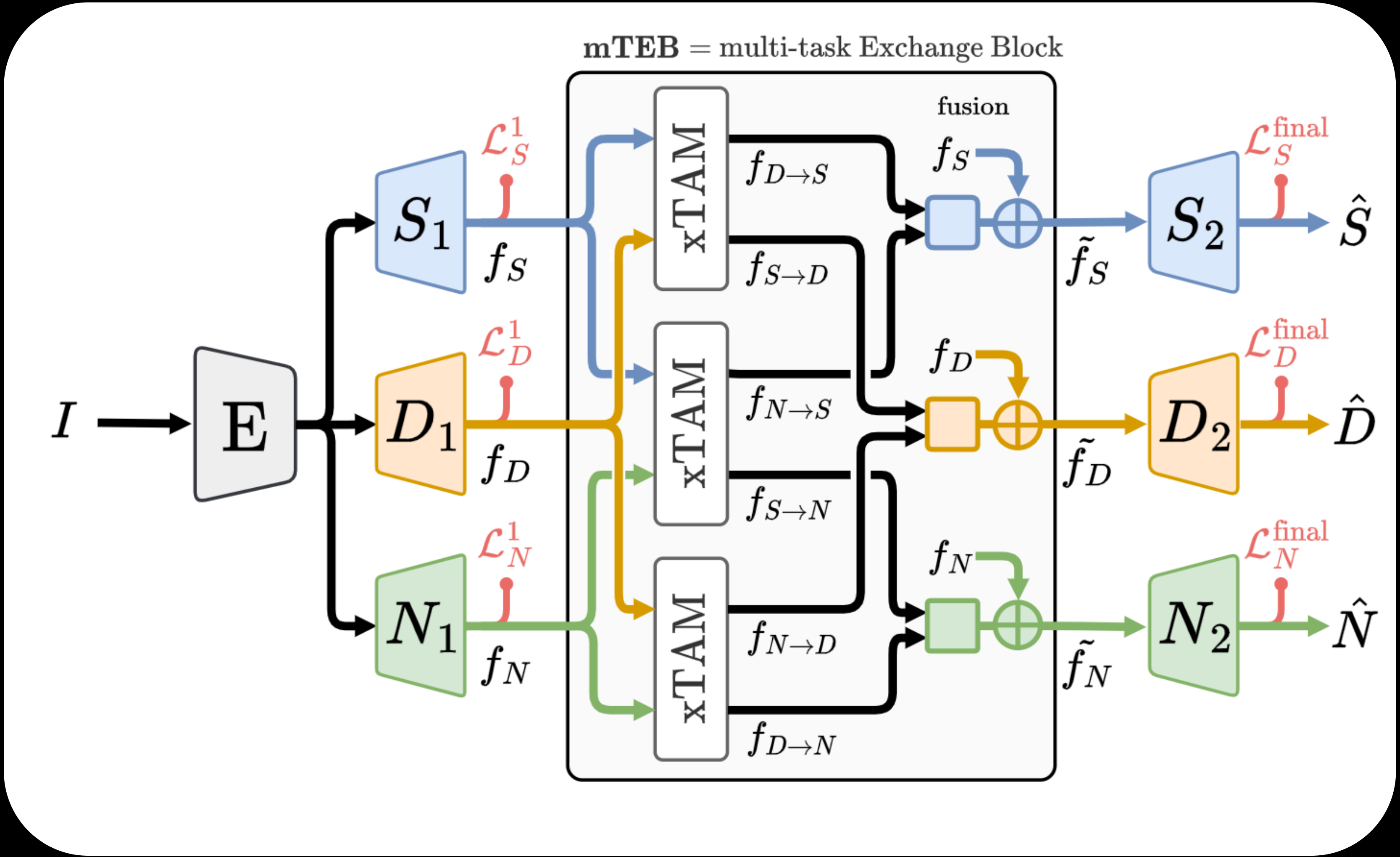
$$\mathbf{self}_{j\to i} = F_f(f_j) \odot \sigma(F_m(f_j))$$

Fusion



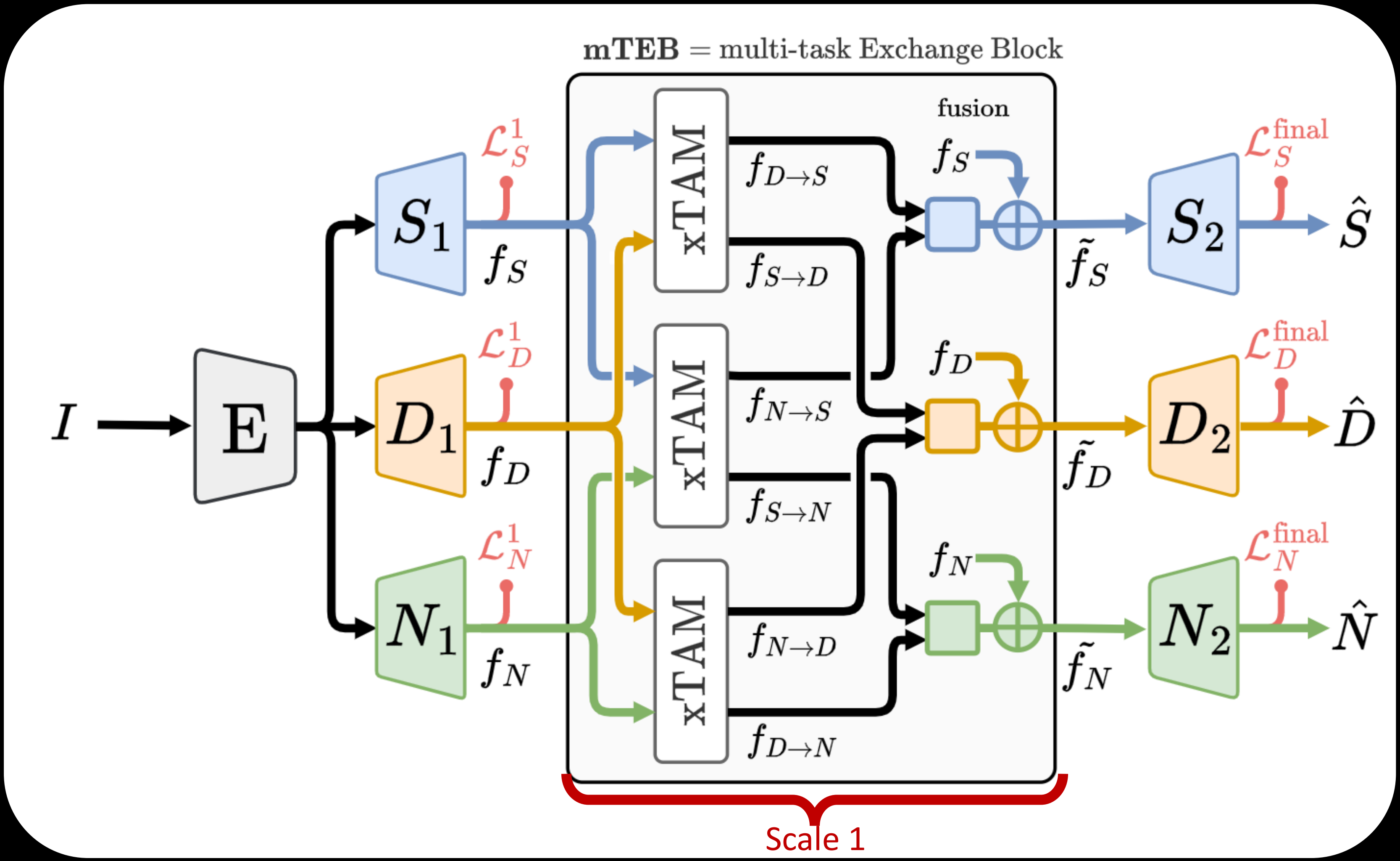
f_i
f_j
extract
spatial down-sample
project
$\downarrow_s$
P_Q
Q
$(c, h_Q w_Q)$
xTAM $_{j \to i}$
$\downarrow_{s^2}$
P_K
K^T
(hw, c)
$\mathcal{C}_{j \to i}$
$(hw, h_Q w_Q)$
(c, h_Q, w_Q)
$\uparrow_s$
spatial up-sample
xtask$_{j \to i}$
$\downarrow_{s^2}$
P_V
V
(c, hw)
F_f
F_m
σ
mask
self$_{j \to i}$
$\alpha_{1..c}$
channel concat
project
$f_{j \to i}$
xTAM $_{i \to j}$
$f_{i \to j}$

multi-task Exchange Block (mTEB)



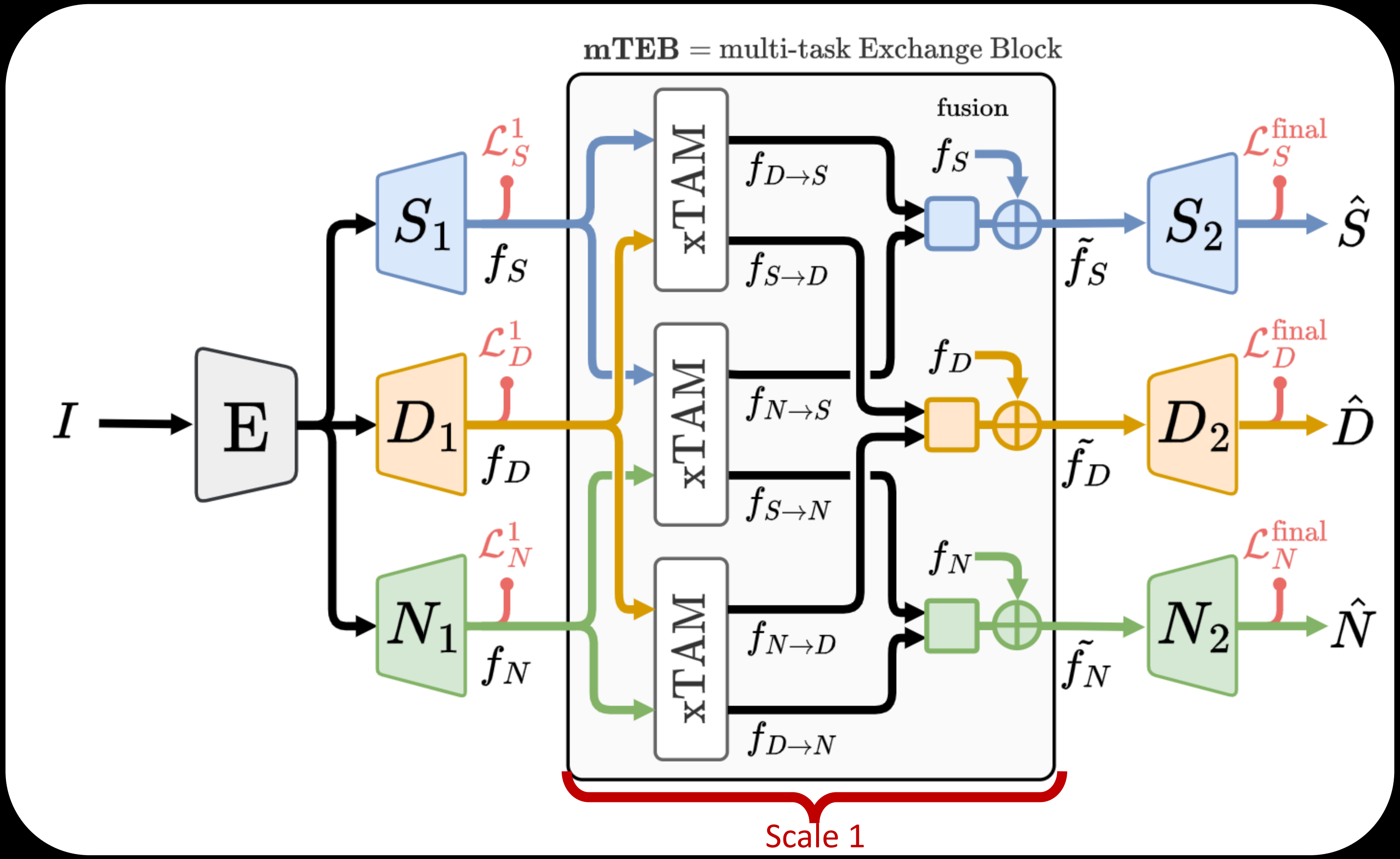
mTEB can be inserted at any scale

multi-task Exchange Block (mTEB)



mTEB can be inserted at any scale

multi-task Exchange Block (mTEB)



mTEB can be inserted at any scale

$$\mathcal{L}_{\text{tasks}} = \frac{1}{|S|} \sum_{s\in S} \sum_{t\in T} \omega_t \mathcal{L}_t^s + \sum_{t\in T} \omega_t \mathcal{L}_t^{\text{final}}$$

The challenge of MTL metrics

- Metrics and scale differ per task: depth (RMSE), semantics (mIoU), etc.
- MTL should favor **all** metrics

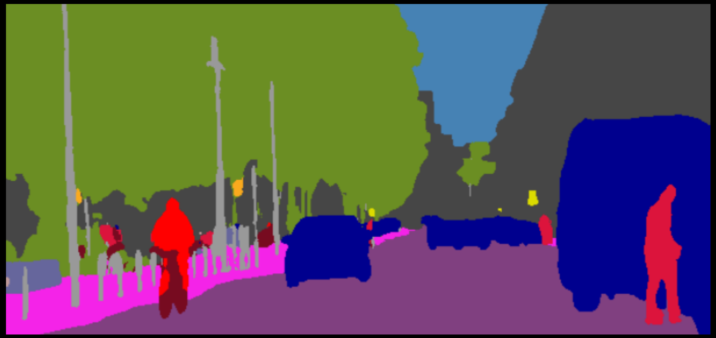
0 for higher is better, 1 lower is better

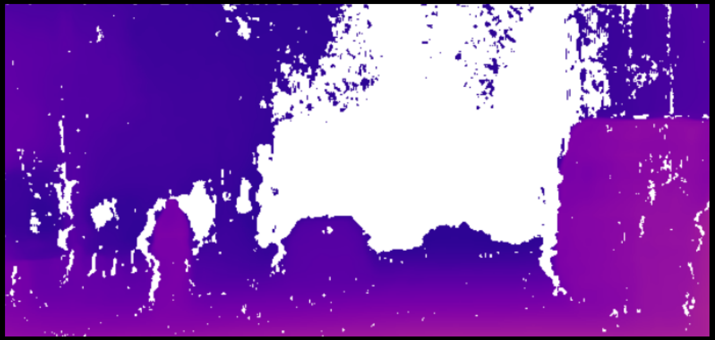
$$\Delta_{\mathrm{T}}(\mathbf{f}) = 1/n \sum_{i \in T} (-1)^{g_i} (m_i - b_i)/b_i$$

MTL performance

STL performance

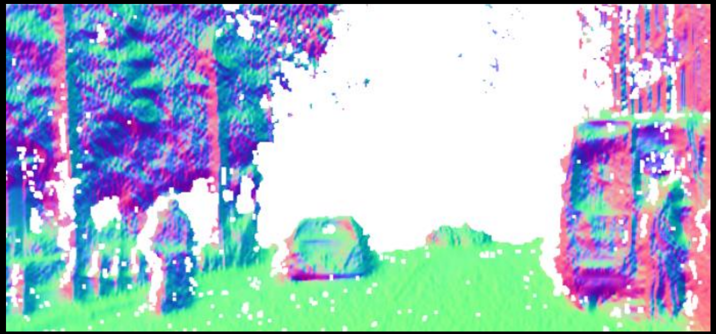
Segmentation

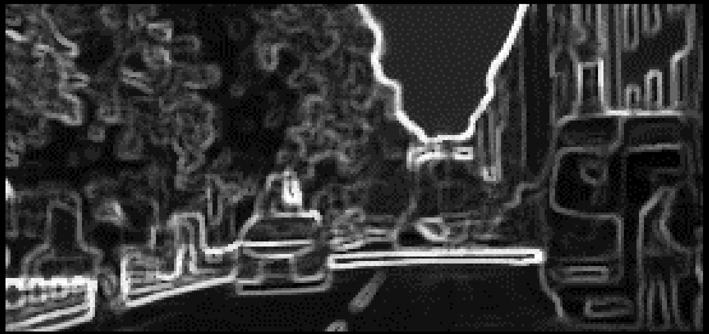




Depth

Normal





Edge

Fully supervised

NYUDv2

Methods	'S-D'			'S-D-N'				
	Semseg ↑ mIoU %	Depth ↓ RMSE m	Delta ↑ Δ_{SD} %	Semseg ↑ mIoU %	Depth ↓ RMSE m	Delta ↑ Δ_{SD} %	Normals ↓ mErr. °	Delta ↑ Δ_{SDN} %
STL [8]	38.70 ±0.10	0.635 ±0.013	↴	*idem*	*idem*	↴	36.90 ±0.26	↴
MTL [8]	39.44 ±0.34	0.638 ±0.004	+1.63 ±0.37	39.90 ±0.41	0.642 ±0.003	+1.89 ±0.67	36.07 ±0.09	+1.76 ±0.53
PAD-Net [135]	35.30 ±0.84	0.659 ±0.004	-5.36 ±0.83	36.14 ±0.30	0.660 ±0.006	-4.32 ±0.68	36.72 ±0.08	-2.97 ±0.43
3-ways$_{\text{PAD-Net}}$ [23]	**39.47** ±0.16	0.622 ±0.001	+2.90 ±0.23	**40.28** ±0.30	0.619 ±0.004	+4.16 ±0.50	35.35 ±0.09	+3.93 ±0.27
Ours	38.93 ±0.35	**0.604** ±0.004	**+3.54** ±0.21	**40.28** ±0.41	**0.598** ±0.002	**+5.80** ±0.65	**33.72** ±0.14	**+6.49** ±0.50

[8] MTL survey [Vandenhende *et al.*, TPAMI'2021] [23] 3-Ways [Hoyer *et al.*, CVPR'21] [135] PAD-Net [Xu *et al.*, CVPR'18]

Fully supervised

	Methods	'S-D' Semseg ↑ mIoU %	'S-D' Depth ↓ RMSE m	'S-D' Delta ↑ Δ_{SD} %	'S-D-N' Semseg ↑ mIoU %	'S-D-N' Depth ↓ RMSE m	'S-D-N' Delta ↑ Δ_{SD} %	'S-D-N' Normals ↓ mErr. °	'S-D-N' Delta ↑ Δ_{SDN} %
NYUDv2	STL [8]	38.70 ±0.10	0.635 ±0.013	↴	*idem*	*idem*	↴	36.90 ±0.26	↴
	MTL [8]	39.44 ±0.34	0.638 ±0.004	+1.63 ±0.37	39.90 ±0.41	0.642 ±0.003	+1.89 ±0.67	36.07 ±0.09	+1.76 ±0.53
	PAD-Net [135]	35.30 ±0.84	0.659 ±0.004	-5.36 ±0.83	36.14 ±0.30	0.660 ±0.006	-4.32 ±0.68	36.72 ±0.08	-2.97 ±0.43
	3-ways$_{\text{PAD-Net}}$ [23]	**39.47** ±0.16	0.622 ±0.001	+2.90 ±0.23	**40.28** ±0.30	0.619 ±0.004	+4.16 ±0.50	35.35 ±0.09	+3.93 ±0.27
	Ours	38.93 ±0.35	**0.604** ±0.004	**+3.54** ±0.21	**40.28** ±0.41	**0.598** ±0.002	**+5.80** ±0.65	**33.72** ±0.14	**+6.49** ±0.50

[8] MTL survey [Vandenhende *et al.*, TPAMI'2021]   [23] 3-Ways [Hoyer *et al.*, CVPR'21]   [135] PAD-Net [Xu *et al.*, CVPR'18]

Fully supervised

NYUDv2

Methods	'S-D' Semseg ↑ mIoU %	Depth ↓ RMSE m	Delta ↑ Δ_{SD} %	'S-D-N' Semseg ↑ mIoU %	Depth ↓ RMSE m	Delta ↑ Δ_{SD} %	Normals ↓ mErr. °	Delta ↑ Δ_{SDN} %	'S-D-N-E' Semseg ↑ mIoU %	Depth ↓ RMSE m	Normals ↓ mErr. °	Delta ↑ Δ_{SDN} %	Edges ↑ F1 %	Delta ↑ Δ_{SDNE} %
STL [8]	38.70 ±0.10	0.635 ±0.013	↴	*idem*	*idem*	↴	36.90 ±0.26	↴	*idem*	*idem*	*idem*	↴	54.90 ±0.00	↴
MTL [8]	39.44 ±0.34	0.638 ±0.004	+1.63 ±0.37	39.90 ±0.41	0.642 ±0.003	+1.89 ±0.67	36.07 ±0.09	+1.76 ±0.53	39.70 ±0.35	0.636 ±0.001	36.10 ±0.12	+1.88 ±0.33	55.11 ±0.15	+1.50 ±0.20
PAD-Net [135]	35.30 ±0.84	0.659 ±0.004	-5.36 ±0.83	36.14 ±0.30	0.660 ±0.006	-4.32 ±0.68	36.72 ±0.08	-2.97 ±0.43	36.19 ±0.24	0.662 ±0.005	36.58 ±0.06	-2.92 ±0.37	54.79 ±0.07	-2.24 ±0.26
3-ways$_{\text{PAD-Net}}$ [23]	**39.47** ±0.16	0.622 ±0.001	+2.90 ±0.23	**40.28** ±0.30	0.619 ±0.004	+4.16 ±0.50	35.35 ±0.09	+3.93 ±0.27	40.16 ±0.28	0.614 ±0.010	35.25 ±0.09	+4.14 ±0.65	59.66 ±0.16	+5.27 ±0.49
Ours	38.93 ±0.35	**0.604** ±0.004	**+3.54** ±0.21	**40.28** ±0.41	**0.598** ±0.002	**+5.80** ±0.65	**33.72** ±0.14	**+6.49** ±0.50	**40.84** ±0.37	**0.593** ±0.004	**33.38** ±0.19	**+7.52** ±0.27	**61.12** ±0.24	**+8.47** ±0.12

[8] MTL survey [Vandenhende *et al.*, TPAMI'2021]   [23] 3-Ways [Hoyer *et al.*, CVPR'21]   [135] PAD-Net [Xu *et al.*, CVPR'18]

Fully supervised

NYUDv2

Methods	'S-D' Semseg ↑ mIoU %	'S-D' Depth ↓ RMSE m	'S-D' Delta ↑ Δ_{SD} %	'S-D-N' Semseg ↑ mIoU %	'S-D-N' Depth ↓ RMSE m	'S-D-N' Delta ↑ Δ_{SD} %	'S-D-N' Normals ↓ mErr. °	'S-D-N' Delta ↑ Δ_{SDN} %	'S-D-N-E' Semseg ↑ mIoU %	'S-D-N-E' Depth ↓ RMSE m	'S-D-N-E' Normals ↓ mErr. °	'S-D-N-E' Delta ↑ Δ_{SDN} %	'S-D-N-E' Edges ↑ F1 %	'S-D-N-E' Delta ↑ Δ_{SDNE} %
STL [8]	38.70 ±0.10	0.635 ±0.013	↴	*idem*	*idem*	↴	36.90 ±0.26	↴	*idem*	*idem*	*idem*	↴	54.90 ±0.00	↴
MTL [8]	39.44 ±0.34	0.638 ±0.004	+1.63 ±0.37	39.90 ±0.41	0.642 ±0.003	+1.89 ±0.67	36.07 ±0.09	+1.76 ±0.53	39.70 ±0.35	0.636 ±0.001	36.10 ±0.12	+1.88 ±0.33	55.11 ±0.15	+1.50 ±0.20
PAD-Net [135]	35.30 ±0.84	0.659 ±0.004	-5.36 ±0.83	36.14 ±0.30	0.660 ±0.006	-4.32 ±0.68	36.72 ±0.08	-2.97 ±0.43	36.19 ±0.24	0.662 ±0.005	36.58 ±0.06	-2.92 ±0.37	54.79 ±0.07	-2.24 ±0.26
3-ways$_{\text{PAD-Net}}$ [23]	**39.47** ±0.16	0.622 ±0.001	+2.90 ±0.23	**40.28** ±0.30	0.619 ±0.004	+4.16 ±0.50	35.35 ±0.09	+3.93 ±0.27	40.16 ±0.28	0.614 ±0.010	35.25 ±0.09	+4.14 ±0.65	59.66 ±0.16	+5.27 ±0.49
Ours	38.93 ±0.35	**0.604** ±0.004	**+3.54** ±0.21	**40.28** ±0.41	**0.598** ±0.002	**+5.80** ±0.65	**33.72** ±0.14	**+6.49** ±0.50	**40.84** ±0.37	**0.593** ±0.004	**33.38** ±0.19	**+7.52** ±0.27	**61.12** ±0.24	**+8.47** ±0.12

[8] MTL survey [Vandenhende *et al.*, TPAMI'2021] [23] 3-Ways [Hoyer *et al.*, CVPR'21] [135] PAD-Net [Xu *et al.*, CVPR'18]

Fully supervised

	Methods	'S-D' Semseg ↑ mIoU %	Depth ↓ RMSE m	Delta ↑ Δ_{SD} %	'S-D-N' Semseg ↑ mIoU %	Depth ↓ RMSE m	Delta ↑ Δ_{SD} %	Normals ↓ mErr. °	Delta ↑ Δ_{SDN} %
Synthia	STL [8]	67.43 ±0.15	5.379 ±0.055	↴	*idem*	*idem*	↴	19.61 ±0.12	↴
	MTL [8]	69.83 ±0.25	5.166 ±0.063	+03.76 ±0.77	71.27 ±0.21	5.108 ±0.076	+05.37 ±0.83	18.51 ±0.10	+05.45 ±0.72
	PAD-Net [135]	70.87 ±0.15	4.917 ±0.014	+06.85 ±0.24	72.27 ±0.25	4.949 ±0.072	+07.58 ±0.56	19.28 ±0.09	+05.62 ±0.43
	3-ways$_{\text{PAD-Net}}$ [23]	77.50 ±0.17	4.289 ±0.028	+17.60 ±0.13	79.93 ±0.5	4.218 ±0.082	+20.06 ±0.92	15.54 ±0.14	+20.29 ±0.84
	Ours	**80.53** ±0.43	**4.161** ±0.022	**+21.04** ±0.52	**82.99** ±0.38	**4.056** ±0.076	**+23.83** ±0.98	**14.30** ±0.15	**+24.92** ±0.87
VKITTI2	STL [8]	84.53 ±0.06	5.720 ±0.027	↴	*idem*	*idem*	↴	23.14 ±0.68	↴
	MTL [8]	87.73 ±0.12	5.720 ±0.029	+01.89 ±0.21	87.83 ±0.21	5.714 ±0.033	+02.00 ±0.27	22.30 ±0.68	+02.54 ±0.80
	PAD-Net [135]	88.43 ±0.12	5.571 ±0.058	+03.63 ±0.45	88.67 ±0.15	5.543 ±0.043	+04.09 ±0.29	22.16 ±0.70	+04.09 ±0.83
	3-ways$_{\text{PAD-Net}}$ [23]	96.13 ±0.15	4.013 ±0.051	+21.78 ±0.54	96.87 ±0.06	3.756 ±0.013	+24.46 ±0.14	15.54 ±0.56	+27.25 ±0.90
	Ours	**97.00** ±0.10	**3.423** ±0.025	**+27.47** ±0.16	**97.53** ±0.06	**3.089** ±0.006	**+30.70** ±0.05	**14.44** ±0.52	**+33.00** ±0.73
Cityscapes	STL [8]	67.93 ±0.06	6.622 ±0.020	↴	*idem*	*idem*	↴	44.10 ±0.01	↴
	MTL [8]	70.43 ±0.12	6.797 ±0.520	+00.52 ±0.32	70.93 ±0.15	6.736 ±0.023	+01.34 ±0.28	43.60 ±0.01	+01.30 ±0.18
	PAD-Net [135]	70.23 ±0.25	6.777 ±0.010	+00.52 ±0.27	70.67 ±0.06	6.755 ±0.018	+01.00 ±0.17	43.52 ±0.00	+01.12 ±0.11
	3-ways$_{\text{PAD-Net}}$ [23]	**75.00** ±0.10	**6.528** ±0.063	**+05.91** ±0.44	75.50 ±0.10	6.491 ±0.081	+06.56 ±0.61	41.84 ±0.05	+06.09 ±0.37
	Ours	74.95 ±0.10	6.649 ±0.003	+04.96 ±0.08	**76.08** ±0.14	**6.407** ±0.013	**+07.61** ±0.04	**40.05** ±0.33	**+08.15** ±0.22

