Domain Adaptation for Semantic Segmentation via Class-Balanced Self-Training

Presenter: Zhiding Yu

Learning & Perception Research Group

NVIDIA Research

Obtaining Per-Pixel Dense Labels is Hard

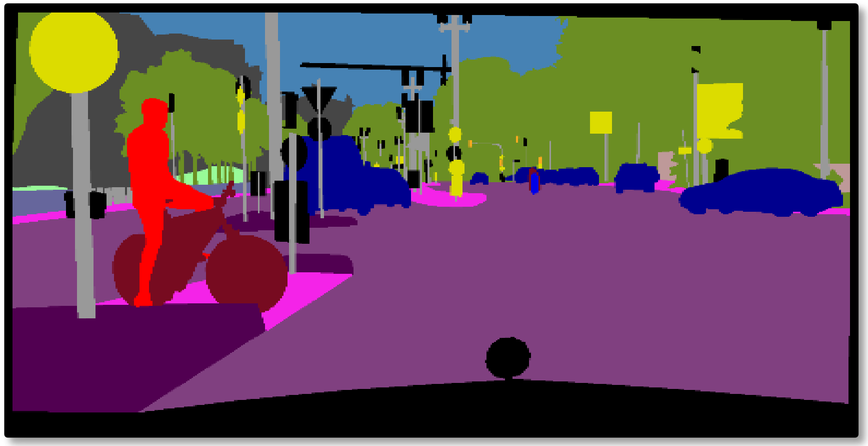
Real application often requires model robustness over scenes with large diversity

- Different cities, different weather, different views

Large scale annotated image data is beneficial

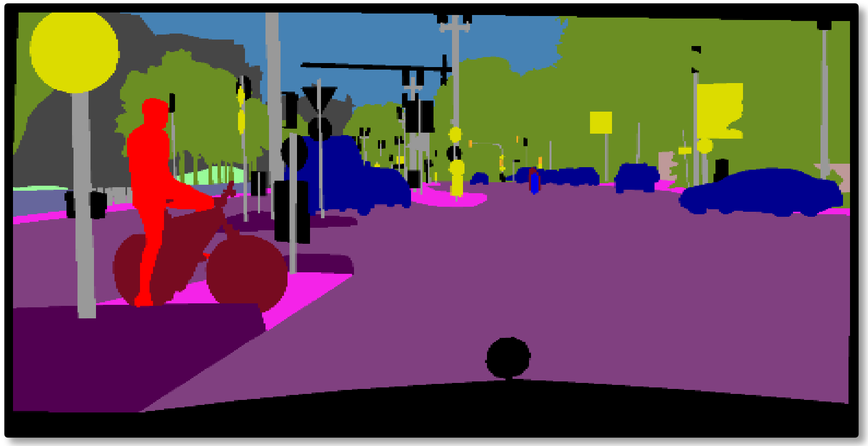
Annotating large scale real world image dataset is expensive

- Cityscapes dataset: 90 minutes per image



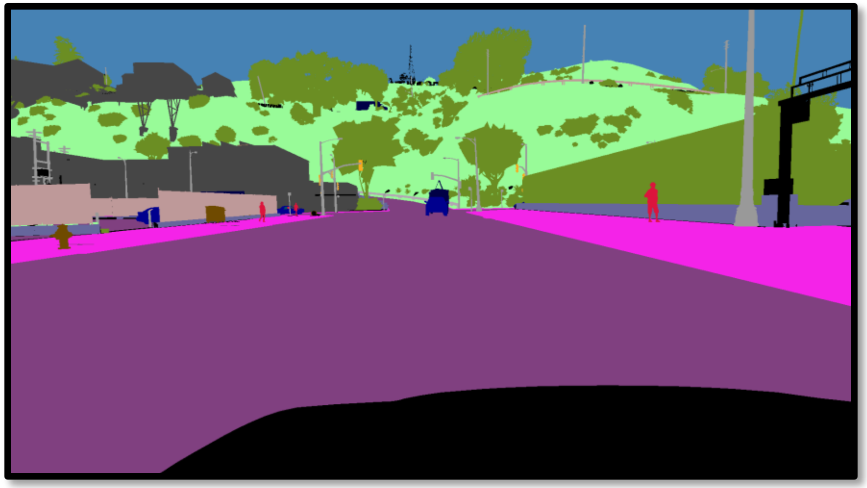
Use Synthetic Data to Obtain Infinite GTs?