[8] MTL survey [Vandenhende *et al.*, TPAMI'2021] [23] 3-Ways [Hoyer *et al.*, CVPR'21] [135] PAD-Net [Xu *et al.*, CVPR'18]

mTEB Scales 4	3	2	1	‘*S-D*’ Param. ↓ #M added	Semseg ↑ mIoU %	Depth ↓ RMSE m	Delta ↑ Δ_{SD} %	‘*S-D-N*’ Param. ↓ #M added	Semseg ↑ mIoU %	Depth ↓ RMSE m	Normals ↓ mErr. °	Delta ↑ Δ_{SDN} %
				0.00	96.88 $\pm$0.30	3.604 $\pm$0.020	25.81 $\pm$0.33	**0.00**	97.38 $\pm$0.02	3.491 $\pm$0.041	14.50 $\pm$0.57	30.51 $\pm$0.98
✓				3.09	**97.32** $\pm$0.06	3.556 $\pm$0.029	26.50 $\pm$0.29	2.32	97.43 $\pm$0.04	3.559 $\pm$0.024	14.51 $\pm$0.50	30.12 $\pm$0.79
	✓			3.09	97.24 $\pm$0.03	3.476 $\pm$0.018	27.15 $\pm$0.16	2.32	97.49 $\pm$0.08	3.353 $\pm$0.025	14.45 $\pm$0.48	31.43 $\pm$0.76
		✓		**0.77**	97.07 $\pm$0.06	3.468 $\pm$0.016	27.11 $\pm$0.12	**9.26**	97.47 $\pm$0.06	3.244 $\pm$0.035	14.57 $\pm$0.51	31.89 $\pm$0.92
			✓	**0.77**	97.00 $\pm$0.10	**3.423** $\pm$0.025	**27.47** $\pm$0.16	**9.26**	**97.53** $\pm$0.06	**3.089** $\pm$0.006	**14.44** $\pm$0.52	**33.00** $\pm$0.73
	✓		✓	**3.86**	**97.09** $\pm$0.03	**3.369** $\pm$0.022	**27.99** $\pm$0.18	**11.58**	**97.53** $\pm$0.02	**3.080** $\pm$0.025	**14.47** $\pm$0.57	**33.02** $\pm$0.90
	✓	✓	✓	4.63	97.01 $\pm$0.02	3.377 $\pm$0.008	27.88 $\pm$0.06	13.89	97.39 $\pm$0.02	3.136 $\pm$0.046	14.81 $\pm$0.77	32.13 $\pm$1.39
✓	✓	✓	✓	7.72	97.05 $\pm$0.03	**3.369** $\pm$0.010	27.97 $\pm$0.08	23.15	96.82 $\pm$0.23	3.307 $\pm$0.066	15.39 $\pm$0.65	30.08 $\pm$1.39

VKITTI2

Where should tasks talk ?

weights ω_S	ω_D	Semseg ↑ mIoU %	Depth ↓ RMSE m	Delta ↑ Δ_{SD} %
1	1	83.83 ±0.15	5.713 ±0.060	-0.35 ±0.47
1	10	79.87 ±0.21	<u>5.708</u> ±0.036	-2.66 ±0.40
10	1	86.20 ±0.71	**5.693** ±0.055	+1.30 ±0.22
50	1	87.73 ±0.12	5.720 ±0.029	**+1.89** ±0.21
100	1	88.00 ±0.20	5.754 ±0.030	<u>+1.75</u> ±0.17
100	10	86.13 ±0.32	**5.693** ±0.039	+1.18 ±0.45
200	1	<u>88.13</u> ±0.12	5.790 ±0.055	+1.52 ±0.45
500	1	**88.17** ±0.15	5.847 ±0.043	+1.04 ±0.30

(a) 'S-D' gridsearch

weights ω_S	ω_D	ω_N	Semseg ↑ mIoU %	Depth ↓ RMSE m	Normals ↓ mErr. °	Delta ↑ Δ_{SDN} %
1	1	1	83.50 ±0.20	5.707 ±0.058	23.03 ±0.70	-0.17 ±0.64
10	1	1	86.53 ±0.21	<u>5.694</u> ±0.032	22.98 ±0.68	+1.17 ±0.93
10	1	10	86.63 ±0.21	**5.675** ±0.050	22.61 ±0.70	+1.85 ±0.80
50	1	1	87.73 ±0.21	5.706 ±0.051	22.90 ±0.71	+1.69 ±0.81
50	1	10	87.77 ±0.15	5.714 ±0.065	22.56 ±0.69	+2.15 ±0.86
50	1	50	87.73 ±0.21	5.701 ±0.062	22.37 ±0.70	+2.49 ±0.76
100	1	1	88.03 ±0.15	5.746 ±0.030	22.95 ±0.69	+1.49 ±0.92
100	1	10	87.97 ±0.15	5.714 ±0.048	22.59 ±0.69	+2.19 ±0.79
100	1	50	88.00 ±0.20	5.717 ±0.048	22.40 ±0.71	+2.45 ±0.99
100	1	100	88.07 ±0.15	5.696 ±0.038	**22.29** ±0.70	**+2.75** ±1.04
150	1	10	<u>88.10</u> ±0.20	5.752 ±0.059	22.59 ±0.70	+2.01 ±0.86
150	1	50	<u>88.10</u> ±0.20	5.738 ±0.039	22.41 ±0.70	+2.35 ±0.99
150	1	100	**88.13** ±0.15	5.732 ±0.037	<u>22.31</u> ±0.71	<u>+2.54</u> ±0.94

(b) 'S-D-N' gridsearch

MTL for Segmentation

{S,D} on Cityscapes

Methods	road	swalk	build	wall	fence	pole	light	sign	veg	sky	person	rider	car	bus	mbike	bike	mIoU %
3-ways$_{\text{PAD-Net}}$ [23]	97.21	79.38	90.50	**47.68**	49.68	51.17	49.41	64.65	91.40	93.85	72.41	46.92	92.66	80.17	42.43	66.39	69.74
3-ways$_{\text{mTEB}}$	**97.62**	**82.29**	**92.44**	46.52	**54.76**	**59.82**	**60.94**	**73.13**	**92.22**	**94.55**	**76.40**	**58.49**	**94.26**	**85.14**	**49.41**	**71.70**	**74.36**

MDE for depth

Self-supervised depth can help segmentation

Cross-Modal Learning

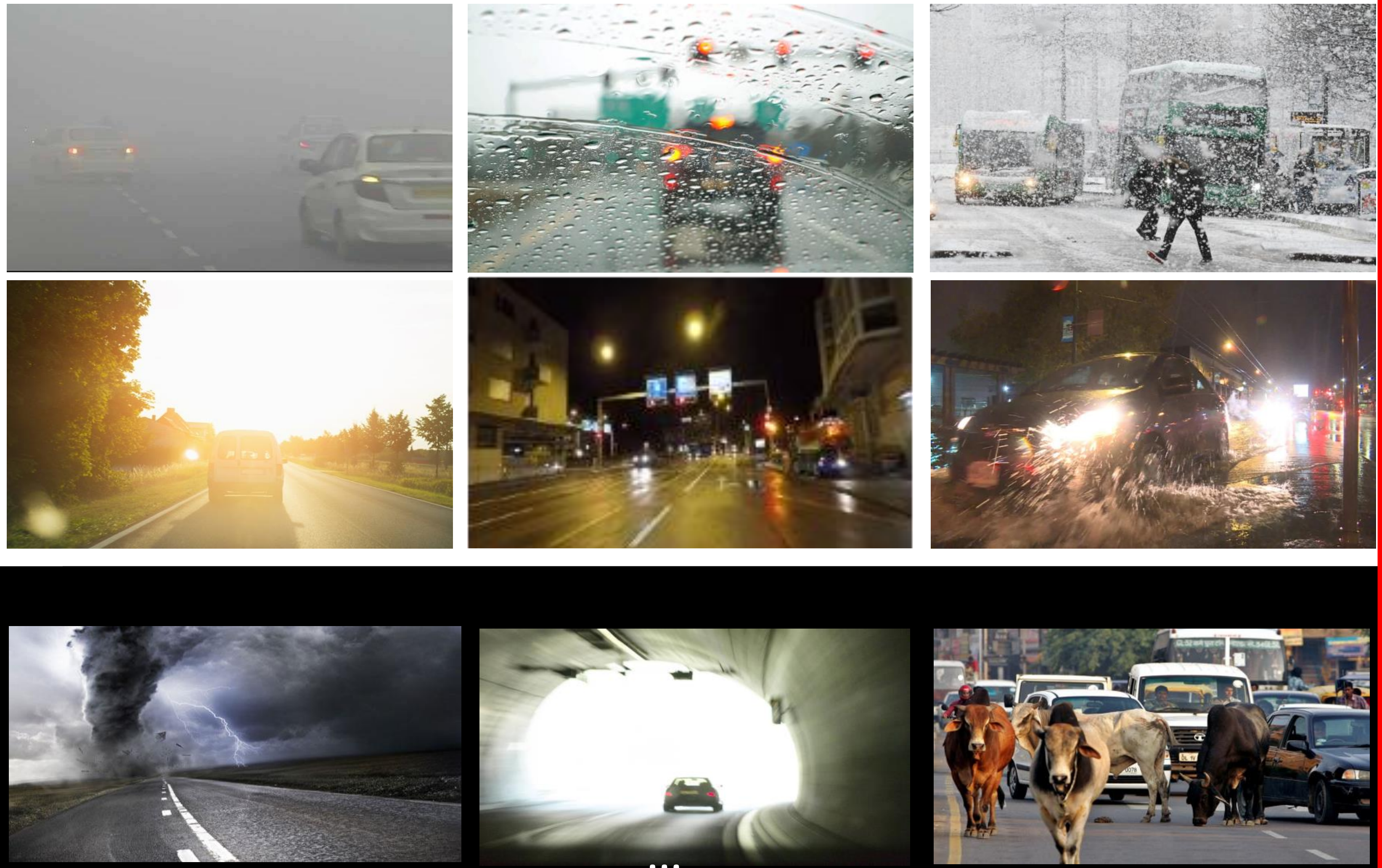
Dataset Bias or Domain Discrepancy

(slide Tuan-Hung Vu)

Cityscapes (CVPR 16)

Mapillary Vistas (ICCV 17)

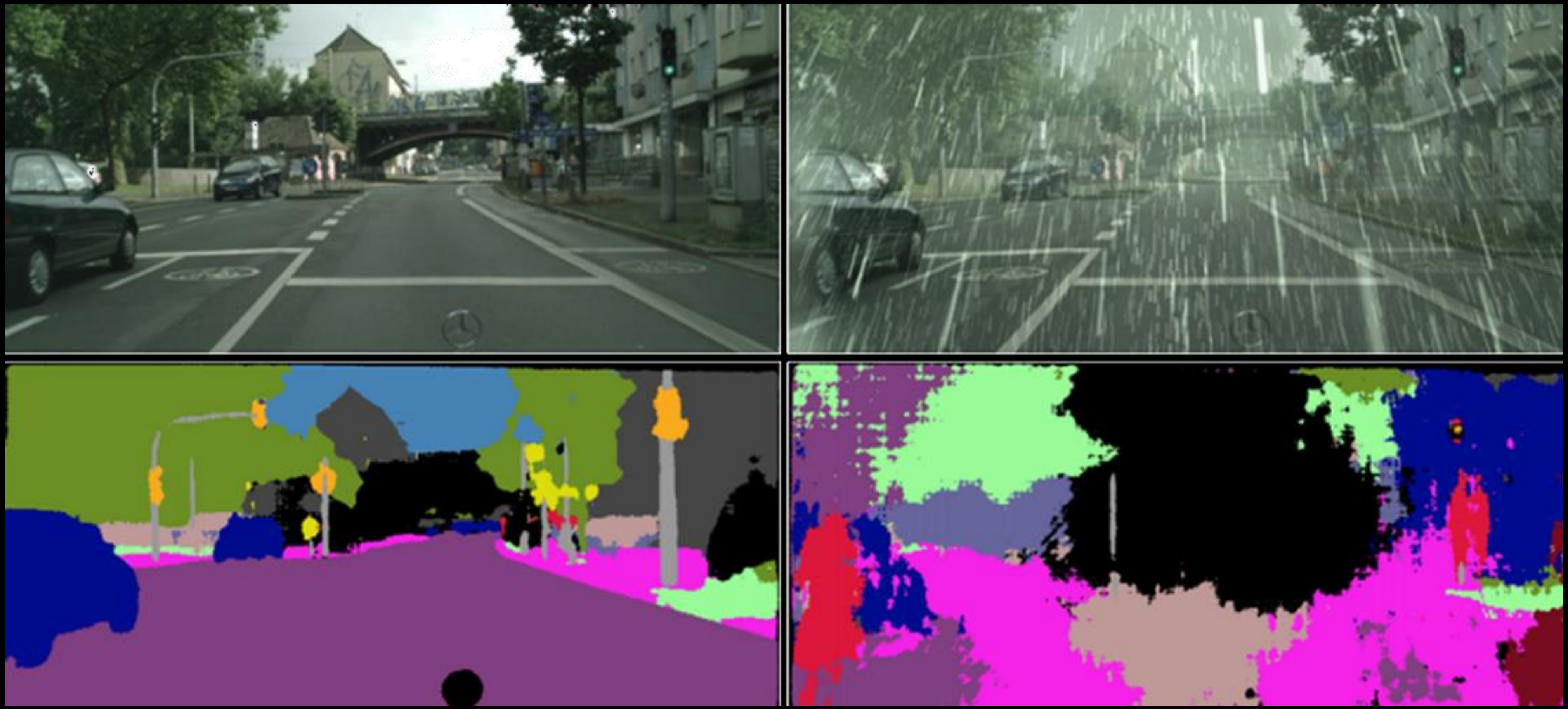
Common train/test



...

Open-world testbed

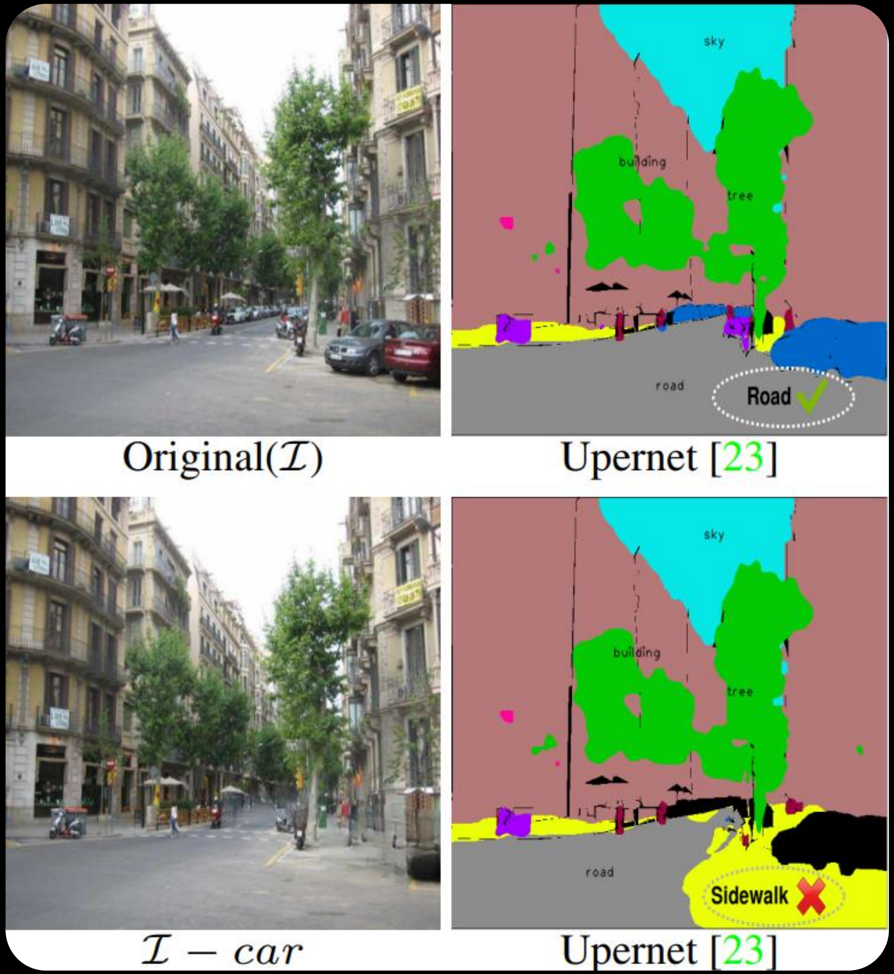
Out of Distribution



Physics-Based Rendering for (..) Rain. (Halder et al., ICCV 19)

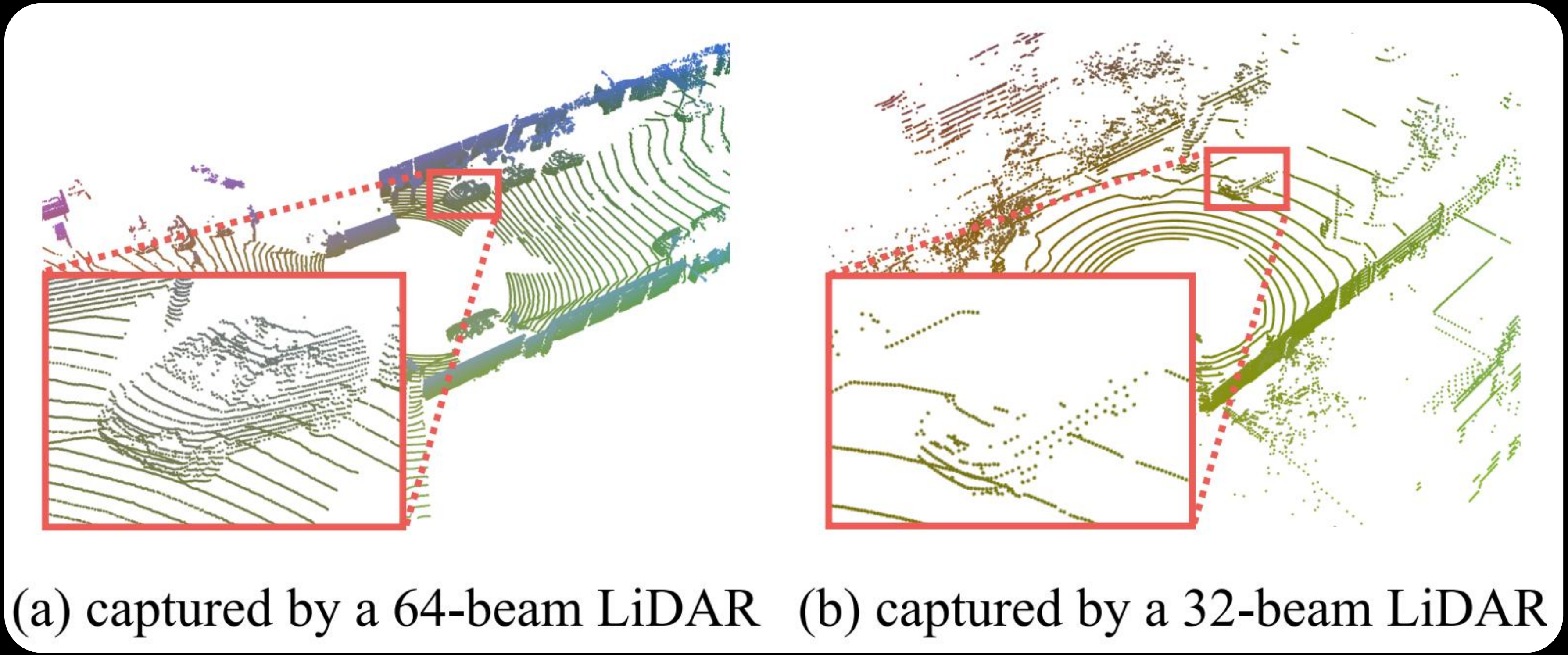
Out of Distribution

Out of Distribution

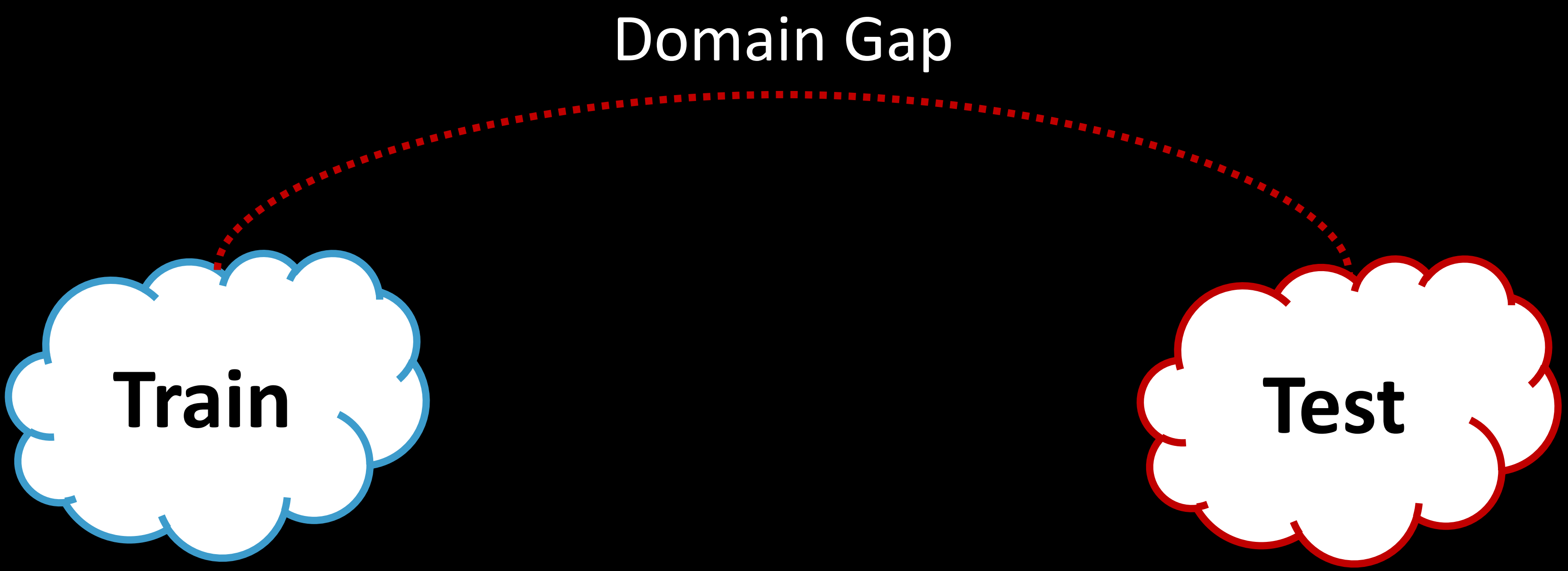


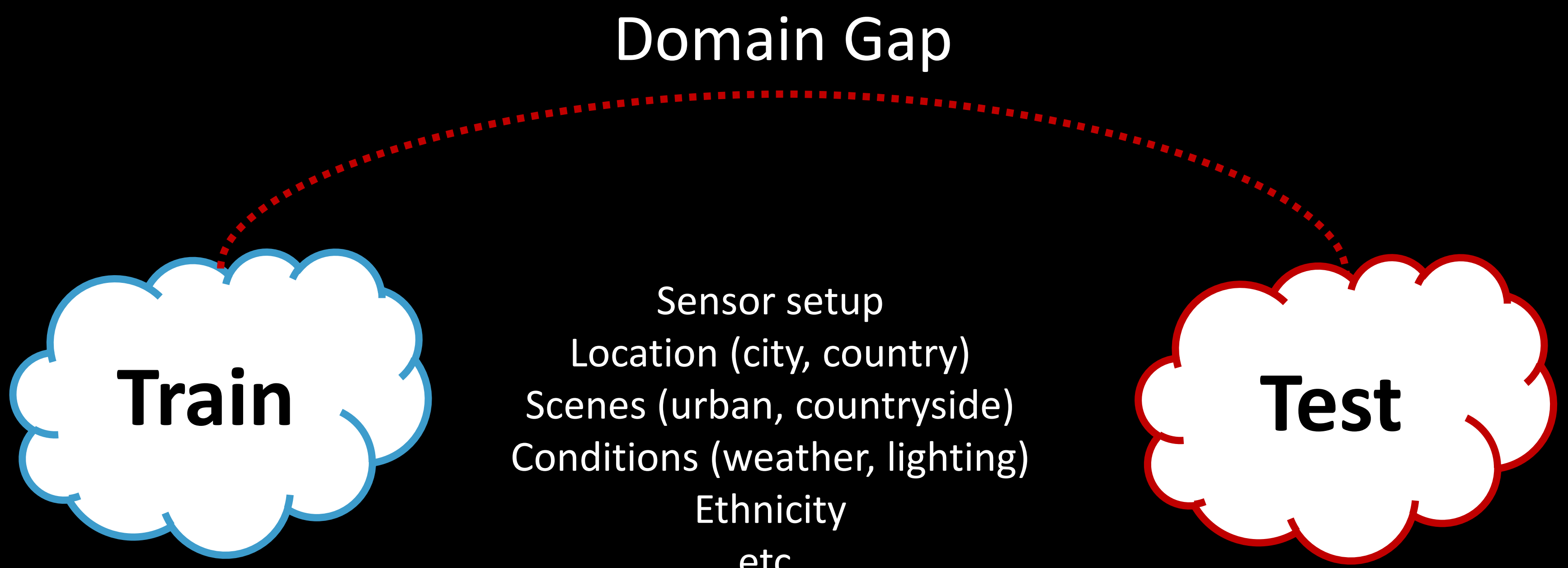
(Shetty et al. CVPR 2019)

Out of Distribution



Complete & Label. (Yi et al. CVPR 2021)


Domain Gap
Train
Test

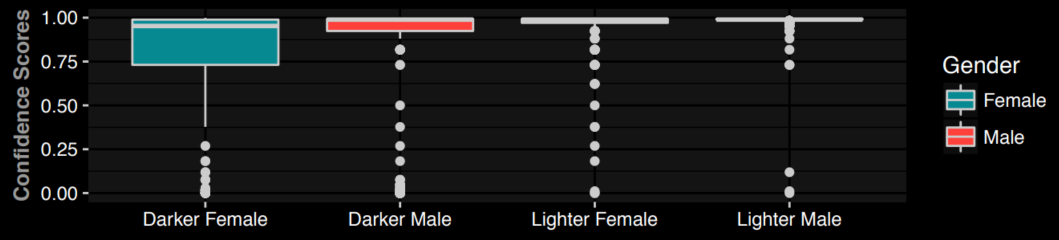

Domain Gap
Train
Sensor setup
Location (city, country)
Scenes (urban, countryside)
Conditions (weather, lighting)
Ethnicity
etc.
Test

Domain Gap

Train

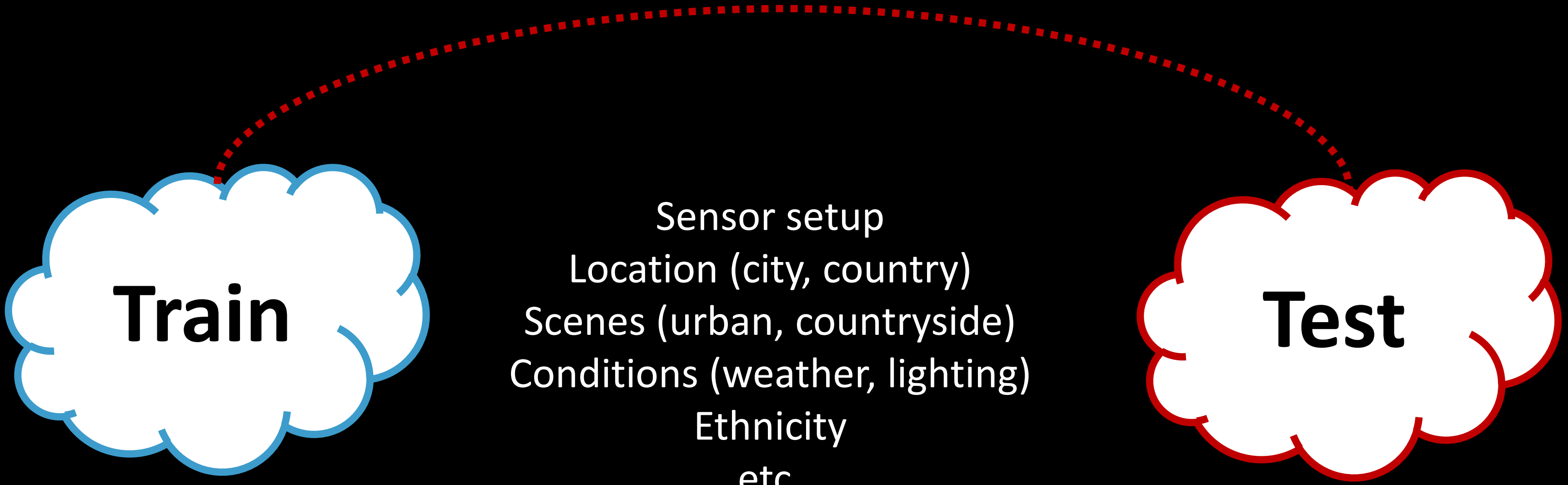
Sensor setup
Location (city, country)
Scenes (urban, countryside)
Conditions (weather, lighting)
Ethnicity
etc.

Test

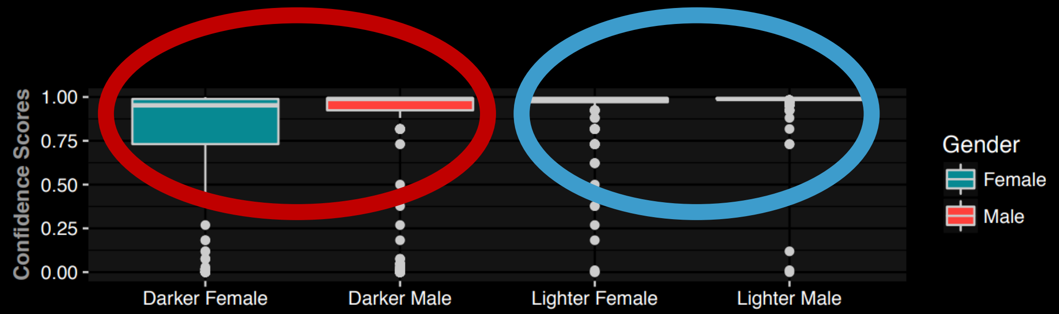


Buolamwini and Gebru. FAccT 2018

Domain Gap

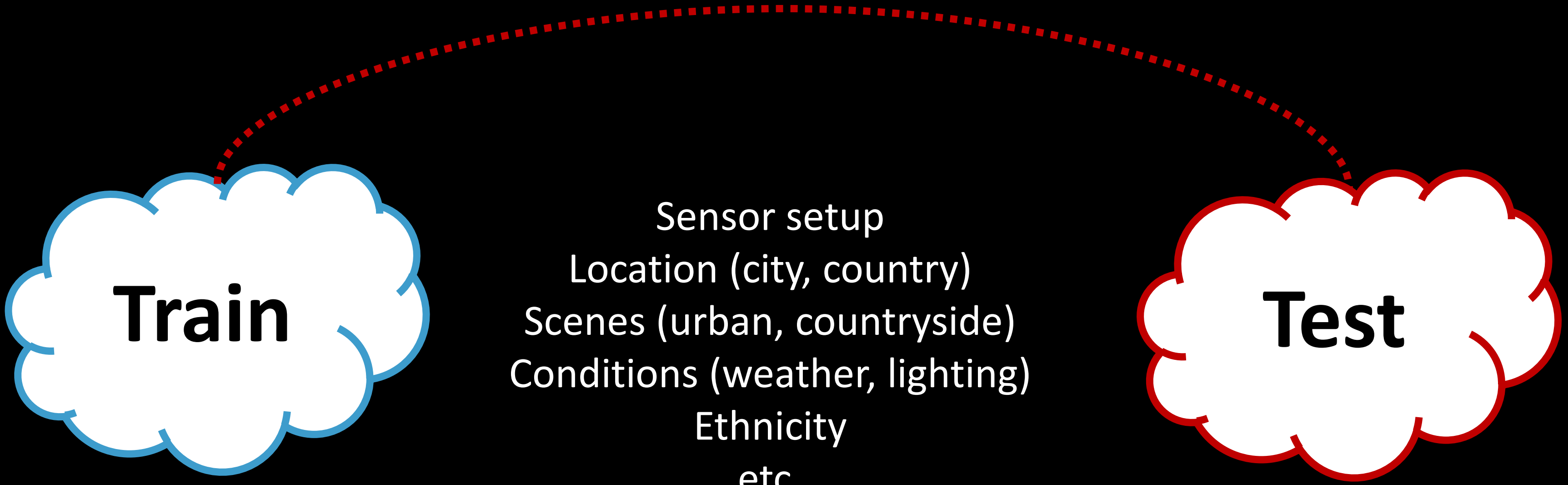


Sensor setup
Location (city, country)
Scenes (urban, countryside)
Conditions (weather, lighting)
Ethnicity
etc.

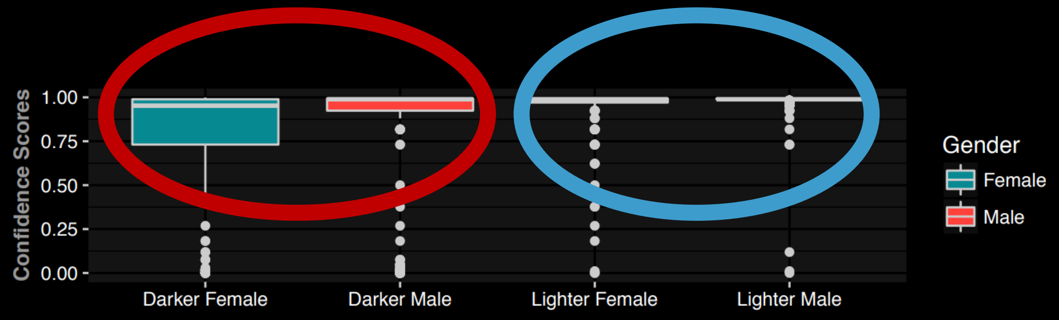


Buolamwini and Gebru. FAccT 2018

Domain Gap



Sensor setup
Location (city, country)
Scenes (urban, countryside)
Conditions (weather, lighting)
Ethnicity
etc.



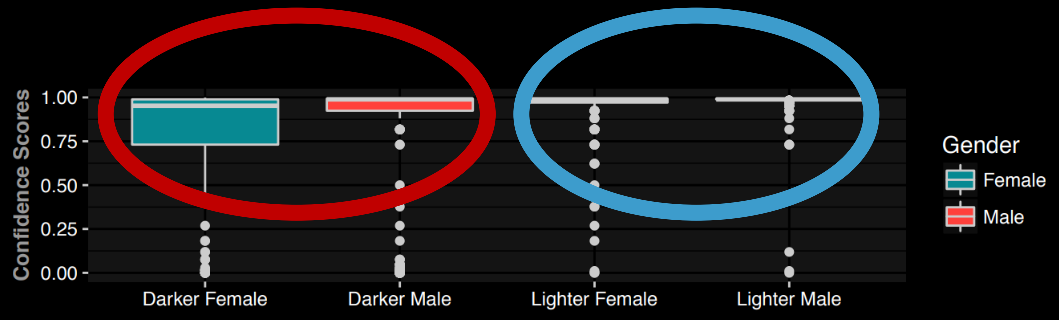
Buolamwini and Gebru. FAccT 2018

Source\Target	KITTI	Argoverse	nuScenes	Lyft	Waymo
KITTI	**88.0 / 82.5**	55.8 / 27.7	47.4 / 13.3	81.7 / 51.8	45.2 / 11.9
Argoverse	69.5 / 33.9	**79.2 / 57.8**	52.5 / 21.8	86.9 / 67.4	83.8 / 40.2
nuScenes	49.7 / 13.4	73.2 / 21.8	**73.4 / 38.1**	89.0 / 38.2	78.8 / 36.7
Lyft	74.3 / 39.4	77.1 / 45.8	63.5 / 23.9	**90.2 / 87.3**	87.0 / 64.7
Waymo	51.9 / 13.1	76.4 / 42.6	55.5 / 21.6	87.9 / 74.5	**90.1 / 85.3**

Wang et al. CVPR 2020

Domain Gap

Sensor setup
Location (city, country)
Scenes (urban, countryside)
Conditions (weather, lighting)
Ethnicity
etc.



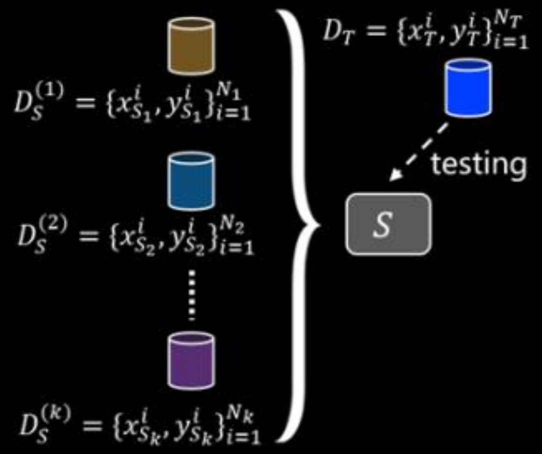
Buolamwini and Gebru. FAccT 2018

		-32	-41	-7	-3
Source\Target	KITTI	Argoverse	nuScenes	Lyft	Waymo
KITTI	**88.0 / 82.5**	55.8 / 27.7	47.4 / 13.3	81.7 / 51.8	45.2 / 11.9
Argoverse	69.5 / 33.9	**79.2 / 57.8**	52.5 / 21.8	86.9 / 67.4	83.8 / 40.2
nuScenes	49.7 / 13.4	73.2 / 21.8	**73.4 / 38.1**	89.0 / 38.2	78.8 / 36.7
Lyft	74.3 / 39.4	77.1 / 45.8	63.5 / 23.9	**90.2 / 87.3**	87.0 / 64.7
Waymo	51.9 / 13.1	76.4 / 42.6	55.5 / 21.6	87.9 / 74.5	**90.1 / 85.3**

Wang et al. CVPR 2020

Domain Adaptation

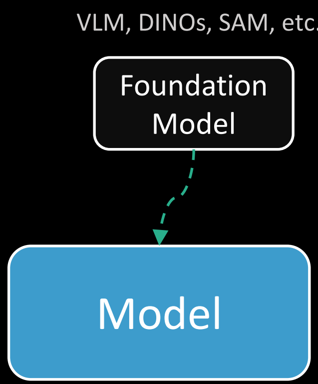
Source: He et al. 2022



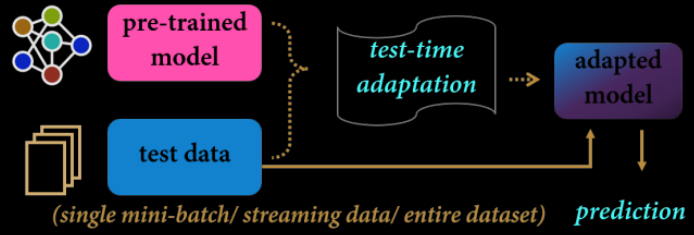
Domain Generalization

Source: Frikha et al. 2022

…



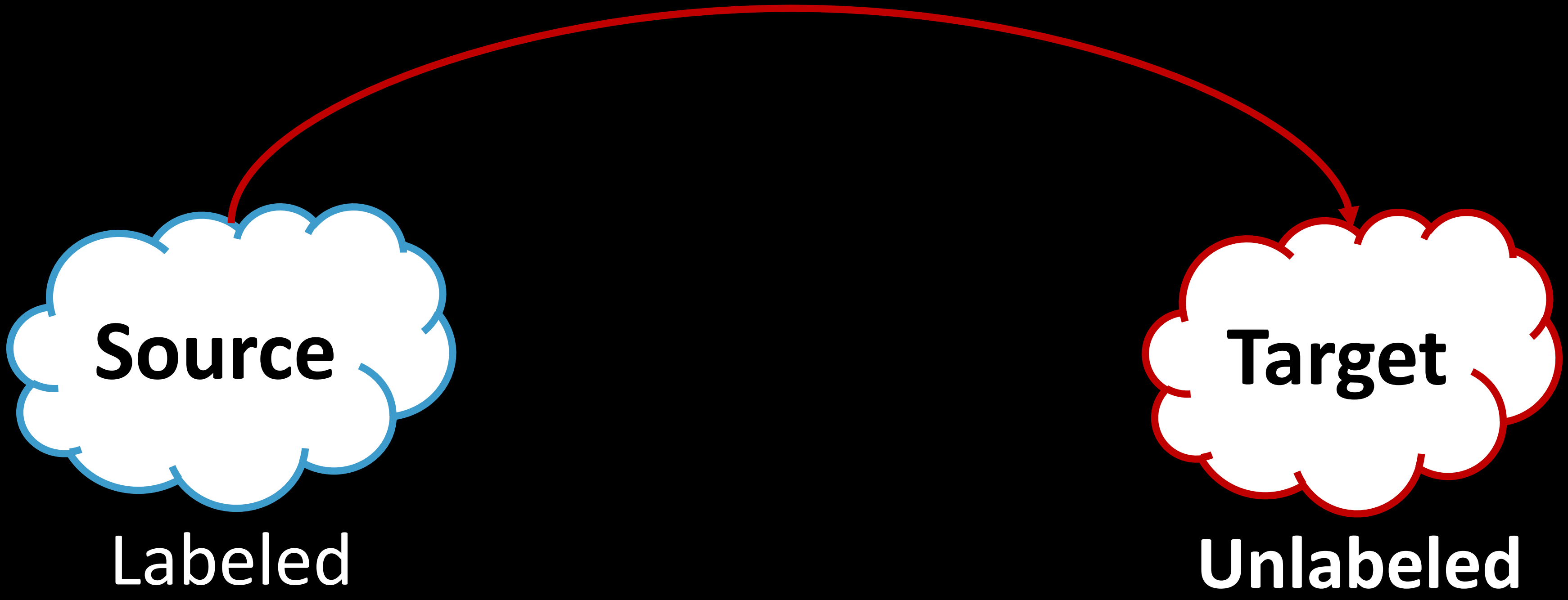
Knowledge distillation

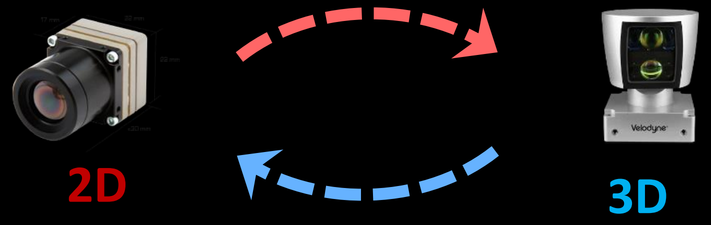


Test time adaptation

Source: Liang et al. 2023

Unsupervised Domain Adaptation (UDA)



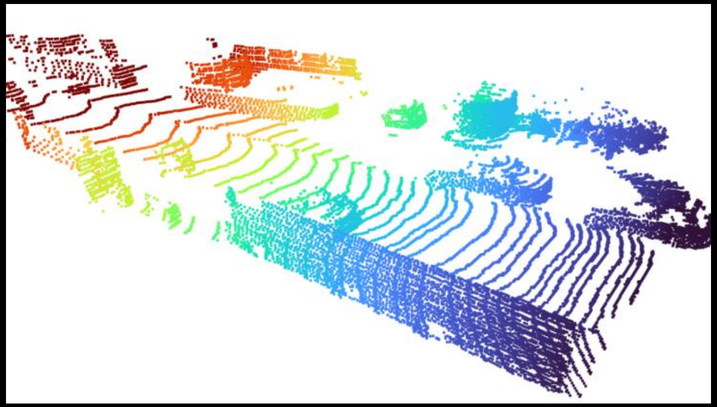


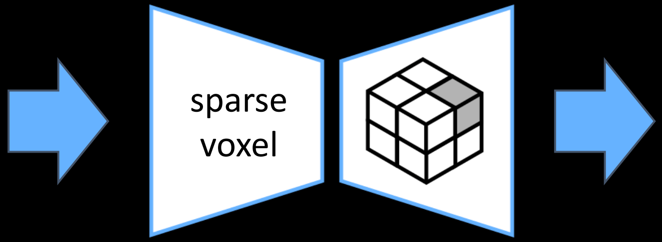
xMUDA
Cross-Modal Learning for Domain Adaptation