Original image from Cityscapes



Human annotated ground truth

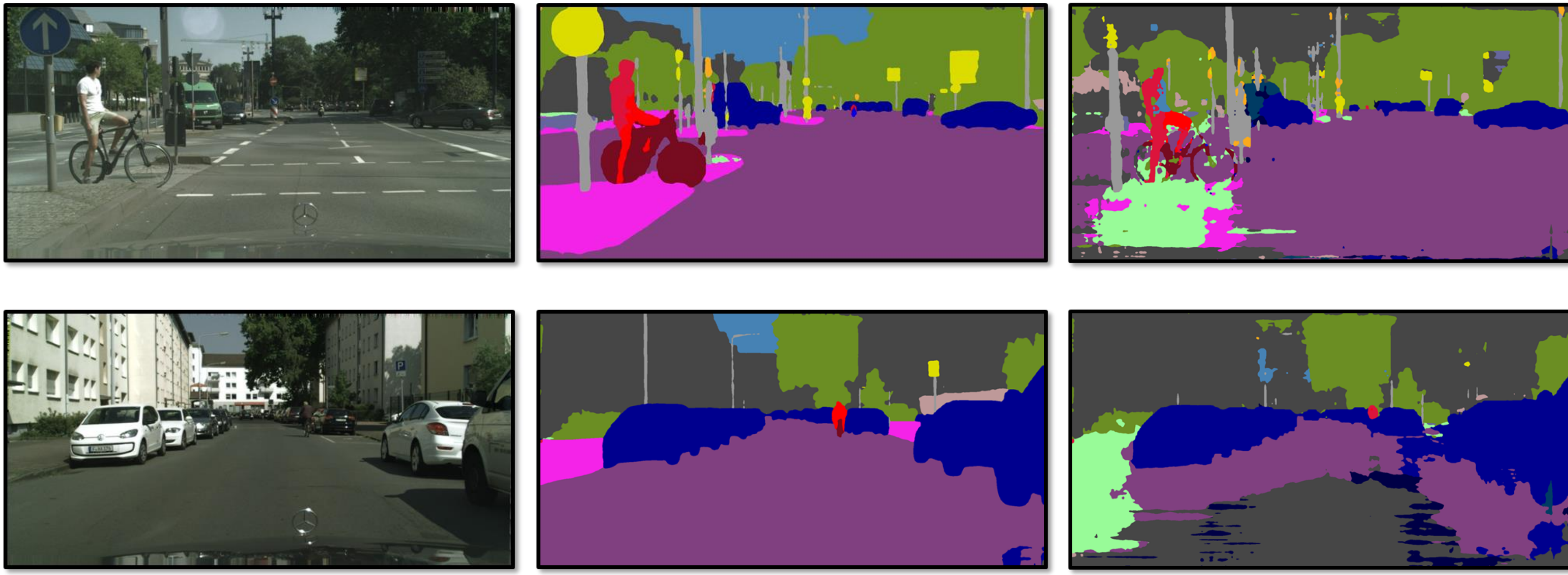
Original image from GTA5



Ground truth from game Engine

road	sidewalk	building	wall	fence	pole	traffic lgt	traffic sgn	vegetation	
terrain	sky	person	rider	car	truck	bus	train	motorcycle	bike