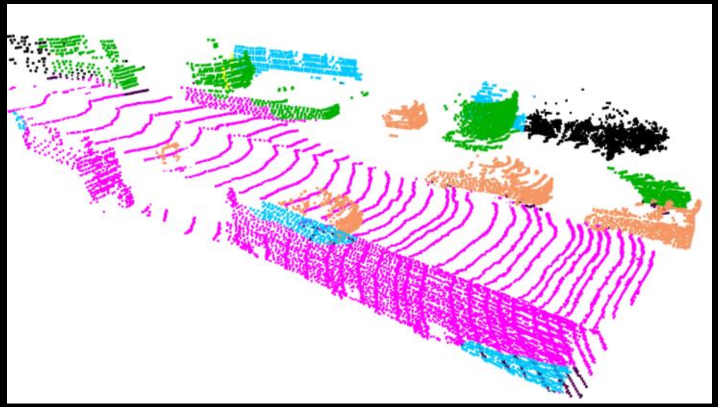
CVPR 20 & TPAMI 22

github.com/valeoai/xmuda_journal

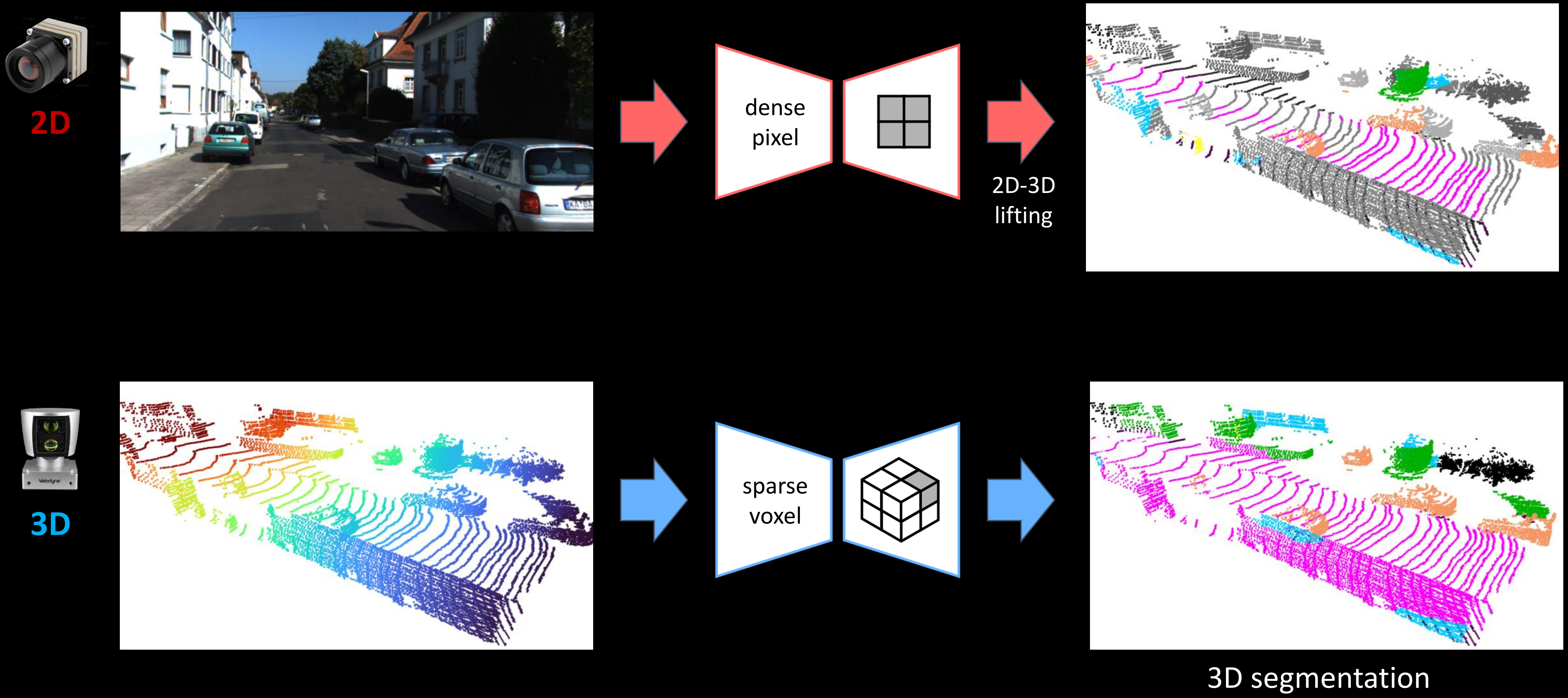
3D

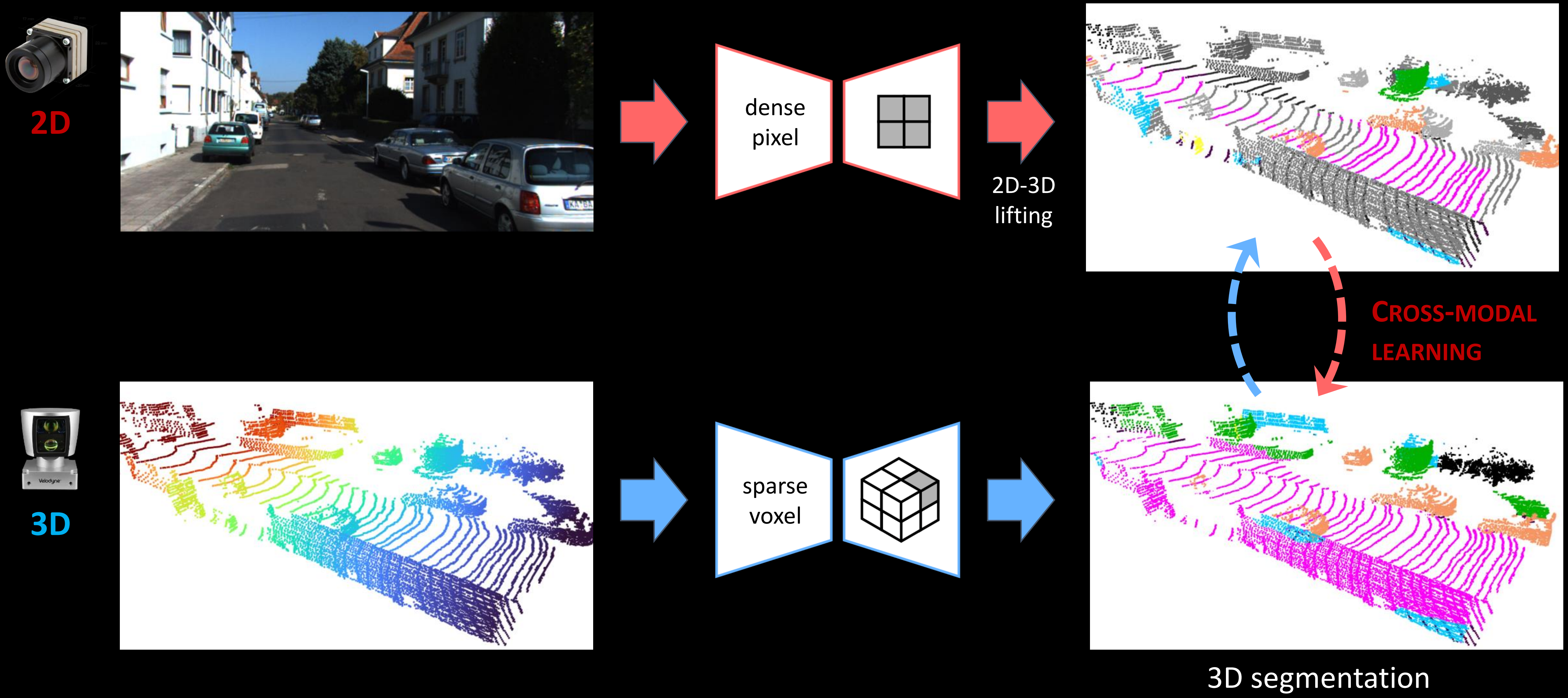


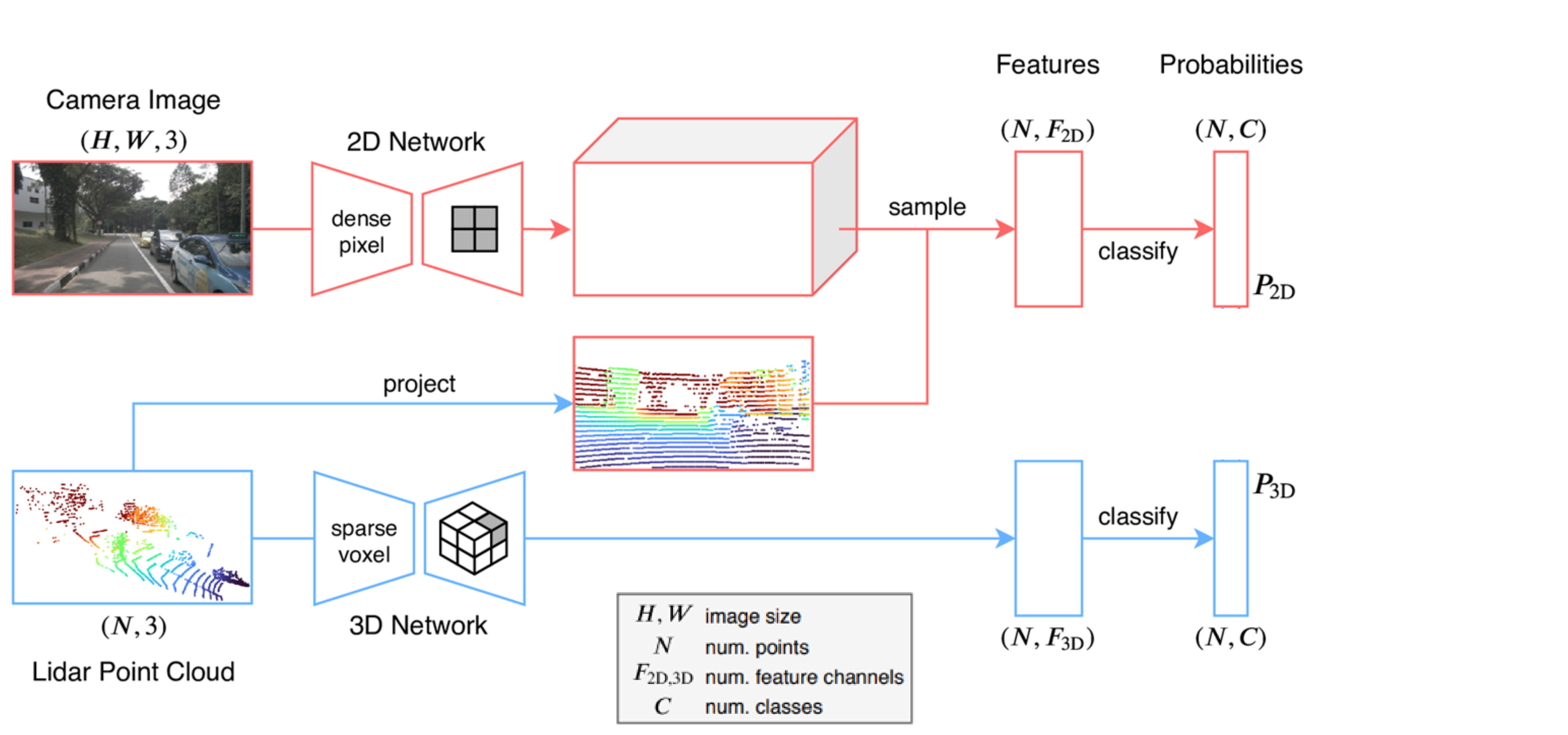


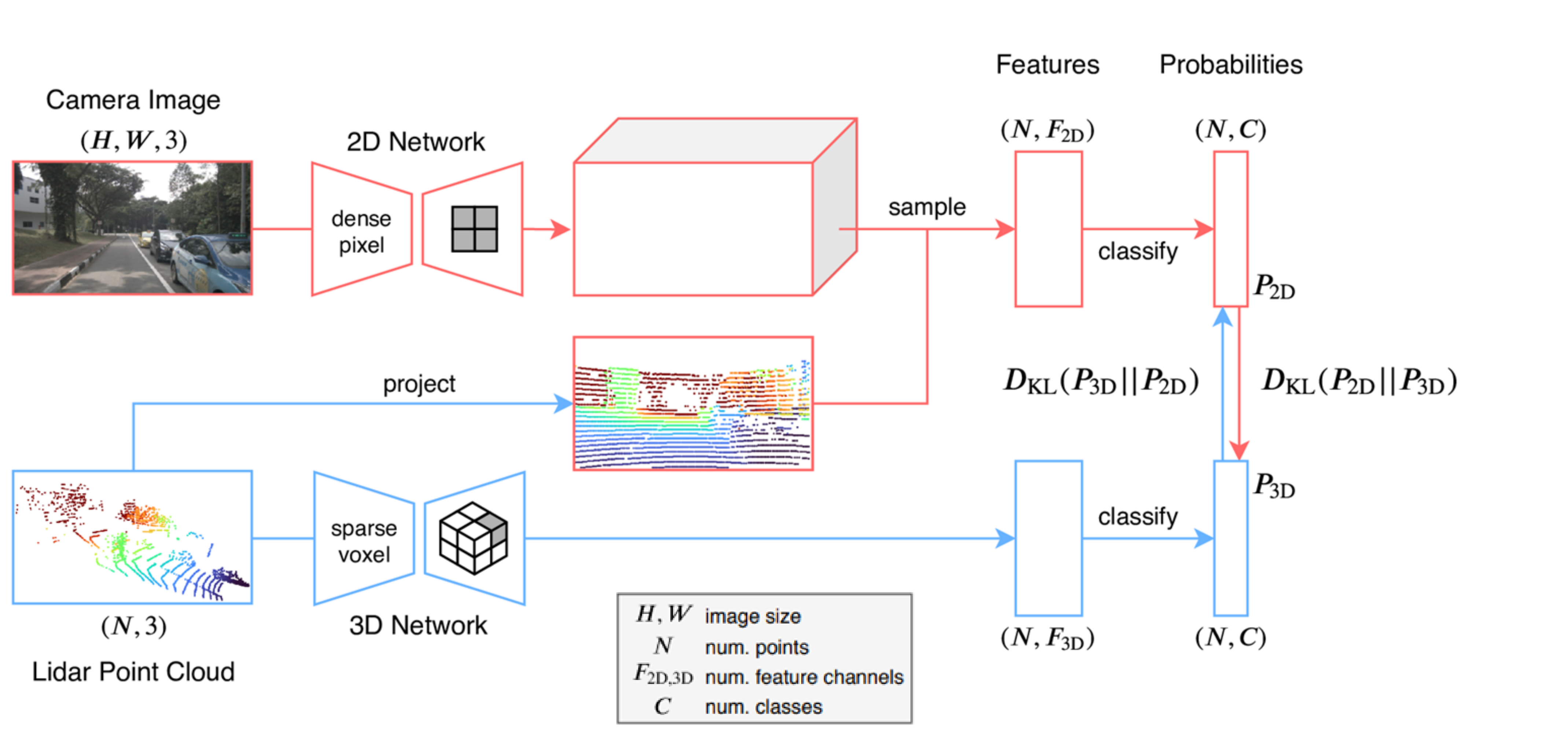


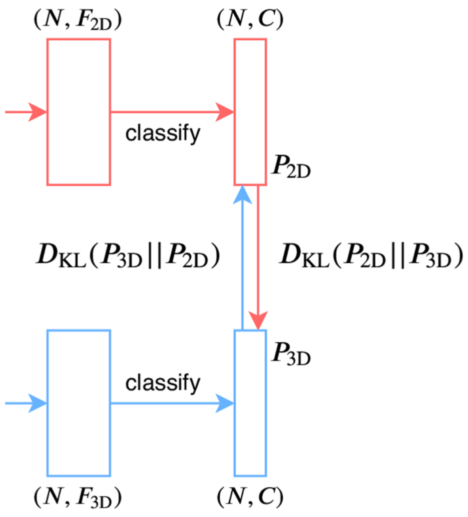
3D segmentation


2D
dense
pixel
2D-3D
lifting
3D
sparse
voxel
3D segmentation


2D
dense
pixel
2D-3D
lifting
CROSS-MODAL
LEARNING
3D
sparse
voxel
3D segmentation


Camera Image
$(H, W, 3)$
2D Network
dense pixel
Features
Probabilities
(N, F_{2D})
(N, C)
sample
classify
P_{2D}
project
P_{3D}
classify
sparse voxel
3D Network
$(N, 3)$
Lidar Point Cloud
H, W image size
N num. points
$F_{2D,3D}$ num. feature channels
C num. classes
(N, F_{3D})
(N, C)


Camera Image
$(H, W, 3)$
2D Network
dense
pixel
sample
Features
Probabilities
(N, F_{2D})
(N, C)
classify
P_{2D}
project
$D_{KL}(P_{3D}||P_{2D})$
$D_{KL}(P_{2D}||P_{3D})$
P_{3D}
sparse
voxel
classify
$(N, 3)$
Lidar Point Cloud
3D Network
H, W image size
N num. points
$F_{2D,3D}$ num. feature channels
C num. classes
(N, F_{3D})
(N, C)

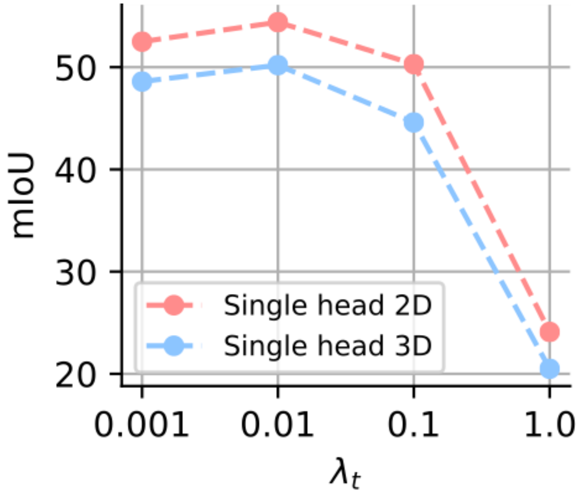


Single Head

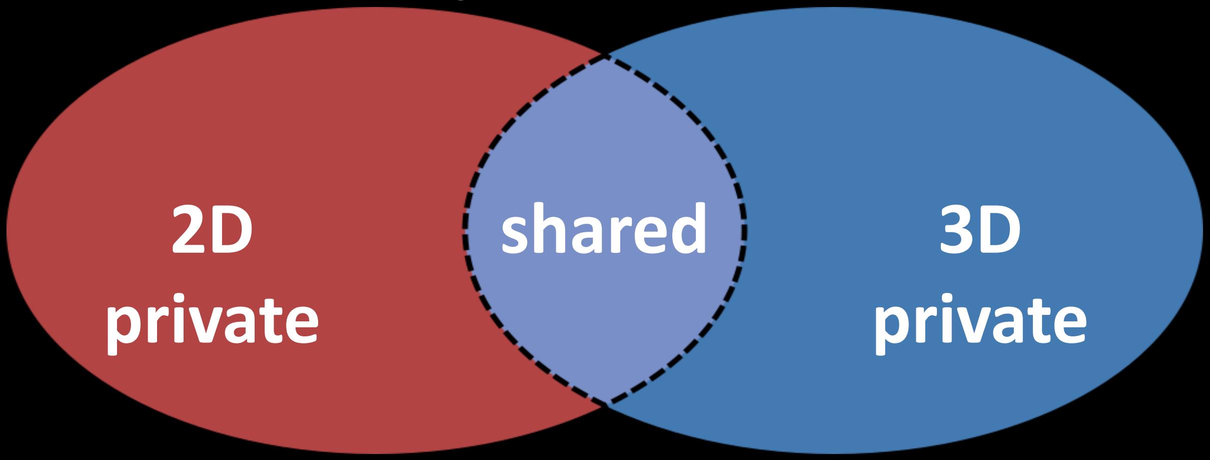
$$\mathcal{L}_{\mathrm{xM}}(\boldsymbol{x}) = \boldsymbol{D}_{\mathrm{KL}}(\boldsymbol{P}_{\boldsymbol{x}}^{(n,c)}||\boldsymbol{Q}_{\boldsymbol{x}}^{(n,c)}) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C}\boldsymbol{P}_{\boldsymbol{x}}^{(n,c)}\log\frac{\boldsymbol{P}_{\boldsymbol{x}}^{(n,c)}}{\boldsymbol{Q}_{\boldsymbol{x}}^{(n,c)}}$$

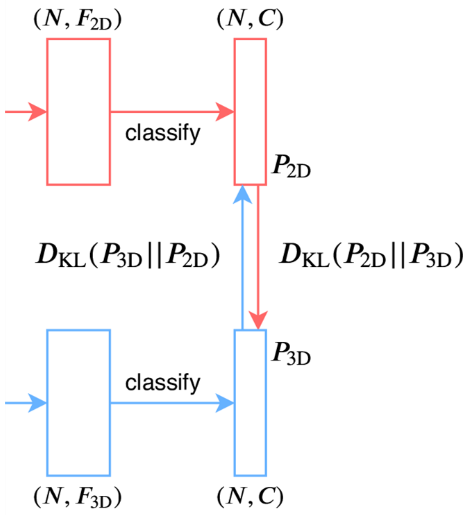
Complete objective:

$$\min_{\theta}\left[\frac{1}{|\mathcal{S}|}\sum_{\boldsymbol{x}_s\in\mathcal{S}}\left(\mathcal{L}_{\mathrm{seg}}(\boldsymbol{x}_s, \boldsymbol{y}_s^{3\mathrm{D}}) + \lambda_s\mathcal{L}_{\mathrm{xM}}(\boldsymbol{x}_s)\right) + \frac{1}{|\mathcal{T}|}\sum_{\boldsymbol{x}_t\in\mathcal{T}}\lambda_t\mathcal{L}_{\mathrm{xM}}(\boldsymbol{x}_t)\right]$$



Why does it fail ?



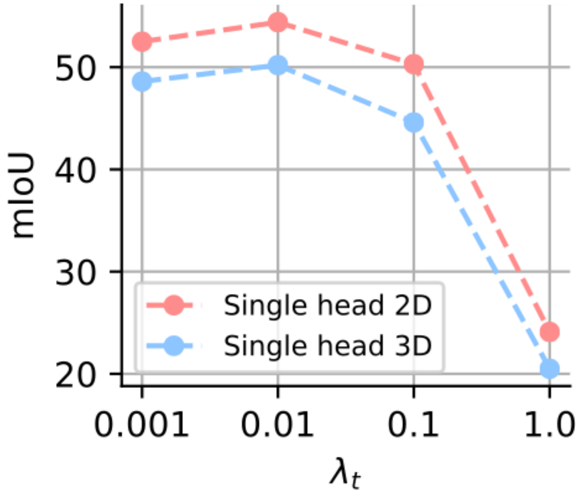


Single Head

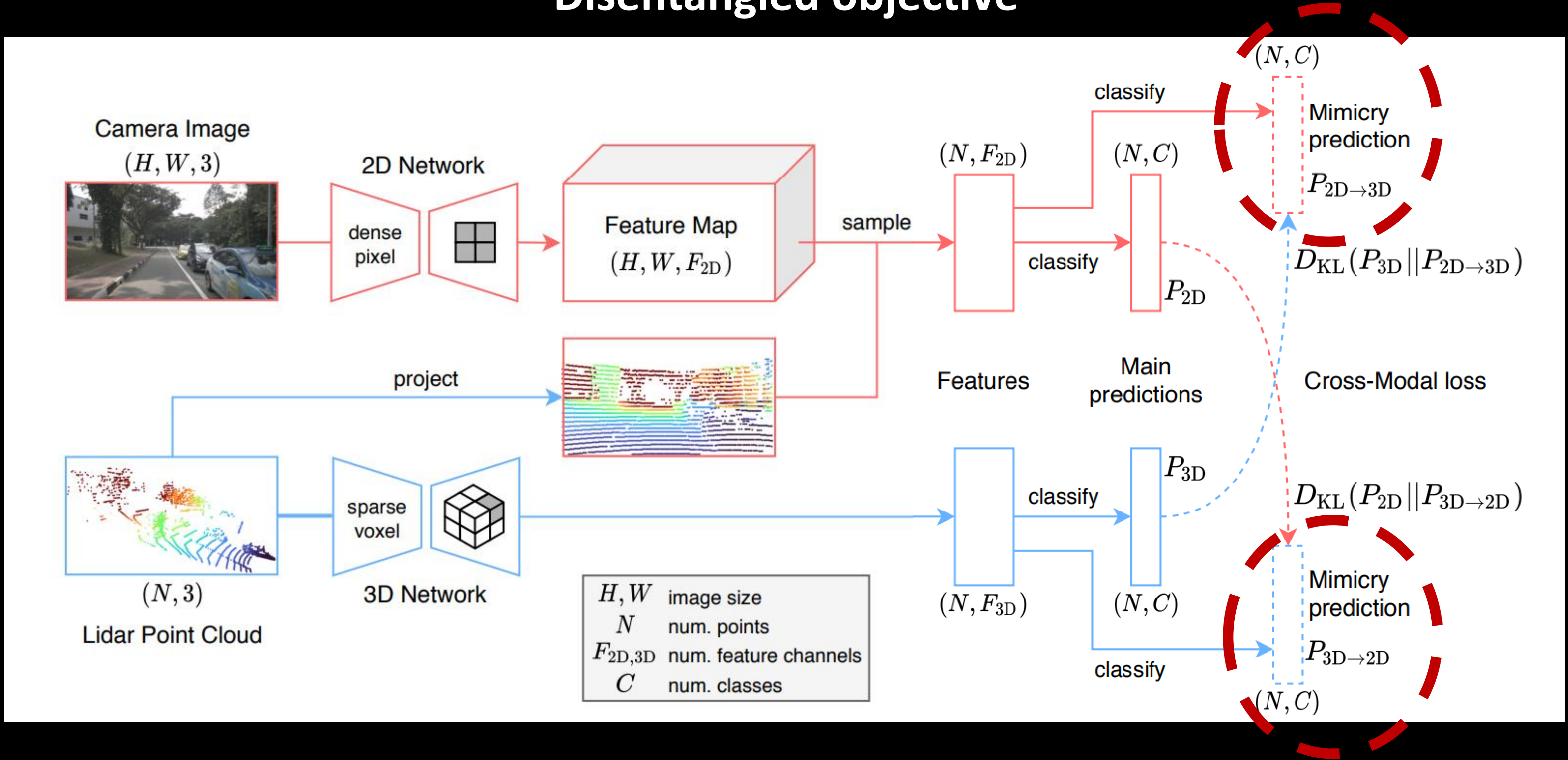
$$\mathcal{L}_{\mathrm{xM}}(\boldsymbol{x}) = \boldsymbol{D}_{\mathrm{KL}}(\boldsymbol{P}_{\boldsymbol{x}}^{(n,c)}||\boldsymbol{Q}_{\boldsymbol{x}}^{(n,c)}) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C}\boldsymbol{P}_{\boldsymbol{x}}^{(n,c)}\log\frac{\boldsymbol{P}_{\boldsymbol{x}}^{(n,c)}}{\boldsymbol{Q}_{\boldsymbol{x}}^{(n,c)}}$$

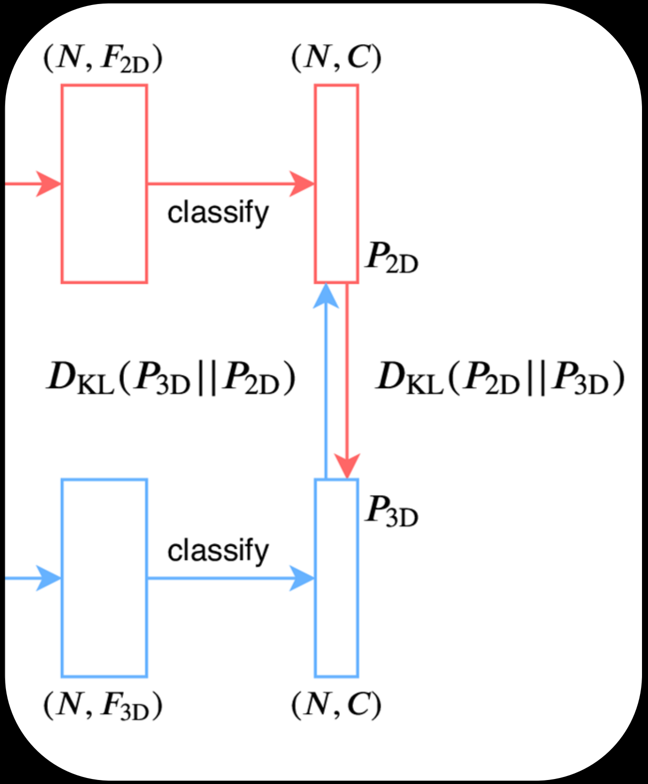
Complete objective:

$$\min_{\theta}\left[\frac{1}{|\mathcal{S}|}\sum_{\boldsymbol{x}_s\in\mathcal{S}}\left(\mathcal{L}_{\mathrm{seg}}(\boldsymbol{x}_s, \boldsymbol{y}_s^{3\mathrm{D}}) + \lambda_s\mathcal{L}_{\mathrm{xM}}(\boldsymbol{x}_s)\right) + \frac{1}{|\mathcal{T}|}\sum_{\boldsymbol{x}_t\in\mathcal{T}}\lambda_t\mathcal{L}_{\mathrm{xM}}(\boldsymbol{x}_t)\right]$$

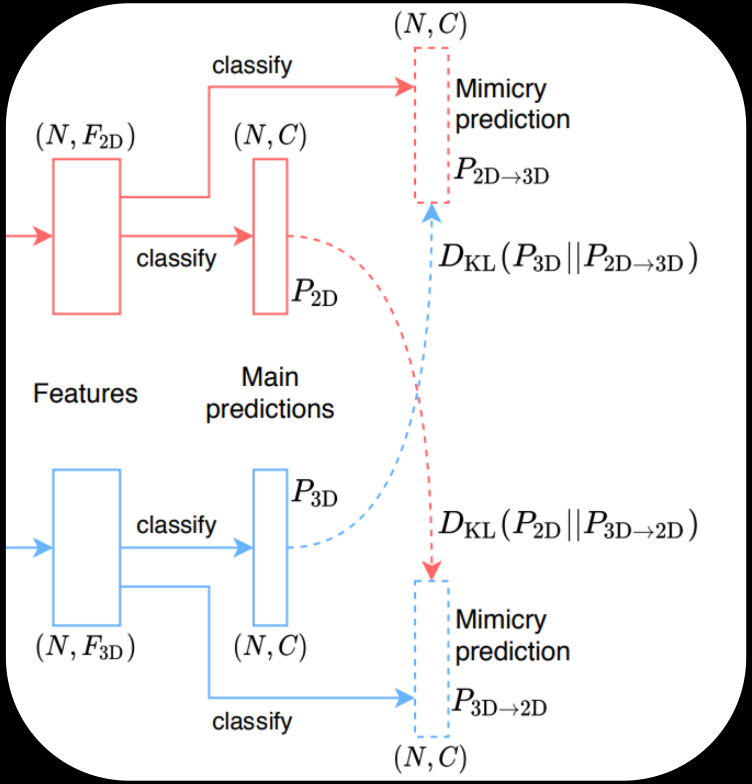


Disentangled objective

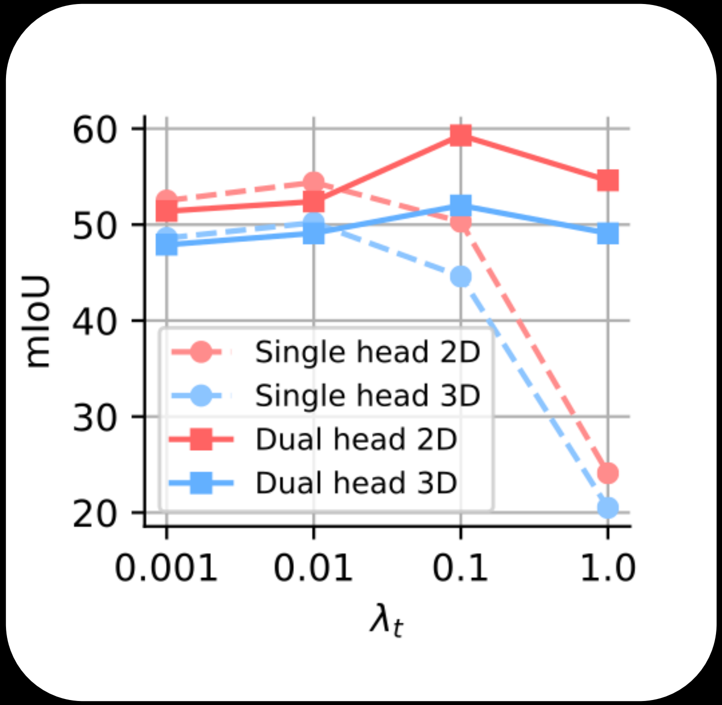


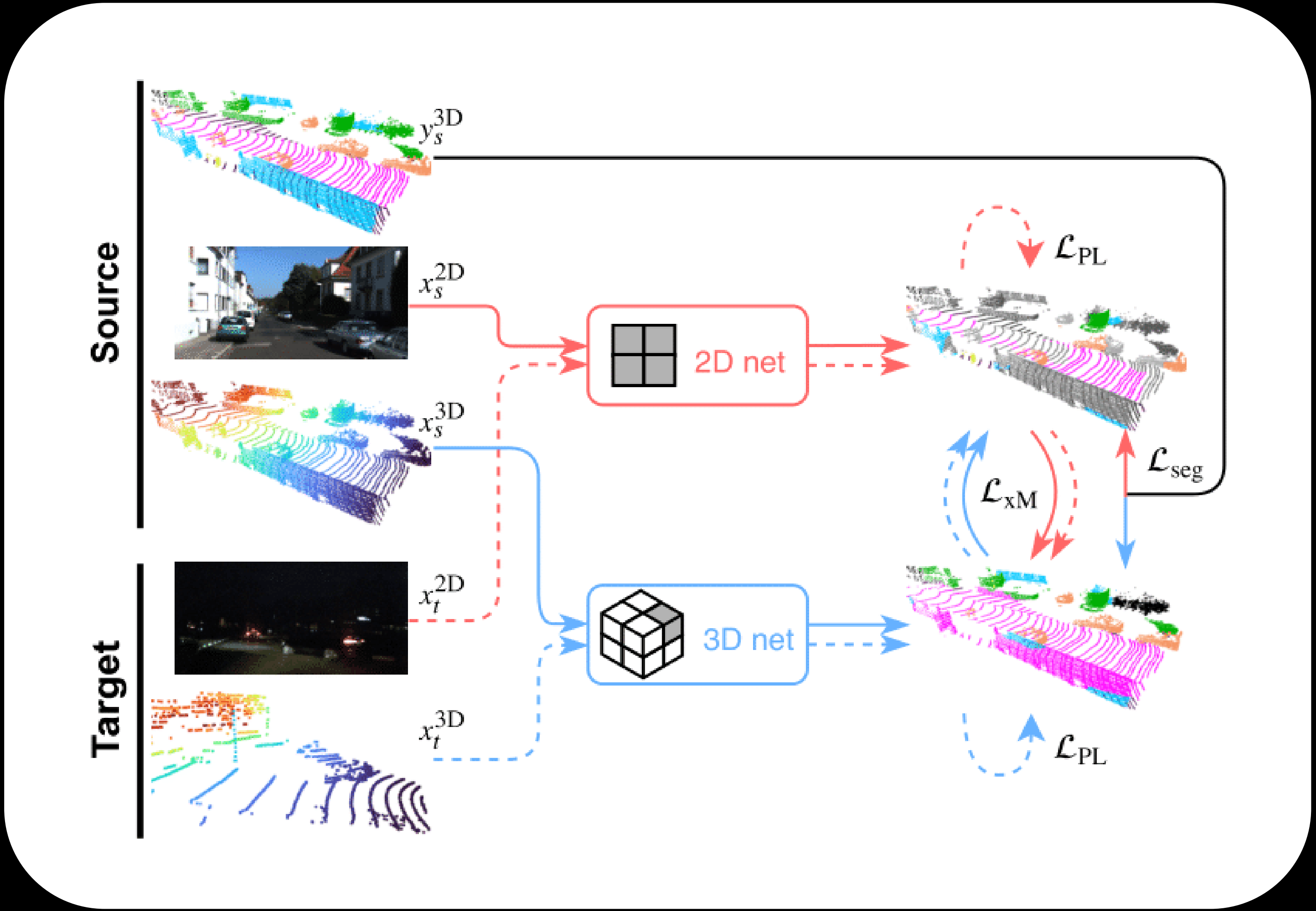


Single head



Dual head
(xMUDA)

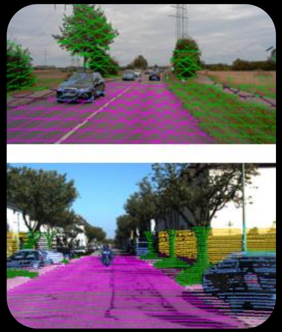




xMUDA / xMUDA$_{PL}$

Right-to-left driving | Illumination at night | Lack of Realism | Lidar density changes | Weather changes

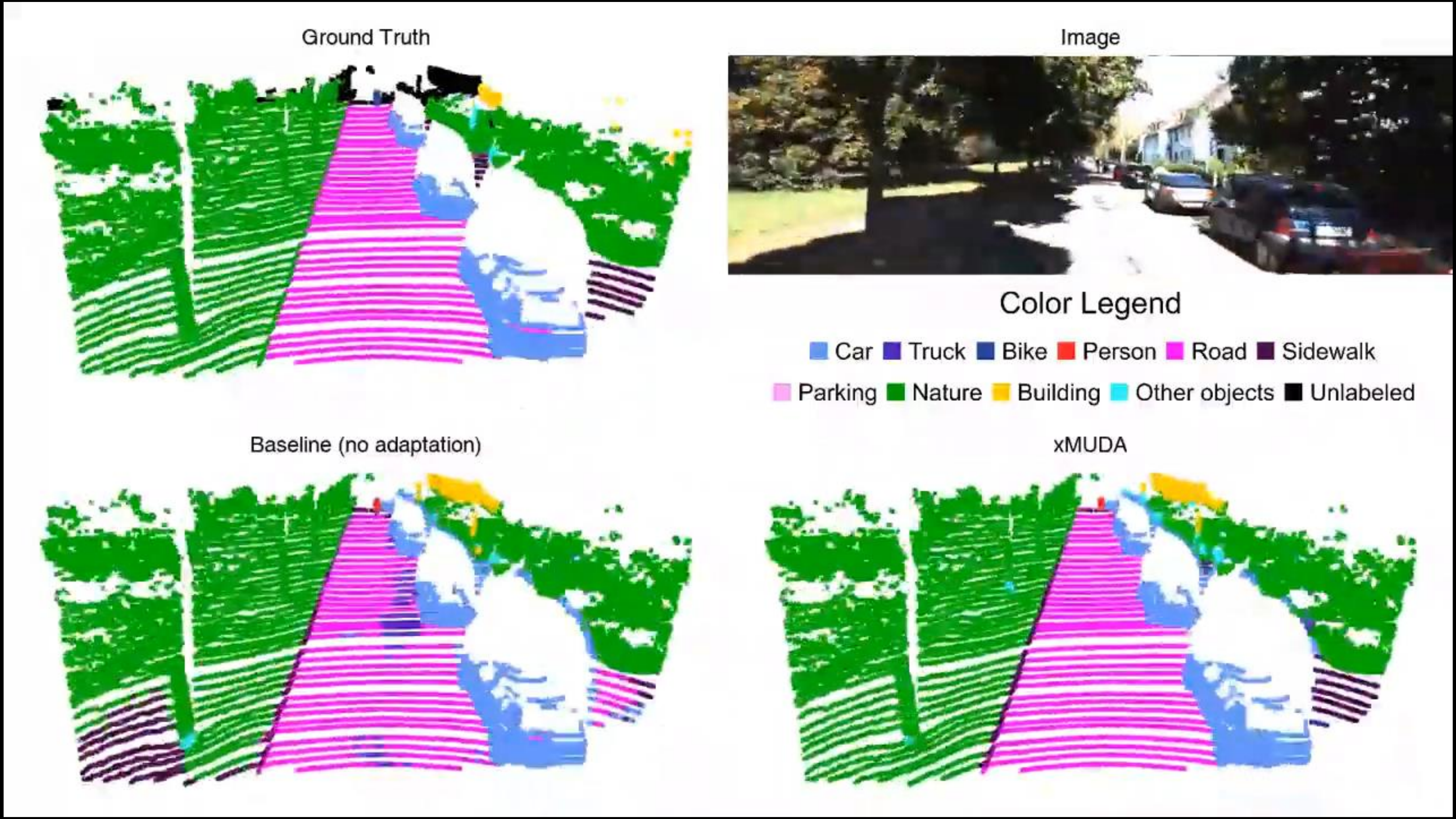
Main challenges



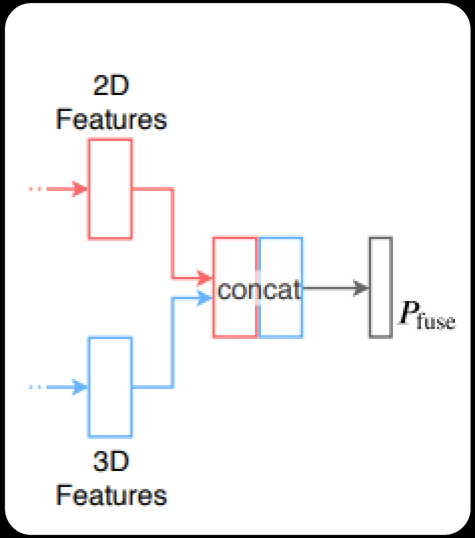
Method	nuSc-Lidarseg: USA/Singap.			nuSc-Lidarseg: Day/Night			Virt.KITTI/Sem.KITTI			A2D2/Sem.KITTI		
	2D	3D	2D+3D	2D	3D	2D+3D	2D	3D	2D+3D	2D	3D	2D+3D
Baseline (src only)	58.4	62.8	68.2	47.8	68.8	63.3	26.8	42.0	42.2	34.2	35.9	40.4
Deep logCORAL [20]	64.4	63.2	69.4	47.7	68.7	63.7	41.4*	36.8	47.0	35.1*	41.0	42.2
MinEnt [5]	57.6	61.5	66.0	47.1	68.8	63.6	39.2	43.3	47.1	37.8	39.6	42.6
PL [7]	62.0	64.8	70.4	47.0	**69.6**	63.0	21.5	44.3	35.6	34.7	41.7	45.2
FDA [14]	60.8	-	-	48.4	-	-	32.8*	-	-	37.6*	-	-
xMUDA	64.4	63.2	69.4	55.5	69.2	**67.4**	42.1	46.7	48.2	38.3	46.0	44.0
xMUDA$_{PL}$	**67.0**	**65.4**	**71.2**	**57.6**	**69.6**	64.4	**45.8**	**51.4**	**52.0**	**41.2**	**49.8**	**47.5**
Oracle	75.4	76.0	79.6	61.5	69.8	69.2	66.3	78.4	80.1	59.3	71.9	73.6

2D+3D = ensembling

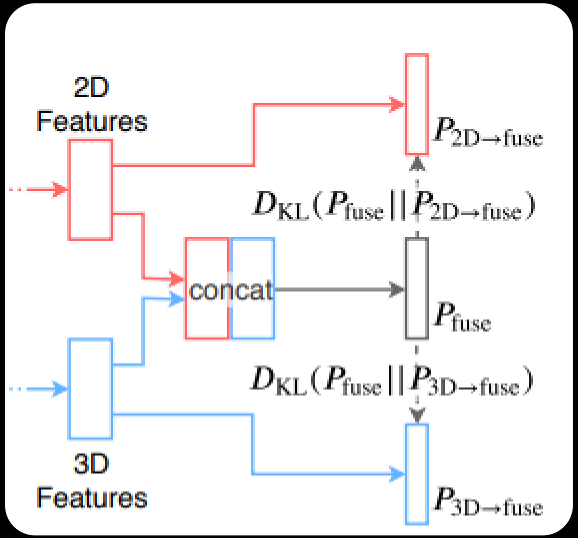
[5] MinEnt / Advent. Vu et al. CVPR 19
[7] Dong-Hyun. ICML 13
[14] Yanchao and Soatto. CVPR 20
[20] Deep LogCoral. Yifei et al. ICCV-W 17


Ground Truth
Image
Color Legend
Car
Truck
Bike
Person
Road
Sidewalk
Parking
Nature
Building
Other objects
Unlabeled
Baseline (no adaptation)
xMUDA

Can fusion benefit from mimicking ?



Vanilla Fusion

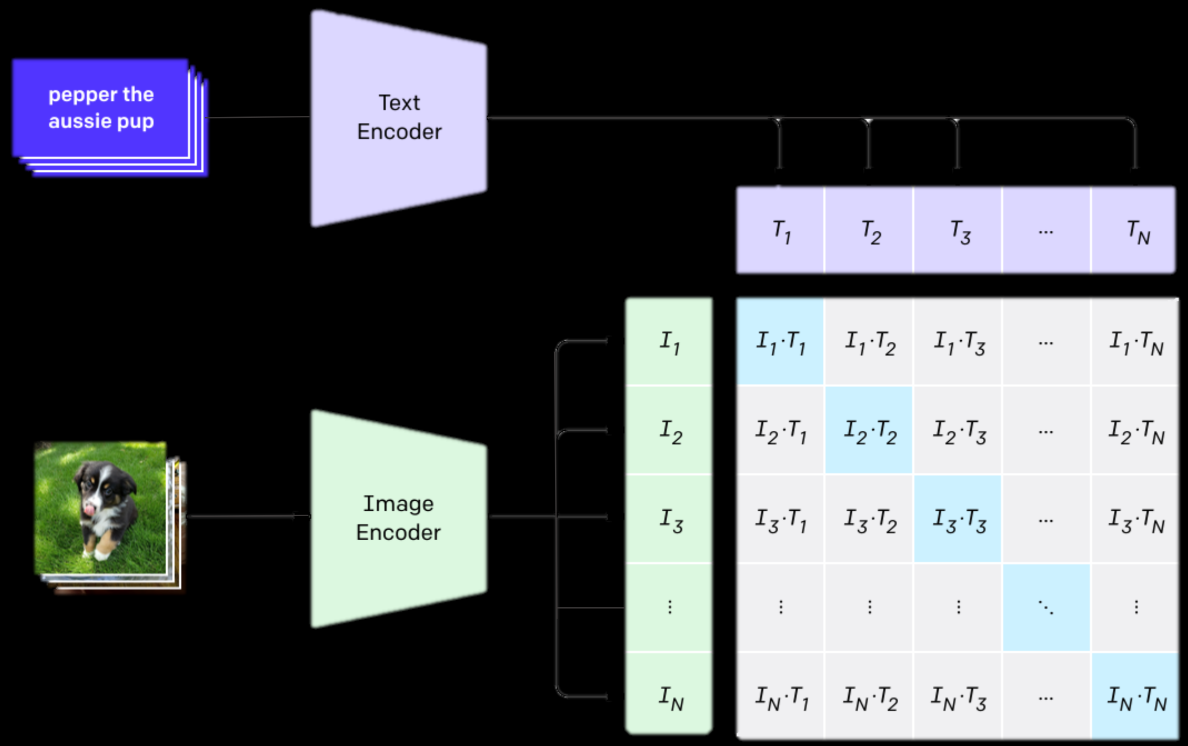


xMUDA Fusion

Method	Arch.	nuSc-Lidarseg: USA/Singap.	A2D2/ Sem.KITTI
Baseline (src only)	Vanilla	66.5	34.2
Deep logCORAL [20]	Vanilla	64.0	36.2
MinEnt [5]	Vanilla	65.4	39.8
PL [7]	Vanilla	70.1	38.6
xMUDA Fusion	xMUDA	69.3	**42.6**
xMUDA_{PL} Fusion	xMUDA	**70.7**	42.2
Oracle	xMUDA	80.6	65.7

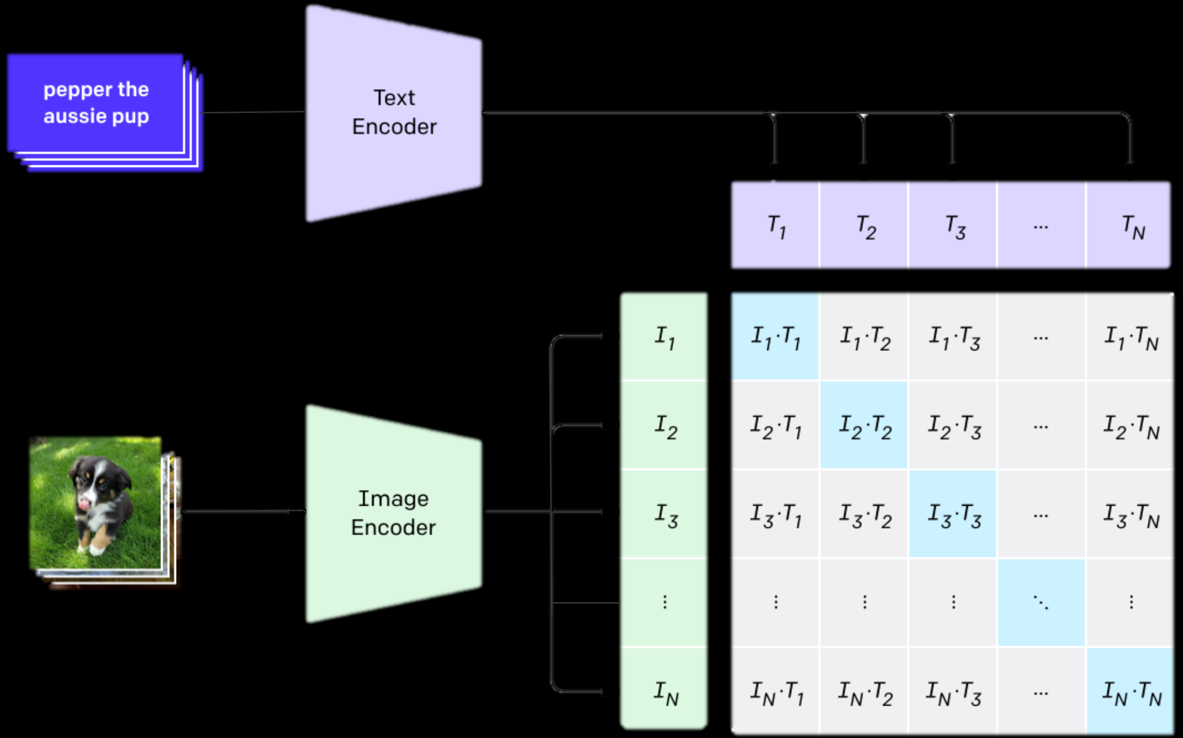
Open-vocabulary

Vision Language Model (VLM)



Clip: (Radford et al., 2021)

Vision Language Model (VLM)



Clip: (Radford et al., 2021)

« Teddy bears mixing sparkling chemicals as mad scientists in a steampunk style »

DALL-E 2 (Openai, 2022)

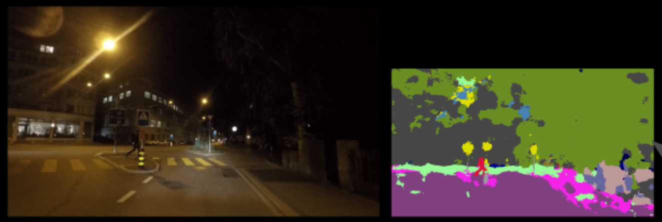
PØDA

Prompt-driven Zero-shot Domain Adaptation

ICCV 23

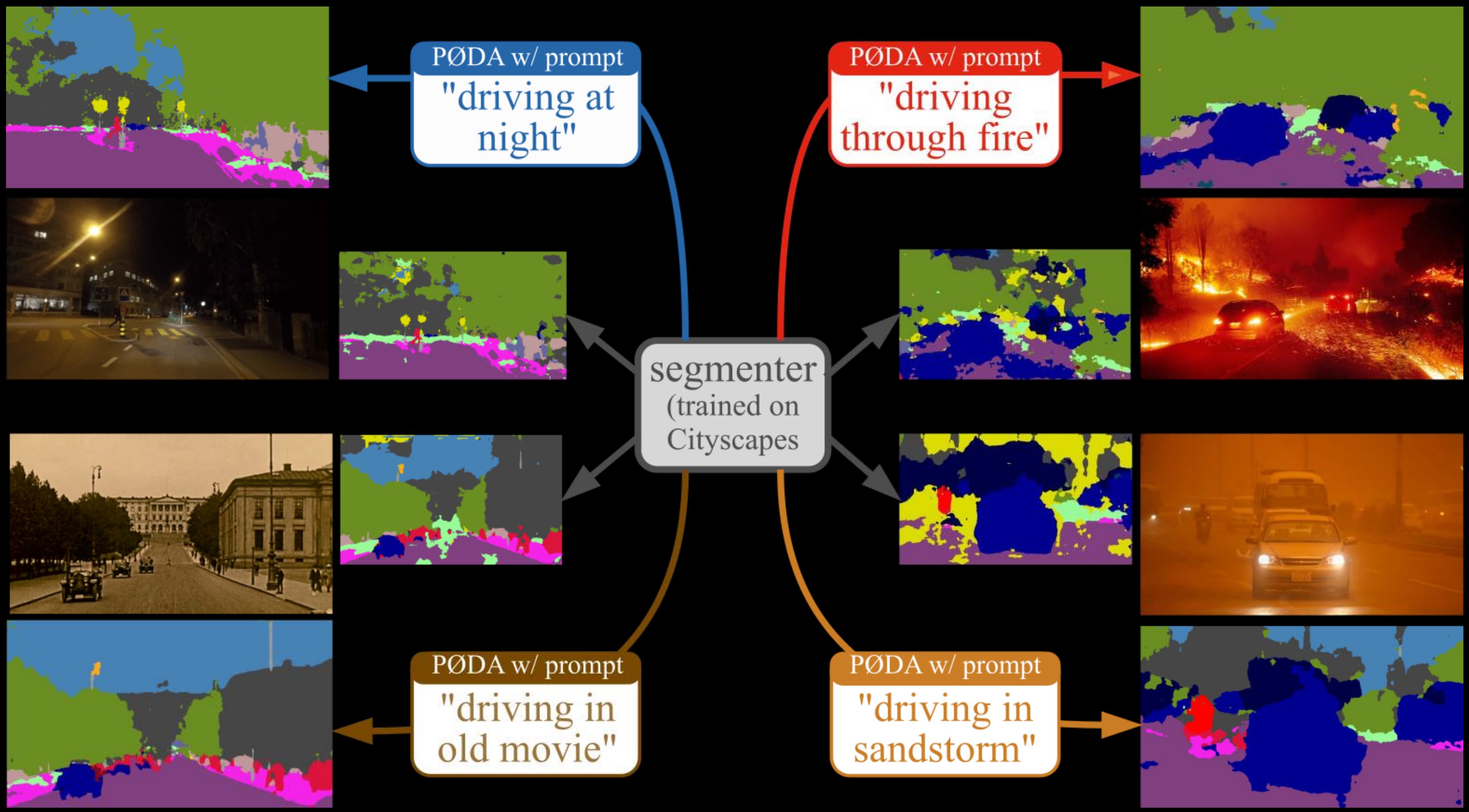
github.com/astra-vision/PODA

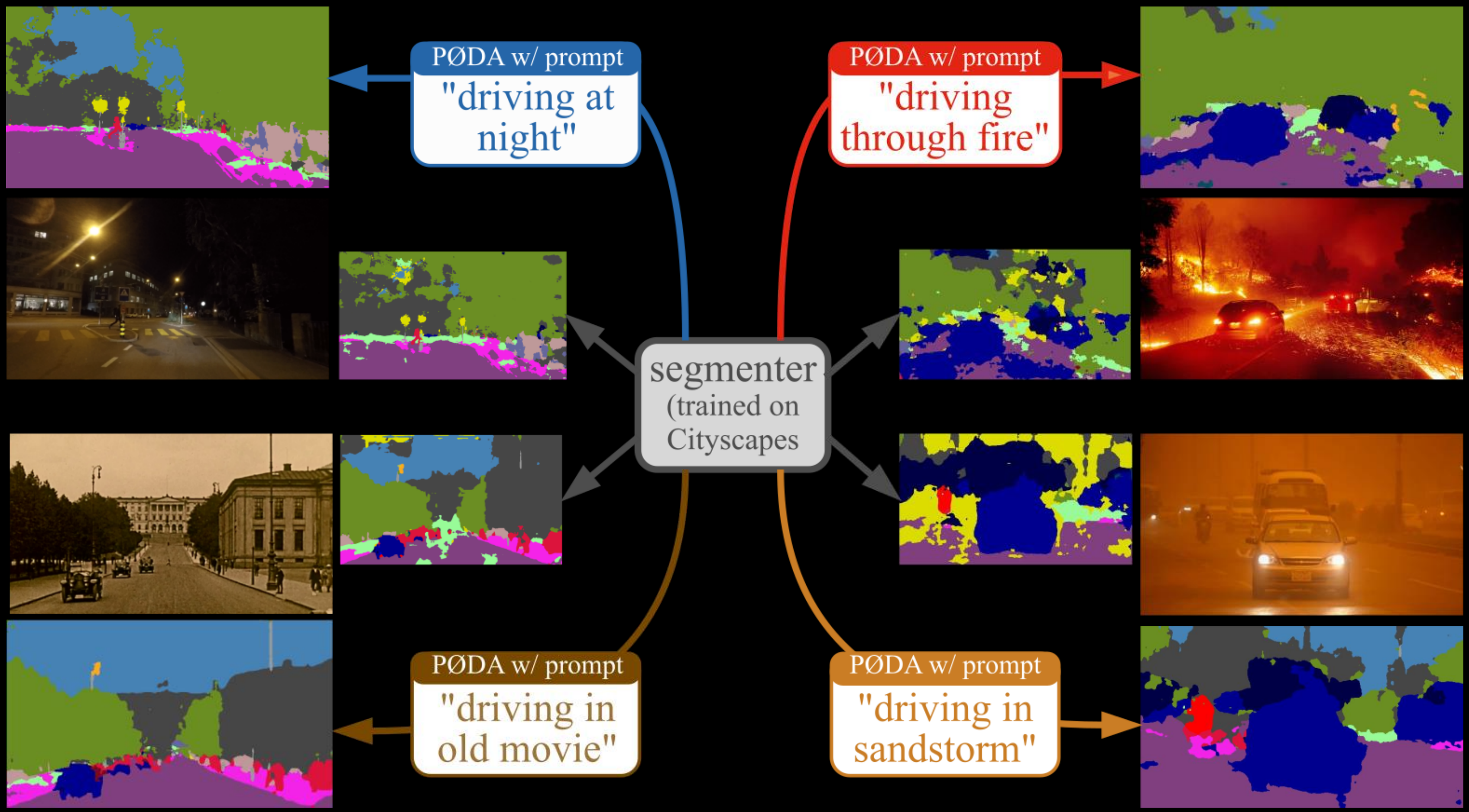
Fahes, Vu, Bursuc, Pérez, de Charette. ICCV 2023



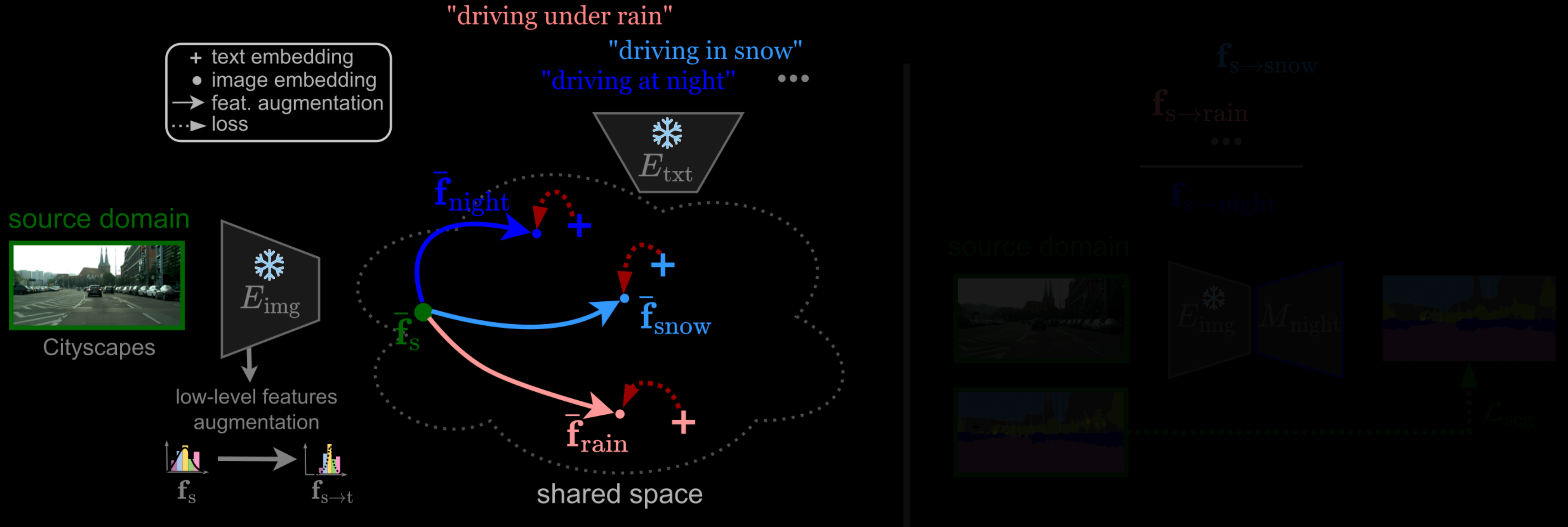
segmenter
(trained on
Cityscapes


PØDA w/ prompt
"driving at night"
segmenter
(trained on
Cityscapes


PØDA w/ prompt
"driving at night"
PØDA w/ prompt
"driving through fire"
segmenter
(trained on
Cityscapes
PØDA w/ prompt
"driving in old movie"
PØDA w/ prompt
"driving in sandstorm"



Minimal features stylization



Prompt-driven feature augmentation

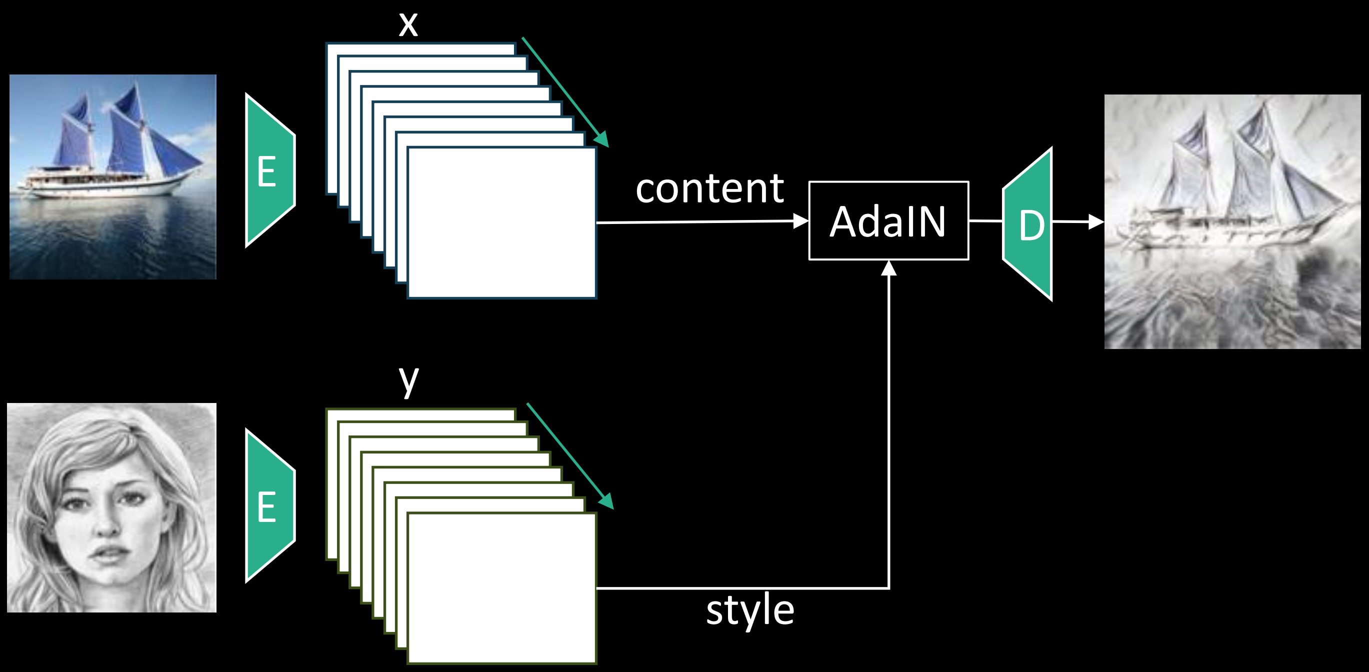
Zero-shot Domain Adaptation

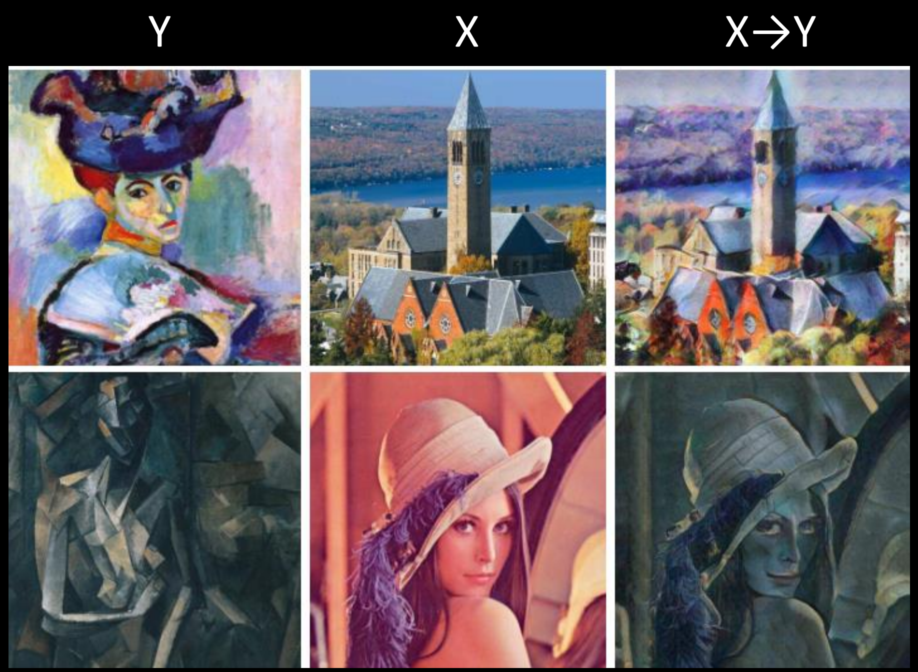
Adaptive Instance Normalization (AdaIN)