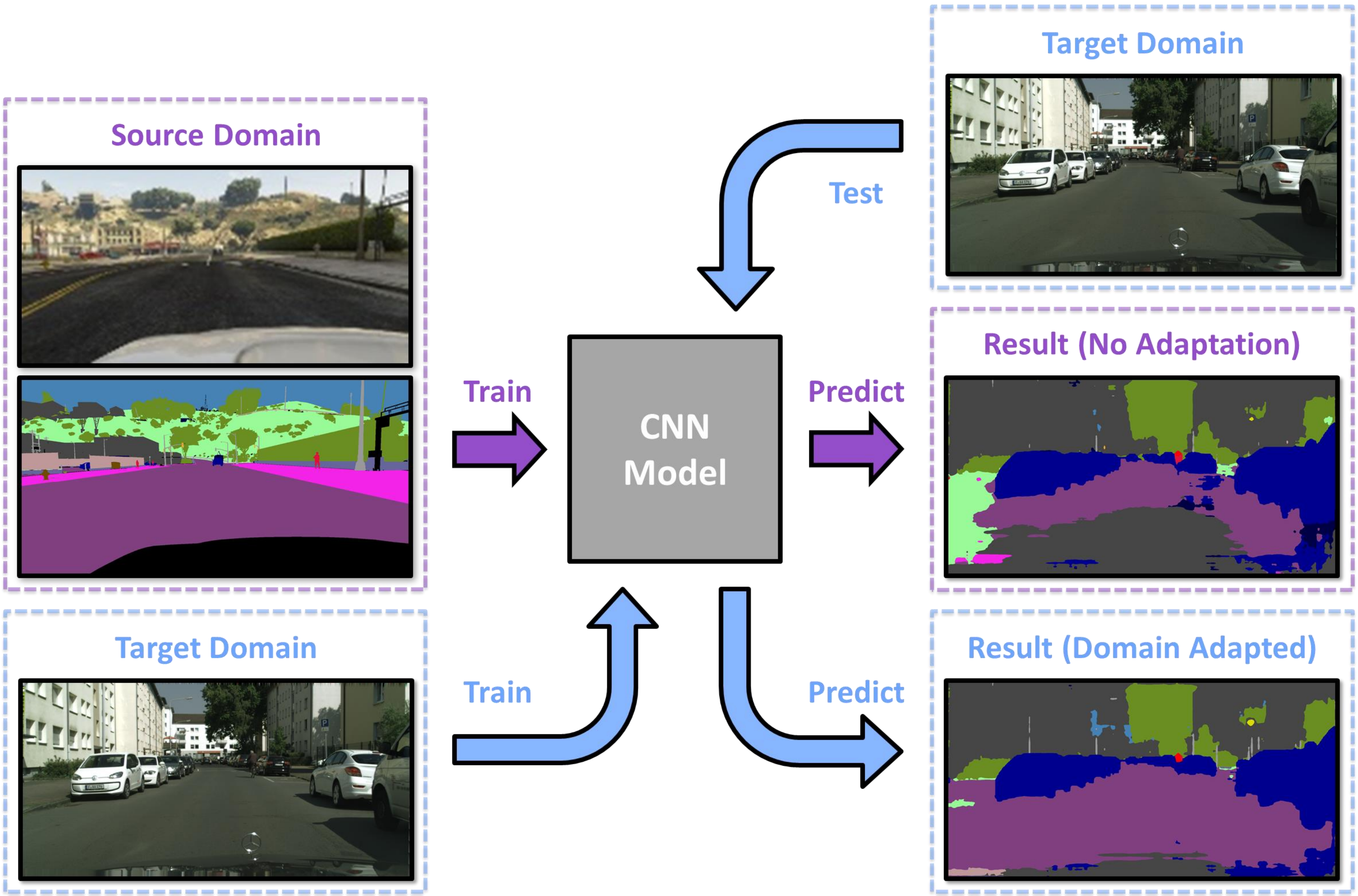
Drop of Performance Due to Domain Gaps



Cityscapes images | Model trained on Cityscapes | Model trained on GTA5

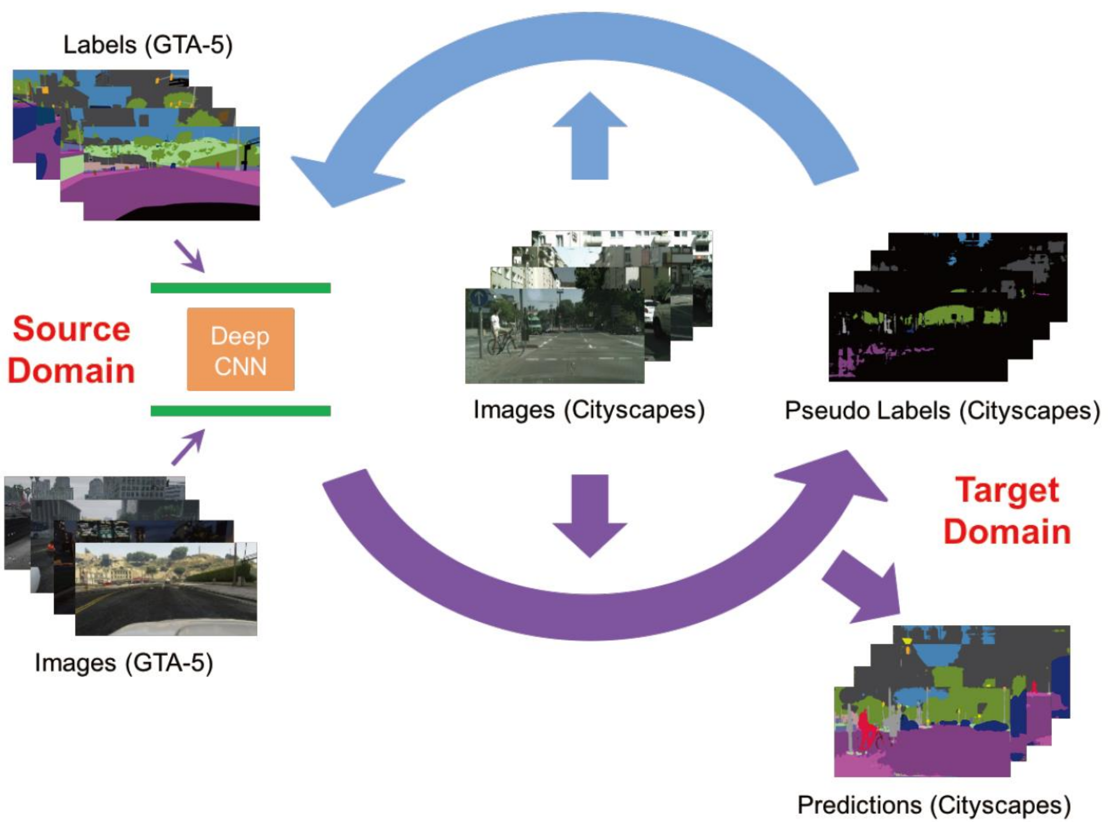
road	sidewalk	building	wall	fence	pole	traffic lgt	traffic sgn	vegetation	
terrain	sky	person	rider	car	truck	bus	train	motorcycle	bike

Unsupervised Domain Adaptation



Proposed Iterative Framework

road	sidewalk	building	wall	fence	pole	traffic lgt	traffic sgn	vegetation	
terrain	sky	person	rider	car	truck	bus	train	motorcycle	bike



Before Adaptation

After Adaptation

Preliminaries and Definitions

Fine-tuning for Supervised Domain Adaptation

$$\min_{\mathbf{w}} \mathcal{L}_S(\mathbf{w}) = -\sum_{s=1}^{S}\sum_{n=1}^{N} \mathbf{y}_{s,n}^{\top} \log(\mathbf{p}_n(\mathbf{w}, \mathbf{I}_s)) - \sum_{t=1}^{T}\sum_{n=1}^{N} \mathbf{y}_{t,n}^{\top} \log(\mathbf{p}_n(\mathbf{w}, \mathbf{I}_t))$$

where: $\mathbf{I}$: input image (crop) $\mathbf{p}$: pixel class probability vector $\mathbf{y}$: pixel label vector

$\mathbf{w}$: network parameters s: source image index t: target image index

Self-Training for Unsupervised Domain Adaptation

$$\min_{\mathbf{w},\hat{\mathbf{y}}} \mathcal{L}_U(\mathbf{w}, \hat{\mathbf{y}}) = -\sum_{s=1}^{S}\sum_{n=1}^{N} \mathbf{y}_{s,n}^{\top} \log(\mathbf{p}_n(\mathbf{w}, \mathbf{I}_s)) - \sum_{t=1}^{T}\sum_{n=1}^{N} \hat{\mathbf{y}}_{t,n}^{\top} \log(\mathbf{p}_n(\mathbf{w}, \mathbf{I}_t))$$

$$s.t.\ \hat{\mathbf{y}}_{t,n} \in \{\mathbf{e}^{(i)} | \mathbf{e}^{(i)} \in \mathbb{R}^C\}, \forall t, n$$

where: $\hat{\mathbf{y}}$: pseudo label vector $\mathbf{e}^{(i)}$: one-hot vector

The Vanilla Self-Training (ST) Framework

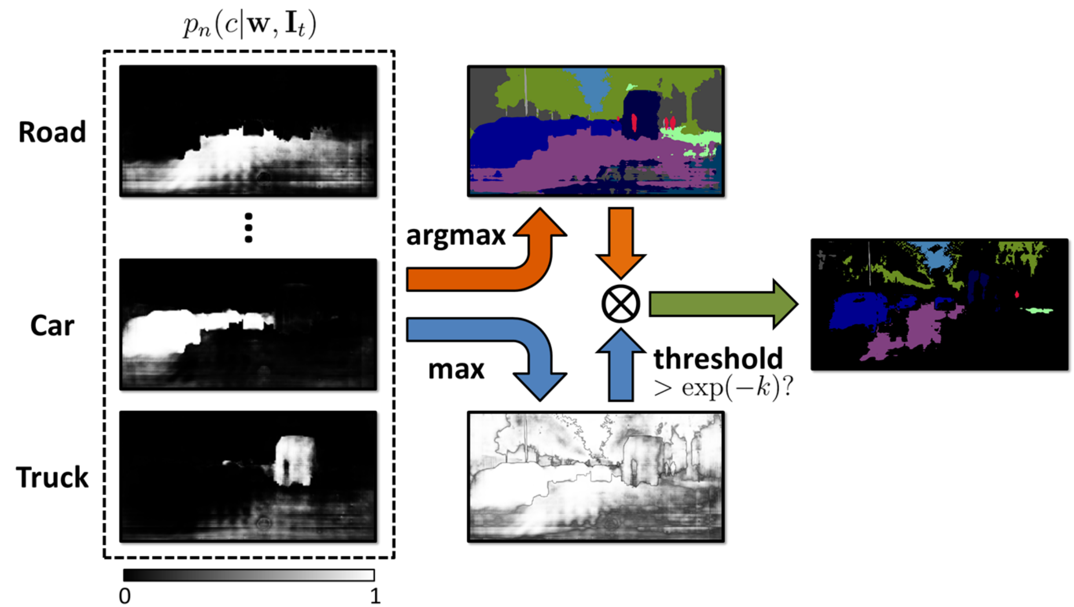
$$\min_{\mathbf{w},\hat{\mathbf{y}}} \mathcal{L}_{ST}(\mathbf{w},\hat{\mathbf{y}}) = -\sum_{s=1}^{S}\sum_{n=1}^{N}\mathbf{y}_{s,n}^{\top}\log(\mathbf{p}_n(\mathbf{w},\mathbf{I}_s)) - \sum_{t=1}^{T}\sum_{n=1}^{N}\left[\hat{\mathbf{y}}_{t,n}^{\top}\log(\mathbf{p}_n(\mathbf{w},\mathbf{I}_t)) + k|\hat{\mathbf{y}}_{t,n}|_1\right]$$

$$s.t.\ \hat{\mathbf{y}}_{t,n} \in \{\{\mathbf{e}^{(i)}|\mathbf{e}^{(i)} \in \mathbb{R}^C\} \cup \mathbf{0}\}, \forall t,n$$

$$k > 0$$

The cost can be minimized via mixed integer programming, which leads to the following solution:

$$\hat{y}_{t,n}^{(c)*} = \begin{cases} 1, \ \textbf{if}\ c = \arg\max_{c} p_n(c|\mathbf{w},\mathbf{I}_t), \\ \qquad p_n(c|\mathbf{w},\mathbf{I}_t) > \exp(-k) \\ 0, \ \text{otherwise} \end{cases}$$



Class-Balanced Self-Training (CBST)