High-level feature statistics captures style

AdaIN simply transfers features statistics from **y** to **x** by normalizing and rescaling

$$\mathrm{AdaIN}(x, y) = \sigma(y)\frac{x-\mu(x)}{\sigma(x)} + \mu(y)$$ No learnable parameters





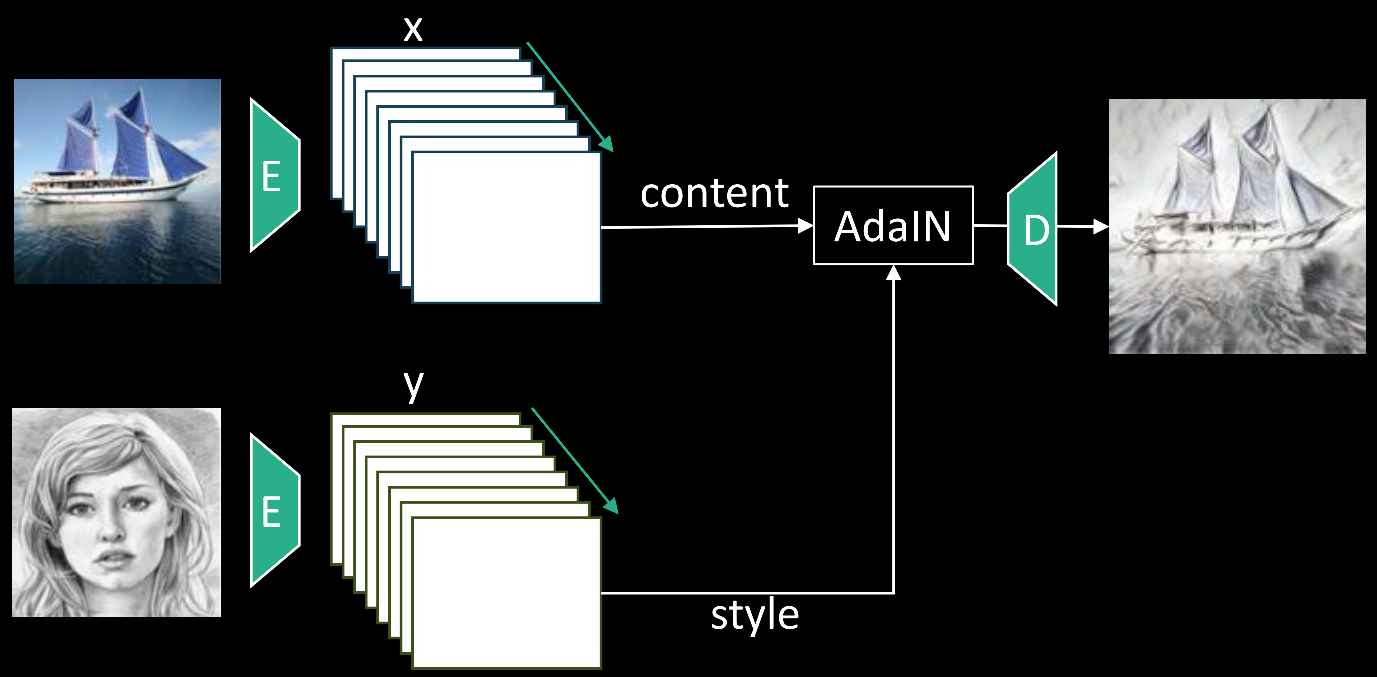
Adaptive Instance Normalization (AdaIN)

High-level feature statistics captures style

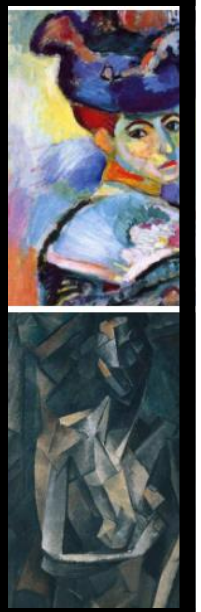
AdaIN simply transfers features statistics from **y** to **x** by normalizing and rescaling

$$\mathrm{AdaIN}(x,y)=\sigma(y)\frac{x-\mu(x)}{\sigma(x)}+\mu(y)$$

No learnable parameters



Y



5. Adaptive Instance Normalization

If IN normalizes the input to a single style specified by the affine parameters, is it possible to adapt it to arbitrarily given styles by using adaptive affine transformations? Here, we propose a simple extension to IN, which we call adaptive instance normalization (AdaIN). AdaIN receives a content input x and a style input y, and simply aligns the channel-wise mean and variance of x to match those of y. Unlike BN, IN or CIN, AdaIN has no learnable affine parameters. Instead, it adaptively computes the affine parameters from the style input:

$$\mathrm{AdaIN}(x,y)=\sigma(y)\left(\frac{x-\mu(x)}{\sigma(x)}\right)+\mu(y) \quad (8)$$

in which we simply scale the normalized content input with $\sigma(y)$, and shift it with $\mu(y)$. Similar to IN, these statistics are computed across spatial locations.

Intuitively, let us consider a feature channel that detects brushstrokes of a certain style. A style image with this kind of strokes will produce a high average activation for this feature. The output produced by AdaIN will have the same high average activation for this feature, while preserving the spatial structure of the content image. The brushstroke feature can be inverted to the image space with a feed-forward decoder, similar to [10]. The variance of this feature channel can encoder more subtle style information, which is also transferred to the AdaIN output and the final output image.

In short, AdaIN performs style transfer in the feature space by transferring feature statistics, specifically the channel-wise mean and variance. Our AdaIN layer plays a similar role as the style swap layer proposed in [6]. While the style swap operation is very time-consuming and memory-consuming, our AdaIN layer is as simple as an IN layer, adding almost no computational cost.

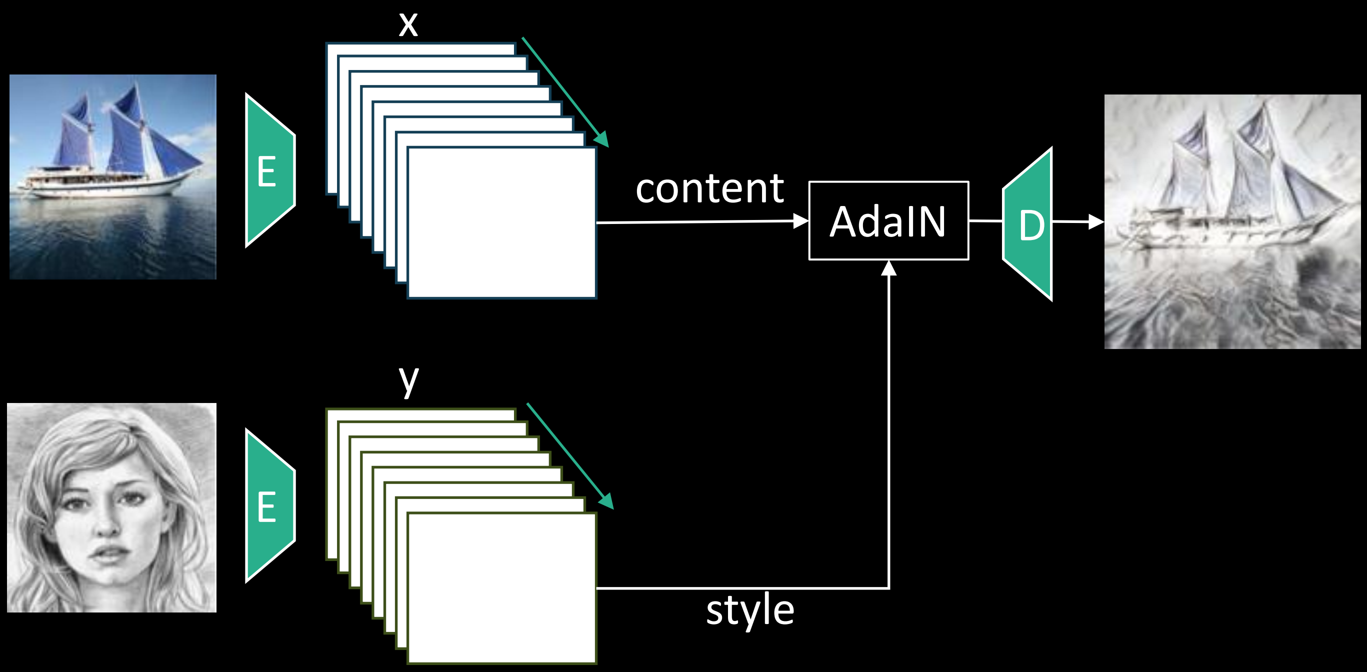
Adaptive Instance Normalization (AdaIN)

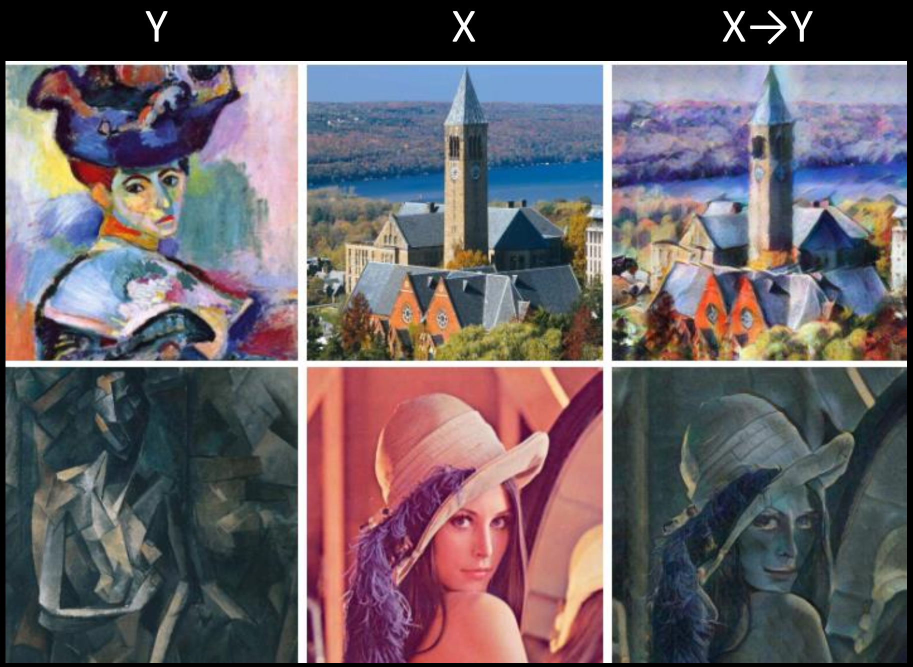
High-level feature statistics captures style

AdaIN simply transfers features statistics from **y** to **x** by normalizing and rescaling

$$\mathrm{AdaIN}(x,y) = \sigma(y)\frac{x-\mu(x)}{\sigma(x)} + \mu(y)$$

No learnable parameters





Huang. and Belongie, AdaIN. ICCV 2017

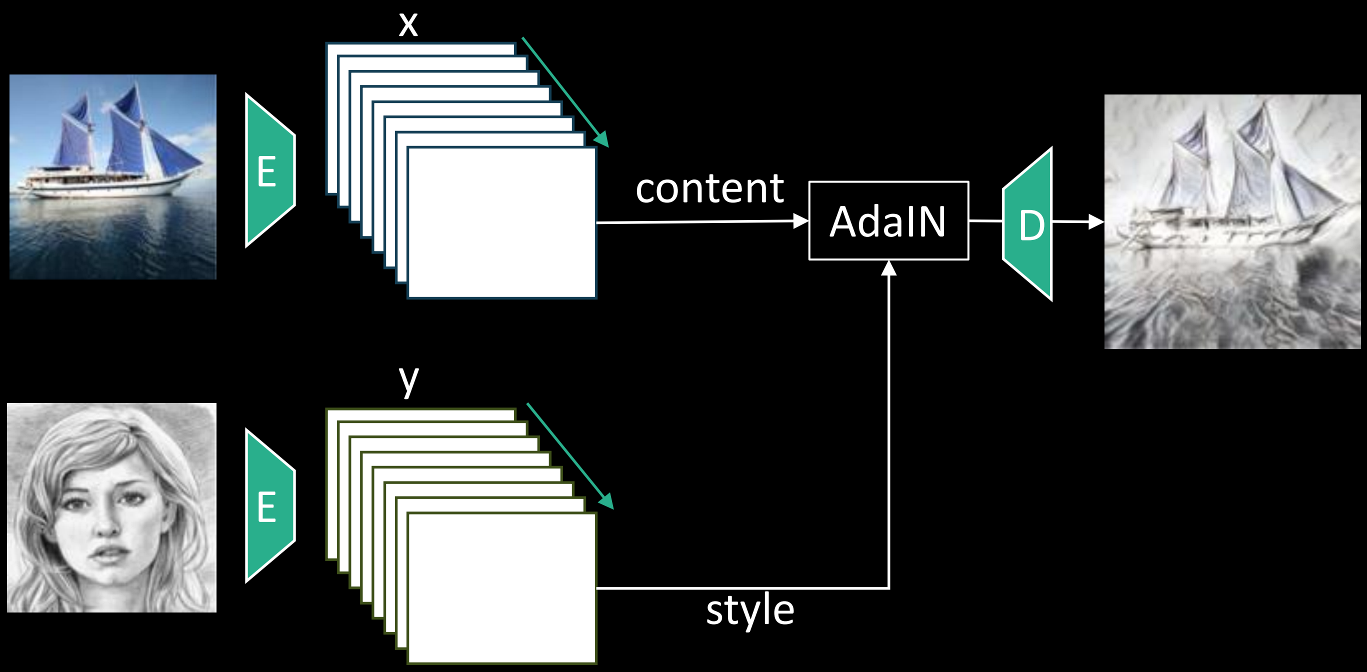
Adaptive Instance Normalization (AdaIN)

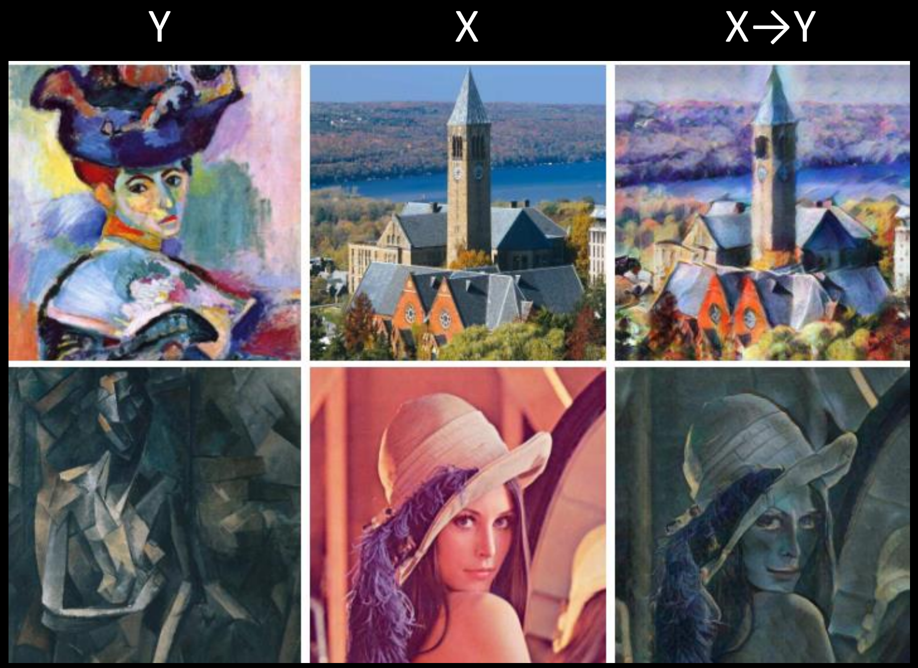
High-level feature statistics captures style

AdaIN simply transfers features statistics from **y** to **x** by normalizing and rescaling

$$\mathrm{AdaIN}(x, y) = \sigma(y)\frac{x-\mu(x)}{\sigma(x)} + \mu(y)$$

No learnable parameters

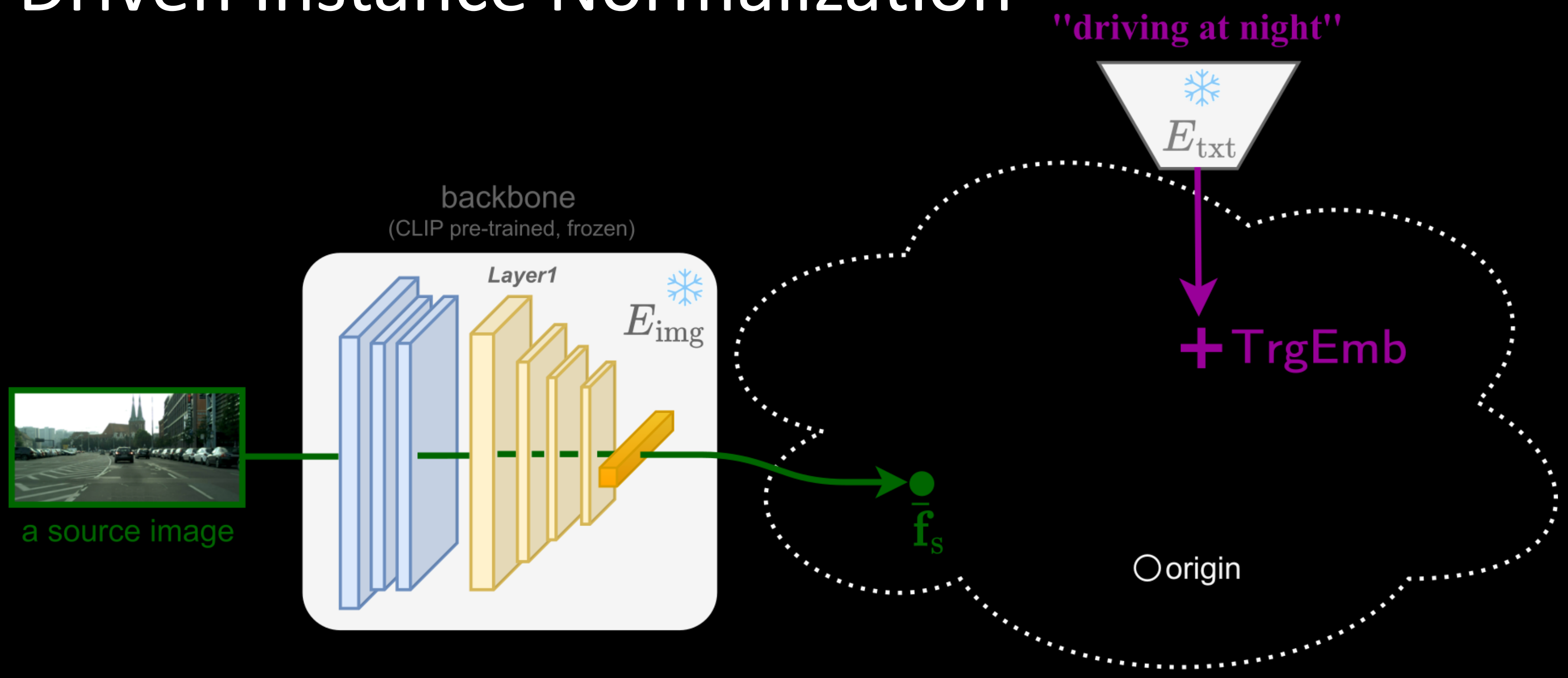




But, we don't have access to target data !

Huang. and Belongie, AdaIN. ICCV 2017

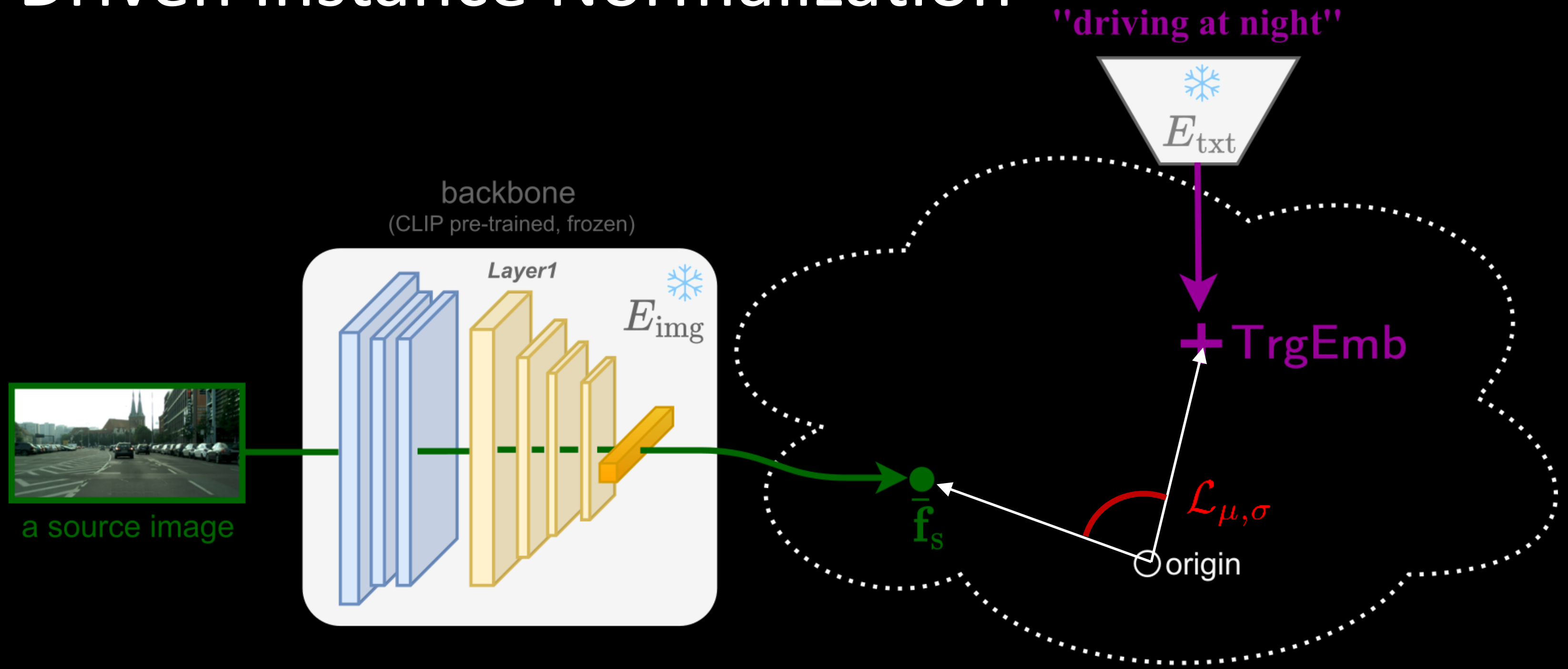
Prompt-Driven Instance Normalization (PIN)



$$\mathtt{PIN}(\mathbf{f}_s, \boldsymbol{\mu}, \boldsymbol{\sigma}) = \boldsymbol{\sigma}\left(\frac{\mathbf{f}_s - \mu(\mathbf{f}_s)}{\sigma(\mathbf{f}_s)}\right)$$

Gradient Descent

Prompt-Driven Instance Normalization (PIN)

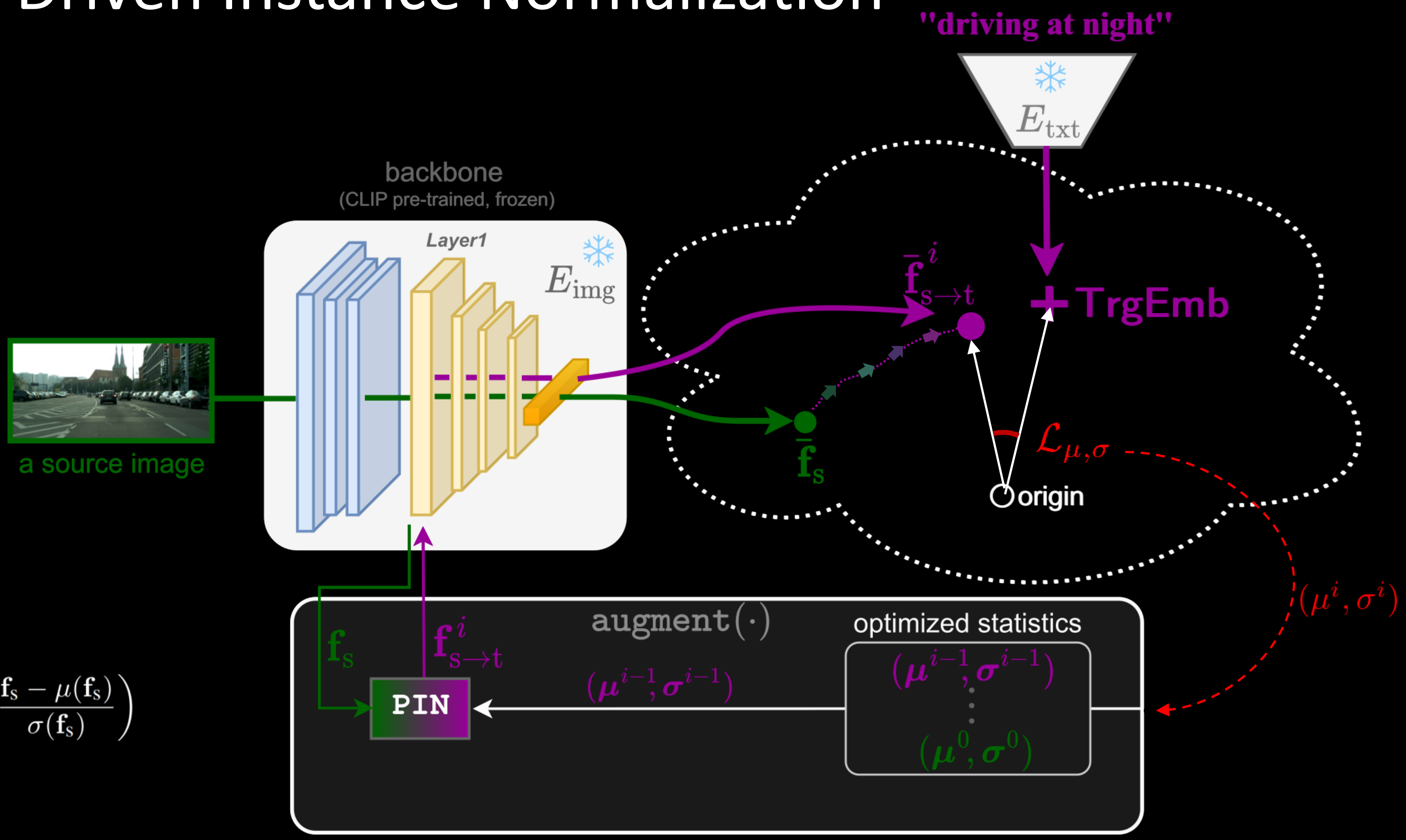


$$\texttt{PIN}(\mathbf{f}_{\text{s}}, \boldsymbol{\mu}, \boldsymbol{\sigma}) = \boldsymbol{\sigma}\left(\frac{\mathbf{f}_{\text{s}} - \mu(\mathbf{f}_{\text{s}})}{\sigma(\mathbf{f}_{\text{s}})}\right)$$

Gradient Descent

$$\mathcal{L}_{\mu,\sigma}(\bar{\mathbf{f}}_{\text{s}\to\text{t}}, \mathbf{TrgEmb}) = 1 - \frac{\bar{\mathbf{f}}_{\text{s}\to\text{t}} \cdot \mathbf{TrgEmb}}{\|\bar{\mathbf{f}}_{\text{s}\to\text{t}}\| \, \|\mathbf{TrgEmb}\|}$$

Prompt-Driven Instance Normalization (PIN)



$$\mathtt{PIN}(\mathrm{f_s}, \boldsymbol{\mu}, \boldsymbol{\sigma}) = \boldsymbol{\sigma}\left(\frac{\mathrm{f_s} - \mu(\mathrm{f_s})}{\sigma(\mathrm{f_s})}\right)$$

Gradient Descent

$$\mathcal{L}_{\mu,\sigma}(\bar{\mathrm{f}}_{\mathrm{s}\to\mathrm{t}}, \mathbf{TrgEmb}) = 1 - \frac{\bar{\mathrm{f}}_{\mathrm{s}\to\mathrm{t}} \cdot \mathbf{TrgEmb}}{\|\bar{\mathrm{f}}_{\mathrm{s}\to\mathrm{t}}\| \, \|\mathbf{TrgEmb}\|}$$

$\boldsymbol{\mu}^0$	$\boldsymbol{\sigma}^0$	mIoU
$\mu(\mathbf{f}_s)$	$\sigma(\mathbf{f}_s)$	**25.03** ± 0.48
0	**1**	8.59 ± 0.82
$\sim \mathcal{N}(\mathbf{0}, \mathbf{I})$	$\sim \mathcal{N}(\mathbf{0}, \mathbf{I})$	6.80 ± 0.92

Initialization

$\boldsymbol{\mu}^0$	$\boldsymbol{\sigma}^0$	mIoU
$\mu(\mathrm{f_s})$	$\sigma(\mathrm{f_s})$	**25.03** $\pm$0.48
0	**1**	8.59 $\pm$0.82
$\sim\mathcal{N}(\mathbf{0},\mathbf{I})$	$\sim\mathcal{N}(\mathbf{0},\mathbf{I})$	6.80 $\pm$0.92

Initialization

Layer1	*Layer2*	*Layer3*	*Layer4*	ACDC Night
✓	✗	✗	✗	**25.03** $\pm$0.48
✓	✓	✗	✗	23.43 $\pm$0.51
✓	✗	✓	✗	22.93 $\pm$0.53
✓	✗	✗	✓	21.05 $\pm$0.55

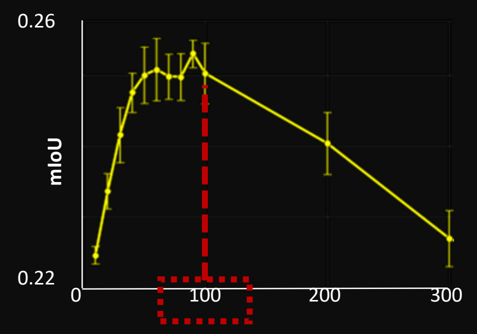
Minimal augmentation

$\boldsymbol{\mu}^0$	$\boldsymbol{\sigma}^0$	mIoU
$\mu(\mathbf{f}_s)$	$\sigma(\mathbf{f}_s)$	**25.03** ±0.48
$\mathbf{0}$	$\mathbf{1}$	8.59 ±0.82
$\sim \mathcal{N}(\mathbf{0}, \mathbf{I})$	$\sim \mathcal{N}(\mathbf{0}, \mathbf{I})$	6.80 ±0.92

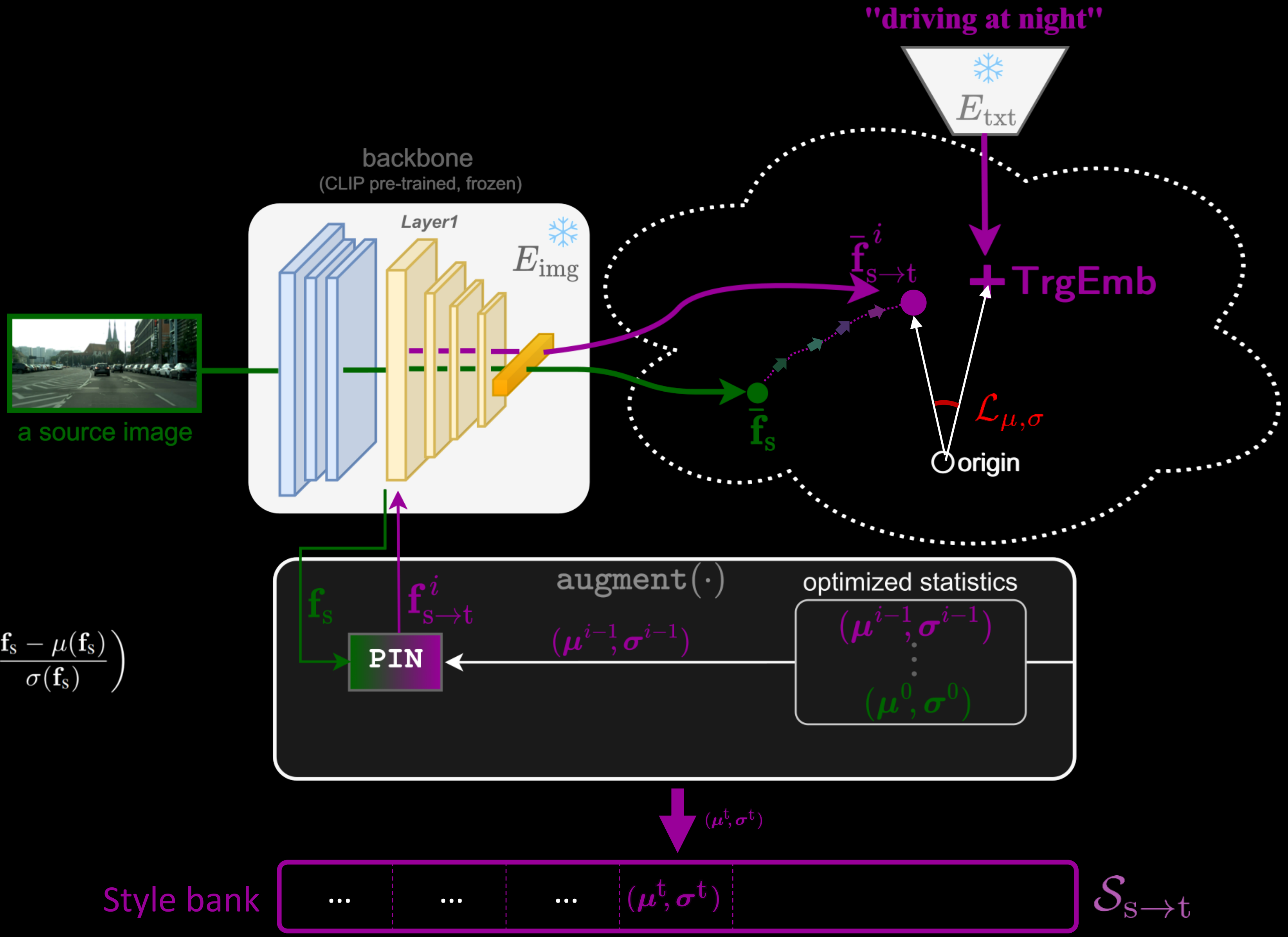
Initialization

Layer1	*Layer2*	*Layer3*	*Layer4*	ACDC Night
✓	✗	✗	✗	**25.03** ±0.48
✓	✓	✗	✗	23.43 ±0.51
✓	✗	✓	✗	22.93 ±0.53
✓	✗	✗	✓	21.05 ±0.55

Minimal augmentation

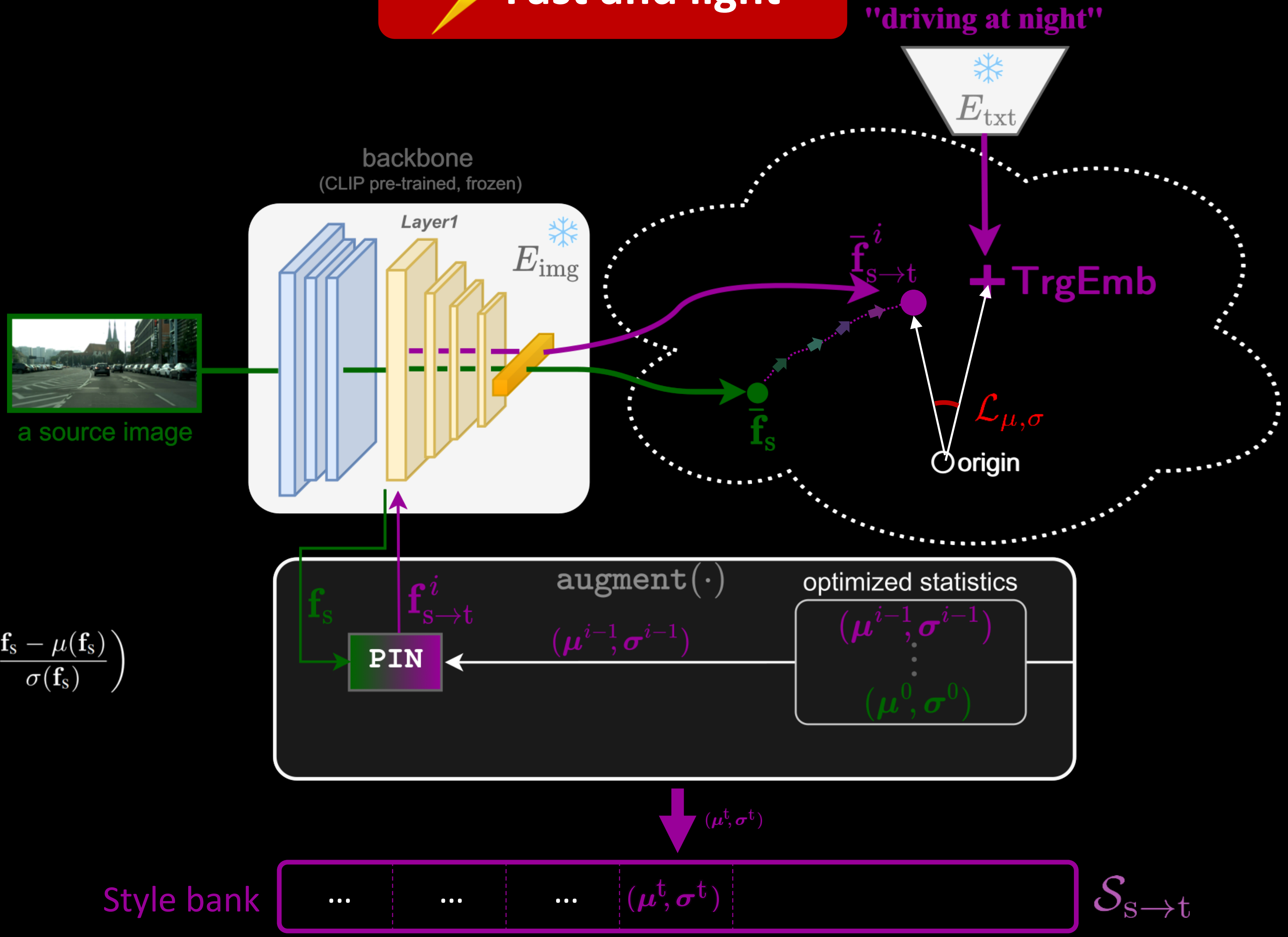


# of iterations

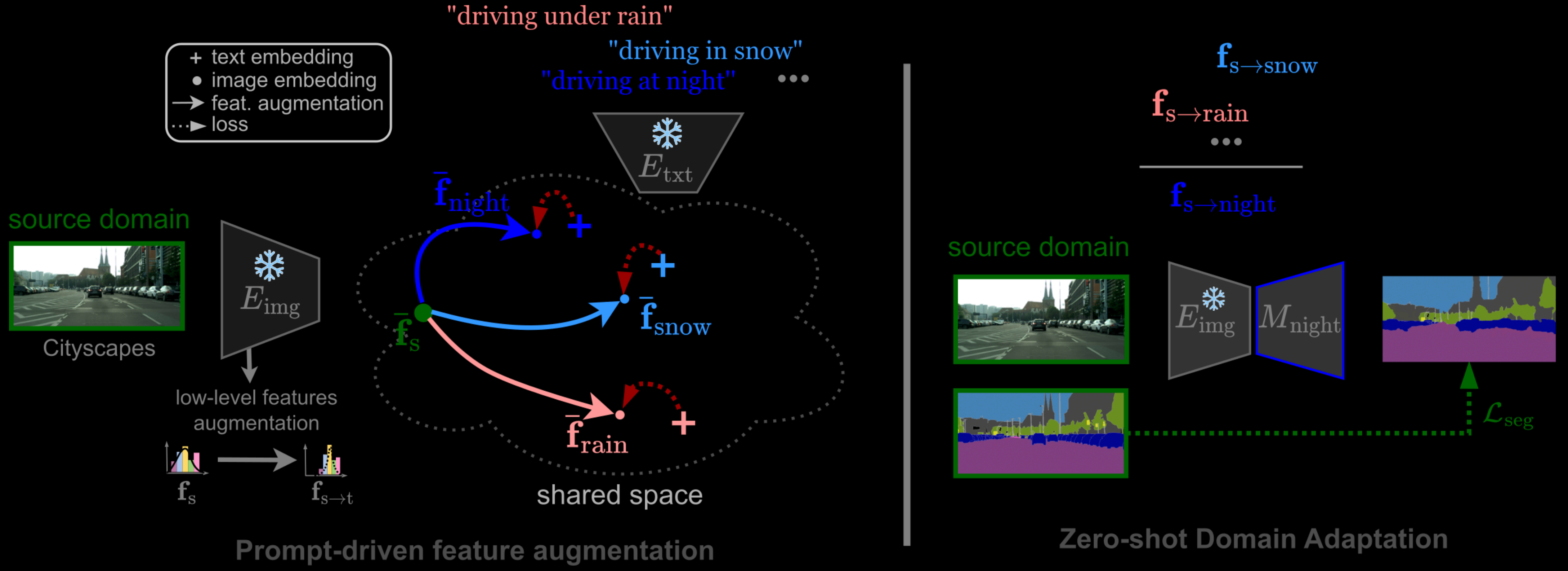


$$\mathtt{PIN}(\mathrm{f_s}, \boldsymbol{\mu}, \boldsymbol{\sigma}) = \boldsymbol{\sigma}\left(\frac{\mathrm{f_s} - \mu(\mathrm{f_s})}{\sigma(\mathrm{f_s})}\right)$$

Fast and light



$$\mathtt{PIN}(f_s, \boldsymbol{\mu}, \boldsymbol{\sigma}) = \boldsymbol{\sigma}\left(\frac{f_s - \mu(f_s)}{\sigma(f_s)}\right)$$



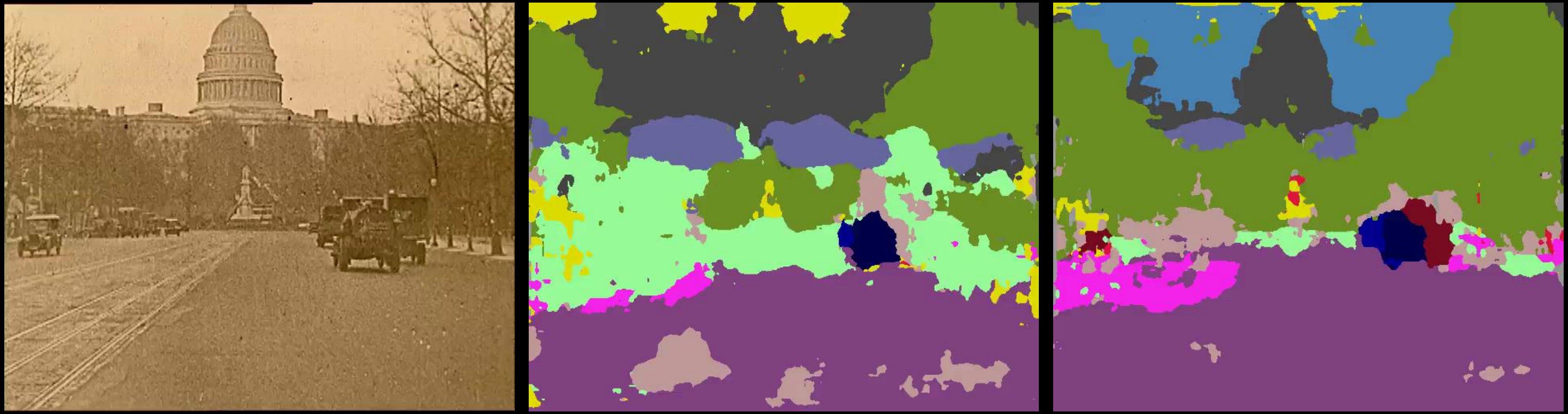
+ text embedding
• image embedding
feat. augmentation
loss
"driving under rain"
"driving in snow"
"driving at night"
•••
E_{txt}
source domain
Cityscapes
E_{img}
low-level features augmentation
$\mathbf{f}_s$
$\mathbf{f}_{s\to t}$
$\bar{\mathbf{f}}_{night}$
$\bar{\mathbf{f}}_{snow}$
$\bar{\mathbf{f}}_{rain}$
$\bar{\mathbf{f}}_s$
shared space
Prompt-driven feature augmentation
$\mathbf{f}_{s\to snow}$
$\mathbf{f}_{s\to rain}$
•••
$\mathbf{f}_{s\to night}$
source domain
E_{img}
M_{night}
$\mathcal{L}_{seg}$
Zero-shot Domain Adaptation

Algorithm 2: Prompt-driven Zero-shot DA

Input: Source dataset $\mathcal{D}_\text{s} = \{(\mathbf{x}_\text{s}, \mathbf{y}_\text{s})\}$
CLIP encoders E_img and E_txt
Target domain description $\mathsf{TrgPrompt}$
Feature backbone $M_\text{feat} \leftarrow E_\text{img}$
Source model: $M = (M_\text{feat}, M_\text{cls})$

Result: Target-adapted model $M' = (M_\text{feat}, M'_\text{cls})$

// Initialization
$\mathsf{TrgEmb} = E_\text{txt}(\mathsf{TrgPrompt})$
$M_\text{cls} \leftarrow \texttt{train}(M_\text{cls}, \mathcal{D}_\text{s})$ ▷ source-only training
// Feature Augmentation
$\mathcal{F}_\text{s} \leftarrow \texttt{feat-ext}(M_\text{feat}, \{\mathbf{x}_\text{s}\})$
$\mathcal{S}_{\text{s}\rightarrow\text{t}} \leftarrow \texttt{augment}(\mathcal{F}_\text{s}, \mathsf{TrgEmb})$
// Adaptation
$M'_\text{cls} \leftarrow \texttt{fine-tune}(M_\text{cls}, \mathcal{F}_\text{s}, \mathcal{S}_{\text{s}\rightarrow\text{t}}, \{\mathbf{y}_\text{s}\})$ ▷ fine-tuning

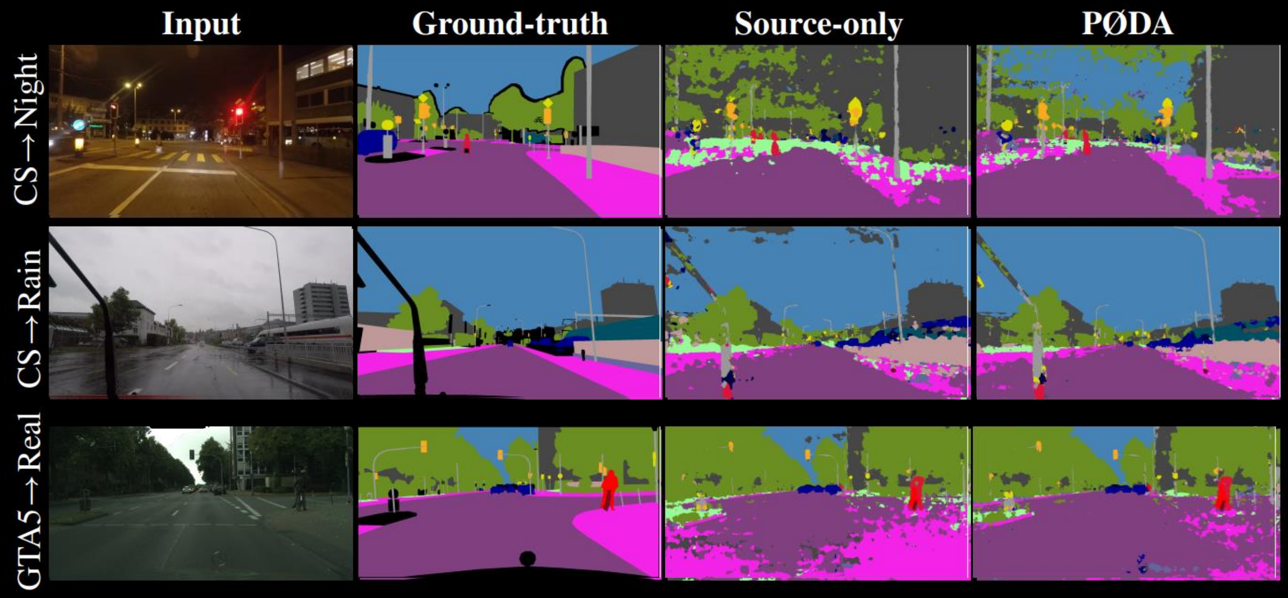


``driving in old movie"

Input | Source-only | PØDA (ours)

"driving at night"

Evaluation on ACDC (Sakaridis et al., ICCV’21) and GTA5 (Richter et al., ECCV’16)

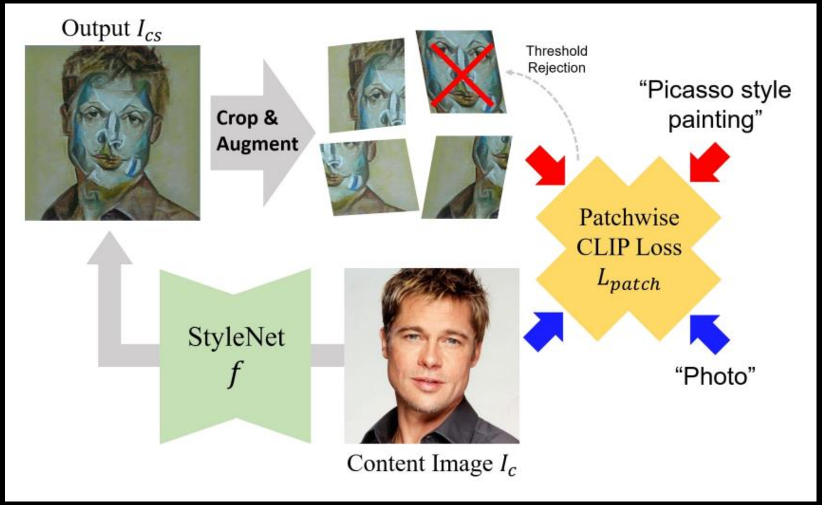


Proxies

Source	Target eval.	Method	mIoU[%]
	TrgPrompt = “driving at night”		
CS	ACDC Night	source-only	18.31
		CLIPstyler [1]	21.38 ±0.36
		PØDA	**25.03** ±0.48
	TrgPrompt = “driving in snow”		
	ACDC Snow	source-only	39.28
		CLIPstyler [1]	41.09 ±0.17
		PØDA	**43.90** ±0.53
	TrgPrompt = “driving under rain”		
	ACDC Rain	source-only	38.20
		CLIPstyler [1]	37.17 ±0.10
		PØDA	**42.31** ±0.55
	TrgPrompt = “driving in a game”		
	GTA5	source-only	39.59
		CLIPstyler [1]	38.73 ±0.16
		PØDA	**41.07** ±0.48
	TrgPrompt = “driving”		
GTA5	CS	source-only	36.38
		CLIPstyler [1]	31.50 ±0.21
		PØDA	**40.08** ±0.52

[1] **Clipstyler**, CVPR 2022

Comparison to CLIPStyler

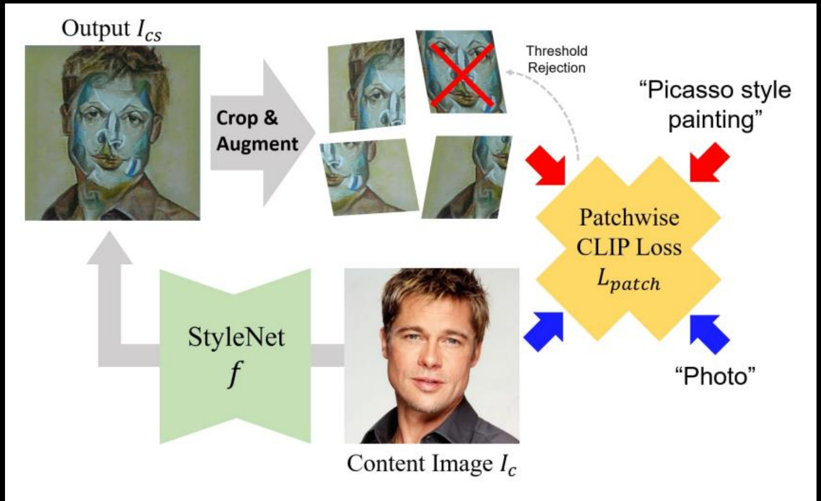


CLIPStyler optimization: 65sec
PODA optimization: **0.3sec**

Source (CS) Rain Night Snow Game

[1] **Clipstyler**, CVPR 2022

Comparison to CLIPStyler

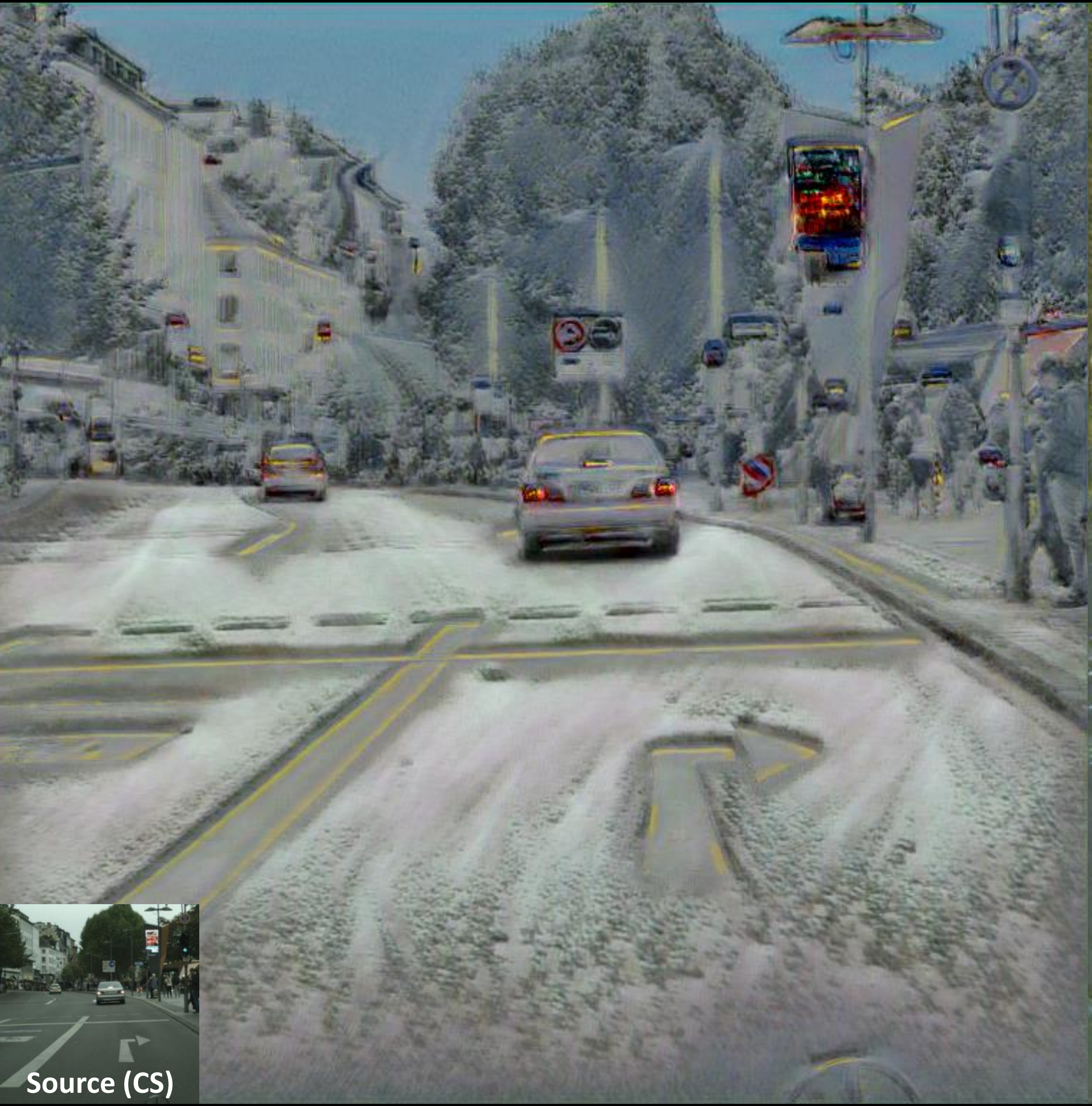


Source (CS) | Rain | Night | Snow | Game

CLIPStyler optimization: 65sec
PODA optimization: **0.3sec**

[1] **Clipstyler**, CVPR 2022

CLIPStyler



“Driving in snow”

“Driving in a game”

Prompt design

give me 5 prompts that have the same exact meaning as "{prompt}"

Chat GPT

give me 5 random prompts of length from 3 to 6 words describing a random photo

Method		ACDC Night	ACDC Snow	ACDC Rain	GTA5
Source only		18.31	39.28	38.20	39.59
Trg		"driving at night"	"driving in snow"	"driving under rain"	"driving in a game"
		25.03 ±0.48	43.90 ±0.53	42.31 ±0.55	41.07 ±0.48
ChatGPT-generated	Relevant	"operating a vehicle after sunset"	"operating a vehicle in snowy conditions"	"operating a vehicle in wet conditions"	"piloting a vehicle in a virtual world"
		24.38 ±0.37	44.33 ±0.36	42.21 ±0.47	41.25 ±0.40
		"driving during the nighttime hours"	"driving on snow-covered roads"	"driving on rain-soaked roads"	"controlling a car in a digital simulation"
		25.22 ±0.64	43.56 ±0.62	**42.51** ±0.33	41.19 ±0.14
		"navigating the roads in darkness"	"piloting a vehicle in snowy terrain"	"navigating through rainfall while driving"	"maneuvering a vehicle in a computerized racing experience""
		24.73 ±0.47	**44.67** ±0.18	41.11 ±0.69	40.34 ±0.49
		"driving in low-light conditions"	"driving in wintry precipitation"	"driving in inclement weather"	"operating a transport in a video game environment"
		24.68 ±0.34	43.11 ±0.56	40.68 ±0.37	41.34 ±0.42
		"travelling by car after dusk"	"travelling by car in a snowstorm"	"travelling by car during a downpour"	"navigating a machine through a digital driving simulation"
		24.89 ±0.24	43.83 ±0.17	42.05 ±0.35	**41.86** ±0.10
		24.82	*43.90*	*41.81*	*41.18*
	Irrelevant	"mesmerizing northern lights display"			
		20.05 ±0.77	40.07 ±0.66	38.43 ±0.82	37.98 ±0.31
		"playful dolphins in the ocean"			
		20.11 ±0.31	39.87 ±0.26	38.56 ±0.58	37.05 ±0.31
		"breathtaking view from mountaintop"			
		20.65 ±0.33	42.08 ±0.28	40.05 ±0.52	40.09 ±0.23
		"cheerful sunflower field in bloom"			
		21.10 ±0.50	39.85 ±0.68	40.09 ±0.41	37.93 ±0.55
		"dramatic cliff overlooking the ocean"			
		20.09 ±0.98	38.20 ±0.54	38.48 ±0.37	37.57 ±0.46
		"majestic eagle in flight over mountains"			
		20.70 ±0.38	39.60 ±0.27	40.38 ±0.86	38.52 ±0.21
		20.45	*39.95*	*39.33*	*38.19*

Prompt design

give me 5 prompts that have the same exact meaning as “{prompt}”

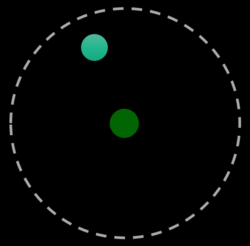
Chat GPT

give me 5 random prompts of length from 3 to 6 words describing a random photo

Method	ACDC Night	ACDC Snow	ACDC Rain	GTA5
Source only	18.31	39.28	38.20	39.59
Trg	“driving at night”	“driving in snow”	“driving under rain”	“driving in a game”
	25.03 ±0.48	43.90 ±0.53	42.31 ±0.55	41.07 ±0.48
ChatGPT-generated — Relevant	“operating a vehicle after sunset”	“operating a vehicle in snowy conditions”	“operating a vehicle in wet conditions”	“piloting a vehicle in a virtual world”
	24.38 ±0.37	44.33 ±0.36	42.21 ±0.47	41.25 ±0.40
	“driving during the nighttime hours”	“driving on snow-covered roads”	“driving on rain-soaked roads”	“controlling a car in a digital simulation”
	25.22 ±0.64	43.56 ±0.62	**42.51** ±0.33	41.19 ±0.14
	“navigating the roads in darkness”	“piloting a vehicle in snowy terrain”	“navigating through rainfall while driving”	“maneuvering a vehicle in a computerized racing experience””
	24.73 ±0.47	**44.67** ±0.18	41.11 ±0.69	40.34 ±0.49
	“driving in low-light conditions”	“driving in wintry precipitation”	“driving in inclement weather”	“operating a transport in a video game environment”
	24.68 ±0.34	43.11 ±0.56	40.68 ±0.37	41.34 ±0.42
	“travelling by car after dusk”	“travelling by car in a snowstorm”	“travelling by car during a downpour”	“navigating a machine through a digital driving simulation”
	24.89 ±0.24	43.83 ±0.17	42.05 ±0.35	**41.86** ±0.10
	24.82	*43.90*	*41.81*	*41.18*
ChatGPT-generated — Irrelevant	“mesmerizing northern lights display”			
	20.05 ±0.77	40.07 ±0.66	38.43 ±0.82	37.98 ±0.31
	“playful dolphins in the ocean”			
	20.11 ±0.31	39.87 ±0.26	38.56 ±0.58	37.05 ±0.31
	“breathtaking view from mountaintop”			
	20.65 ±0.33	42.08 ±0.28	40.05 ±0.52	40.09 ±0.23
	“cheerful sunflower field in bloom”			
	21.10 ±0.50	39.85 ±0.68	40.09 ±0.41	37.93 ±0.55
	“dramatic cliff overlooking the ocean”			
	20.09 ±0.98	38.20 ±0.54	38.48 ±0.37	37.57 ±0.46
	“majestic eagle in flight over mountains”			
	20.70 ±0.38	39.60 ±0.27	40.38 ±0.86	38.52 ±0.21
	20.45	*39.95*	*39.33*	*38.19*

Always better

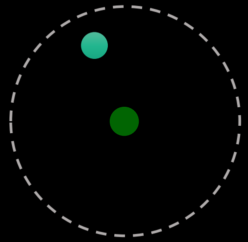
Always worse



$$y = \sigma_s^\star \frac{x - \mu_c}{\sigma_c} + \mu_s^\star,$$

$$\sigma_s^\star = \boxed{\alpha\sigma_c,} \qquad \mu_s^\star = \boxed{\beta\mu_c}$$

Towards robust object detection. (Fan et al., ICLR 23)



$$y = \sigma_s^\star \frac{x - \mu_c}{\sigma_c} + \mu_s^\star,$$

$$\sigma_s^\star = \boxed{\alpha\sigma_c,} \qquad \mu_s^\star = \boxed{\beta\mu_c}$$

Towards robust object detection. (Fan et al., ICLR 23)

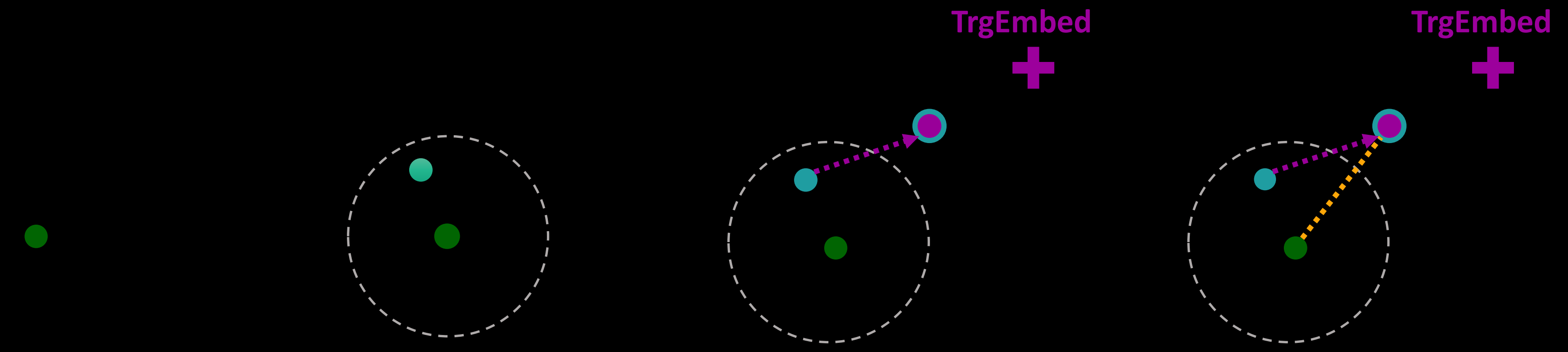
Method	Target	S → C	C → F
Our Baseline	✗	32.8	22.0
BIN	✗	44.3	28.4
IBN	✗	47.4	31.2
SFA	✗	38.4	25.3
pAdaIN	✗	43.7	27.6
Mixstyle	✗	46.4	30.1
DSU	✗	49.3	34.1
NP (Ours)	✗	54.1	44.0
NP+ (Ours)	✗	58.7	46.3

+9% +12%

	BDD Day → Night			BDD Night → Day			WaymoL → BDD			WaymoR → BDD		
	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75
Faster R-CNN	17.84	31.35	17.68	19.14	33.04	19.16	10.07	19.62	9.05	8.65	17.26	7.49
+ CycConsist	18.35	32.44	18.07	18.89	33.50	18.31	11.55	23.44	10.00	9.11	17.92	7.98
+ CycConf	19.09	33.58	19.14	19.57	34.34	**19.26**	12.27	26.01	10.24	9.99	20.58	8.30
+ NP (Ours)	20.73	36.22	20.85	19.32	34.42	18.63	17.85	35.34	15.52	14.97	29.42	13.11
+ NP+ (Ours)	**20.97**	**36.76**	**21.10**	**19.73**	**35.30**	19.19	**21.18**	**42.16**	**18.67**	**19.64**	**38.69**	**17.07**

+2% +0.5% +9% +10%

Method	Night	Snow	Rain	GTA5
Source-only	18.31	39.28	38.20	39.59
Source-only-G	21.07	42.84	42.38	41.54
PØDA-G	**24.86** ±0.70	44.34 ±0.36	43.17 ±0.63	41.73 ±0.39
PØDA-G+style-mix	24.18 ±0.23	**44.46** ±0.34	**43.56** ±0.46	**42.98** ±0.12



Source-only

Source-only-G

$(\mu, \sigma) \in \mathrm{U}(0,1)$

$\mathbf{f}_{s\to t} = \mathtt{AdaIN}(\mathbf{f}_s, \boldsymbol{\mu}, \boldsymbol{\sigma})$

Inspiration [1]

PODA-G

(μ_t, σ_t) PODA on (μ, σ)

$\mathbf{f}_{s\to t} = \mathtt{PIN}(\mathbf{f}_s, \boldsymbol{\mu}_t, \boldsymbol{\sigma}_t)$

PODA-G + style mixing

(μ_t, σ_t) PODA on (μ, σ)

$\mathbf{f}_{s\to t} = \mathtt{PIN}(\mathbf{f}_s, \boldsymbol{\mu}_{\mathrm{mix}}, \boldsymbol{\sigma}_{\mathrm{mix}})$

Inspiration [2]

[1] Fan et al. Towards robust object detection invariant to real-world domain shifts. ICLR 23
[2] Wu et al. Style mixing and patchwise prototypical matching for oneshot unsupervised domain adaptive semantic segmentation. AAAI 22

Various backbones

Backbone	Method	Night	Snow	Rain	GTA5
Sem. FPN	src-only	18.10	35.75	36.07	40.67
	PØDA	**21.48** ±0.15	**39.55** ±0.13	**38.34** ±0.29	**41.59** ±0.24
DLv3+	src-only	22.17	44.53	42.53	40.49
	PØDA	**26.54** ±0.12	**46.71** ±0.43	**46.36** ±0.20	**43.17** ±0.13

Effect of priors

Method	Prior	ACDC Night
CIConv* [26]	physics	30.60 / 34.50 (Δ=3.90)
SM-PPM [56]	1 target image	13.07 / 14.60 (Δ=1.53)
CLIPstyler [25]	1 prompt	18.31 / 21.38 (Δ=3.07)
PØDA	1 prompt	18.31 / 25.03 (Δ=6.72)

* Results of CIConv are on DarkZurich, a subset of ACDC Night [45].

Cityscapes-Foggy. Sakaridis et al., IJCV 2018
Diverse Weather Dataset, Wu et al. ECCV'22
CUB-200. Wah et al. 2011
CUB-200-Paintings. Wang et al. CVPR'20
[8] DA-Faster. Chen et al. IJCV'21
[15] NP+. Fan et al. ICLR'23
[25] ClipStyler. Kwon and Ye, CVPR'22
[26] CIConv. Lengyel et al. ICCV'21
[42] ViSGA. Rezaeianaran et al. ICCV'21
[49] CLIP The Gap. Vidit et al. CVPR'23
[55] S-DGOD. Wu and Deng, ECCV'22
[56] SM-PPM. Wu et al., AAAI'22

Method	Target	CS→ CS Foggy	DWD-Day Clear → Night Clear	Dusk Rainy	Night Rainy	Day Foggy
DA-Faster [8]	✓	32.0	-	-	-	-
ViSGA [42]	✓	43.3	-	-	-	-
NP+ [15]	✗	46.3	-	-	-	-
S-DGOD [55]	✗	-	36.6	28.2	16.6	33.5
CLIP The Gap [49]	✗	-	36.9	32.3	18.7	38.5
PØDA (Faster-RCNN)	✗	**47.3**	**43.4**	**40.2**	**20.5**	**44.4**

Object Detection

"Painting of a bird"
"Blue/Red digits"

Method	CUB-200 paintings	Colored MNIST
src-only	28.90	55.83
PØDA	**30.91** ±0.69	**64.16** ±0.41

Classification

Cityscapes-Foggy. Sakaridis et al., IJCV 2018
Diverse Weather Dataset, Wu et al. ECCV'22
CUB-200. Wah et al. 2011
CUB-200-Paintings. Wang et al. CVPR'20

[8] DA-Faster. Chen et al. IJCV'21
[15] NP+. Fan et al. ICLR'23
[25] ClipStyler. Kwon and Ye, CVPR'22
[26] CIConv. Lengyel et al. ICCV'21
[42] ViSGA. Rezaeianaran et al. ICCV'21
[49] CLIP The Gap. Vidit et al. CVPR'23
[55] S-DGOD. Wu and Deng, ECCV'22
[56] SM-PPM. Wu et al., AAAI'22

Limitations ?

- Global stylization: large structural classes benefit more from it

Limitations ?

- Global stylization: large structural classes benefit more from it

Source	Target eval.	Method	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle	mIoU%
		TrgPrompt = “driving at night”																				
CS	ACDC Night	source-only	70.42	18.32	**43.83**	6.11	17.08	23.52	**24.51**	19.76	39.74	6.11	0.78	21.62	8.96	23.08	2.53	0.00	3.27	8.42	9.87	18.31
		CLIPstyler	73.96	23.26	42.16	3.31	7.21	**35.49**	23.34	19.01	**45.41**	8.81	**27.87**	21.06	8.48	38.17	1.84	0.00	11.54	**10.38**	4.89	21.38 ±0.36
		PØDA	**77.54**	**26.90**	42.71	**13.51**	**21.36**	33.52	23.70	**21.73**	39.91	**9.51**	19.40	**28.80**	**11.85**	**50.89**	**10.14**	0.00	**20.76**	8.76	**14.50**	**25.03** ±0.48
		TrgPrompt = “driving in snow”																				
	ACDC snow	source-only	70.47	23.50	63.80	17.96	**27.36**	**38.52**	**56.26**	**45.00**	**83.00**	10.75	83.65	47.73	0.72	61.42	21.87	5.90	21.58	35.83	31.01	39.28
		CLIPstyler	74.29	31.25	69.17	15.21	25.21	36.83	44.79	42.56	76.87	11.07	91.48	**53.23**	0.13	67.66	23.88	**9.14**	36.48	**42.67**	28.76	41.09 ±0.17
		PØDA	**75.40**	**34.61**	**75.22**	**26.77**	27.34	35.20	52.68	44.37	82.01	**14.16**	**93.72**	50.51	**0.99**	**69.11**	**26.64**	2.72	**46.98**	42.64	**33.09**	**43.90** ±0.53
		TrgPrompt = “driving under rain”																				
	ACDC rain	source-only	74.10	31.98	63.07	**15.08**	**23.92**	**41.31**	**50.12**	**44.43**	79.93	22.07	87.45	47.99	4.39	68.92	10.35	18.52	13.64	7.03	21.58	38.20
		CLIPstyler	73.71	36.09	68.91	3.77	16.99	36.94	39.75	36.44	78.21	20.64	91.79	40.34	**9.65**	**74.54**	13.16	**20.33**	12.73	14.06	18.26	37.17 ±0.10
		PØDA	**76.60**	**38.52**	**78.01**	15.02	22.53	40.33	45.39	41.40	**86.85**	**37.97**	**96.46**	**50.39**	6.35	74.19	**19.19**	7.98	**22.06**	**21.04**	**23.65**	**42.31** ±0.55
		TrgPrompt = “driving in a game”																				
	GTA5	source-only	68.72	22.65	78.79	36.81	**17.31**	39.66	**39.33**	14.84	**72.61**	22.53	87.31	57.50	26.14	74.29	**44.57**	**20.45**	0.00	18.30	10.35	39.59
		CLIPstyler	73.06	**29.89**	77.86	25.50	11.69	**39.72**	35.88	**24.04**	67.38	12.75	88.77	46.58	33.38	72.03	42.79	11.12	0.00	28.84	**14.61**	38.73 ±0.16
		PØDA	**73.93**	22.69	**78.82**	**37.52**	14.17	36.97	33.14	17.34	72.44	**26.22**	**88.85**	**62.69**	**37.04**	**74.33**	43.03	11.91	0.00	**35.33**	13.91	**41.07** ±0.48
		TrgPrompt = “driving”																				
GTA5	CS	source-only	58.97	20.92	72.84	16.53	**24.58**	31.37	34.77	23.62	82.12	17.04	66.28	**63.46**	14.72	**81.27**	**20.83**	17.19	4.68	**20.57**	19.56	36.38
		CLIPstyler	66.70	23.63	64.12	5.08	3.66	20.67	19.31	18.10	81.68	12.36	**81.04**	54.64	0.52	73.47	20.65	**22.30**	4.03	15.79	10.73	31.50 ±0.21
		PØDA	**84.34**	**36.73**	**79.43**	**18.33**	16.54	**36.93**	**38.45**	**33.81**	**82.44**	**19.14**	75.90	62.65	**16.47**	75.48	15.68	19.57	**11.28**	16.53	**21.76**	**40.08** ±0.52

Limitations ?

- Global stylization: large structural classes benefit more from it

Source	Target eval.	Method	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle	mIoU%
		TrgPrompt = “driving at night”																				
CS	ACDC Night	source-only	70.42	18.32	**43.83**	6.11	17.08	23.52	**24.51**	19.76	39.74	6.11	0.78	21.62	8.96	23.08	2.53	0.00	3.27	8.42	9.87	18.31
		CLIPstyler	73.96	23.26	42.16	3.31	7.21	**35.49**	23.34	19.01	**45.41**	8.81	**27.87**	21.06	8.48	38.17	1.84	0.00	11.54	**10.38**	4.89	21.38 ±0.36
		PØDA	**77.54**	**26.90**	42.71	**13.51**	**21.36**	33.52	23.70	**21.73**	39.91	**9.51**	19.40	**28.80**	**11.85**	**50.89**	**10.14**	0.00	**20.76**	8.76	**14.50**	**25.03** ±0.48
		TrgPrompt = “driving in snow”																				
	ACDC snow	source-only	70.47	23.50	63.80	17.96	**27.36**	**38.52**	**56.26**	**45.00**	**83.00**	10.75	83.65	47.73	0.72	61.42	21.87	5.90	21.58	35.83	31.01	39.28
		CLIPstyler	74.29	31.25	69.17	15.21	25.21	36.83	44.79	42.56	76.87	11.07	91.48	**53.23**	0.13	67.66	23.88	**9.14**	36.48	**42.67**	28.76	41.09 ±0.17
		PØDA	**75.40**	**34.61**	**75.22**	**26.77**	27.34	35.20	52.68	44.37	82.01	**14.16**	**93.72**	50.51	**0.99**	**69.11**	**26.64**	2.72	**46.98**	42.64	**33.09**	**43.90** ±0.53
		TrgPrompt = “driving under rain”																				
	ACDC rain	source-only	74.10	31.98	63.07	**15.08**	**23.92**	**41.31**	**50.12**	**44.43**	79.93	22.07	87.45	47.99	4.39	68.92	10.35	18.52	13.64	7.03	21.58	38.20
		CLIPstyler	73.71	36.09	68.91	3.77	16.99	36.94	39.75	36.44	78.21	20.64	91.79	40.34	**9.65**	**74.54**	13.16	**20.33**	12.73	14.06	18.26	37.17 ±0.10
		PØDA	**76.60**	**38.52**	**78.01**	15.02	22.53	40.33	45.39	41.40	**86.85**	**37.97**	**96.46**	**50.39**	6.35	74.19	**19.19**	7.98	**22.06**	**21.04**	**23.65**	**42.31** ±0.55
		TrgPrompt = “driving in a game”																				
	GTA5	source-only	68.72	22.65	78.79	36.81	**17.31**	39.66	**39.33**	14.84	**72.61**	22.53	87.31	57.50	26.14	74.29	**44.57**	**20.45**	0.00	18.30	10.35	39.59
		CLIPstyler	73.06	**29.89**	77.86	25.50	11.69	**39.72**	35.88	**24.04**	67.38	12.75	88.77	46.58	33.38	72.03	42.79	11.12	0.00	28.84	**14.61**	38.73 ±0.16
		PØDA	**73.93**	22.69	**78.82**	**37.52**	14.17	36.97	33.14	17.34	72.44	**26.22**	**88.85**	**62.69**	**37.04**	**74.33**	43.03	11.91	0.00	**35.33**	13.91	**41.07** ±0.48
		TrgPrompt = “driving”																				
GTA5	CS	source-only	58.97	20.92	72.84	16.53	**24.58**	31.37	34.77	23.62	82.12	17.04	66.28	**63.46**	14.72	**81.27**	**20.83**	17.19	4.68	**20.57**	19.56	36.38
		CLIPstyler	66.70	23.63	64.12	5.08	3.66	20.67	19.31	18.10	81.68	12.36	**81.04**	54.64	0.52	73.47	20.65	**22.30**	4.03	15.79	10.73	31.50 ±0.21
		PØDA	**84.34**	**36.73**	**79.43**	**18.33**	16.54	**36.93**	**38.45**	**33.81**	**82.44**	**19.14**	75.90	62.65	**16.47**	75.48	15.68	19.57	**11.28**	16.53	**21.76**	**40.08** ±0.52

Limitations ?

- Global stylization: large structural classes benefit more from it
- Spatial-/Semantic- invariant transformation

	fog
Src only	44.73
Foggy driving	43.50(+-0.17)

Relevant

	fog
Src only	44.73
Foggy driving	43.50(+-0.17)
Driving in low visibility	44.40(+-0.17)
Operating a vehicle in thick fog	44.13(+-0.14)
Navigating through a foggy environment	43.68(+-0.07)
Driving in a dense fog	44.37(+-0.12)
Piloting a car when visibility is limited due to fog	44.16(+-0.18)

Relevant

	fog
Src only	44.73
Foggy driving	43.50(+-0.17)
Driving in low visibility	44.40(+-0.17)
Operating a vehicle in thick fog	44.13(+-0.14)
Navigating through a foggy environment	43.68(+-0.07)
Driving in a dense fog	44.37(+-0.12)
Piloting a car when visibility is limited due to fog	44.16(+-0.18)

Relevant

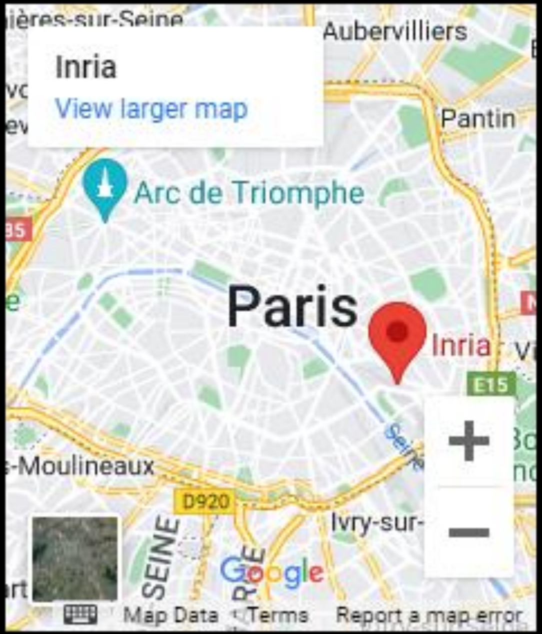
"Mesmerizing Northern Lights display"	41.95(+-0.30)
"Adorable baby's first steps"	**45.08**(+-0.15)
"Intense athlete mid-competition"	**45.79**(+-0.16)
"Playful dolphins in the ocean"	41.72(+-0.23)
"Breathtaking view from mountaintop"	42.88(+-0.22)

Irrelevant

Astra-Vision

Inria valeo.ai

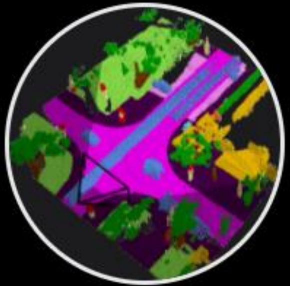
2D/3D Robust Scene Understanding



Inria Paris

Learning with less supervision

Vision in complex conditions



3D scene understanding

Regular job openings.

astra-vision.github.io

DenseMTL, in *WACV 23.* Lopes, Vu, de Charette
github.com/astra-vision/DenseMTL

xMUDA, in *TPAMI 22 & CVPR 20.* Jaritz, Vu, de Charette, Wirbel, Pérez.
https://github.com/valeoai/xmuda_journal

PØDA, in ICCV 23. Fahes, Vu, Bursuc, Pérez, de Charette
github.com/astra-vision/PODA

Any Questions ?

rdecharette.github.io | raoul.de-charette@inria.fr

Group page: **astra-vision.github.io**