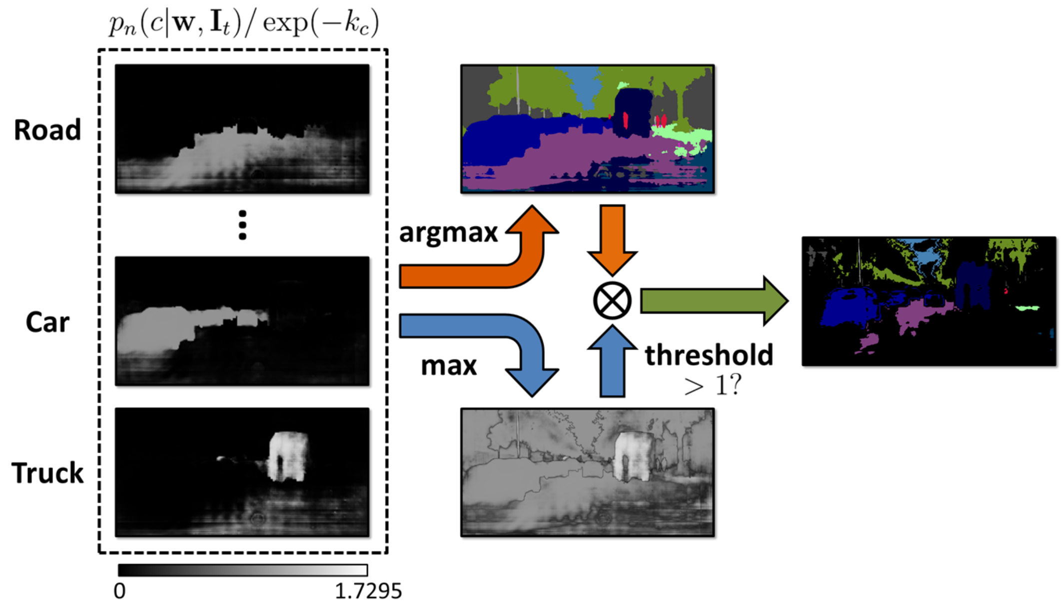
$$\min_{\mathbf{w},\hat{\mathbf{y}}} \mathcal{L}_{CB}(\mathbf{w},\hat{\mathbf{y}}) = -\sum_{s=1}^{S}\sum_{n=1}^{N} \mathbf{y}_{s,n}^{\top} \log(\mathbf{p}_n(\mathbf{w},\mathbf{I}_s)) - \sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{c=1}^{C} \left[\hat{y}_{t,n}^{(c)} \log(p_n(c|\mathbf{w},\mathbf{I}_t)) + k_c \hat{y}_{t,n}^{(c)}\right]$$

$$s.t.\ \hat{\mathbf{y}}_{t,n} = \left[\hat{y}_{t,n}^{(1)}, ..., \hat{y}_{t,n}^{(C)}\right] \in \{\{\mathbf{e}^{(i)}|\mathbf{e}^{(i)} \in \mathbb{R}^C\} \cup \mathbf{0}\}, \forall t, n$$

$$k_c > 0, \forall c$$

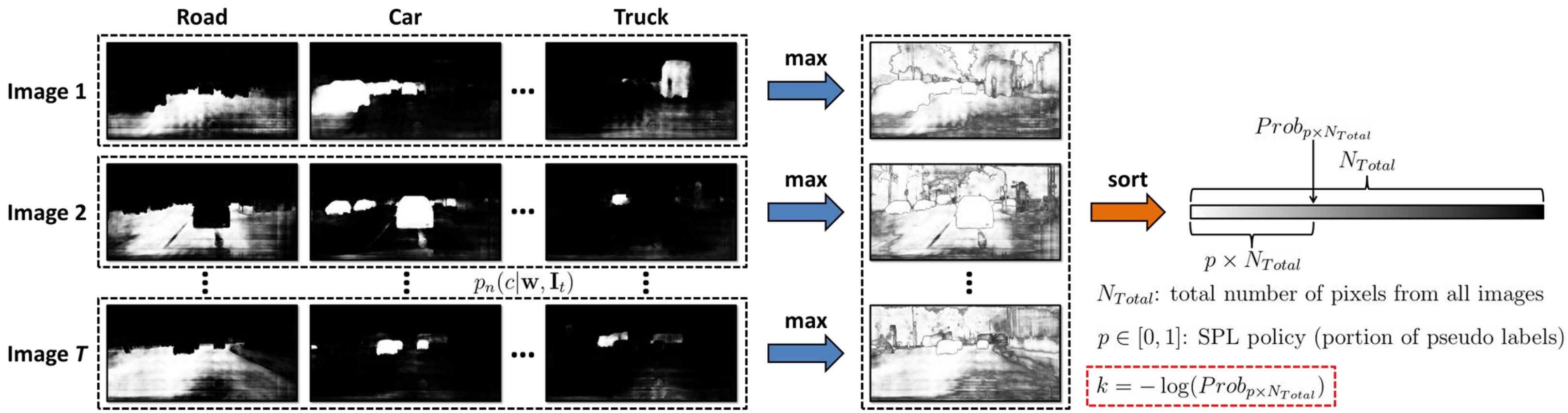
Again using mixed integer programming, one obtains the following solution:

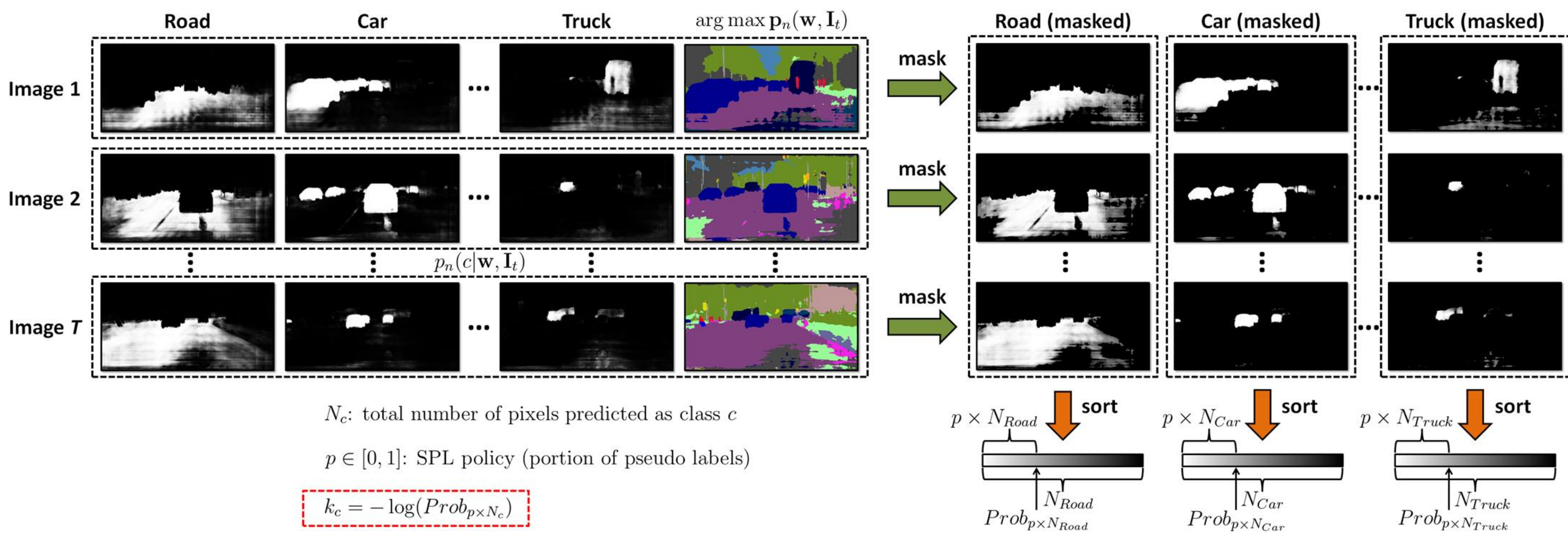
$$\hat{y}_{t,n}^{(c)*} = \begin{cases} 1, \ \textbf{if} \ c = \arg\max\limits_{c} \dfrac{p_n(c|\mathbf{w},\mathbf{I}_t)}{\exp(-k_c)}, \\ \qquad \dfrac{p_n(c|\mathbf{w},\mathbf{I}_t)}{\exp(-k_c)} > 1 \\ 0, \ \text{otherwise} \end{cases}$$



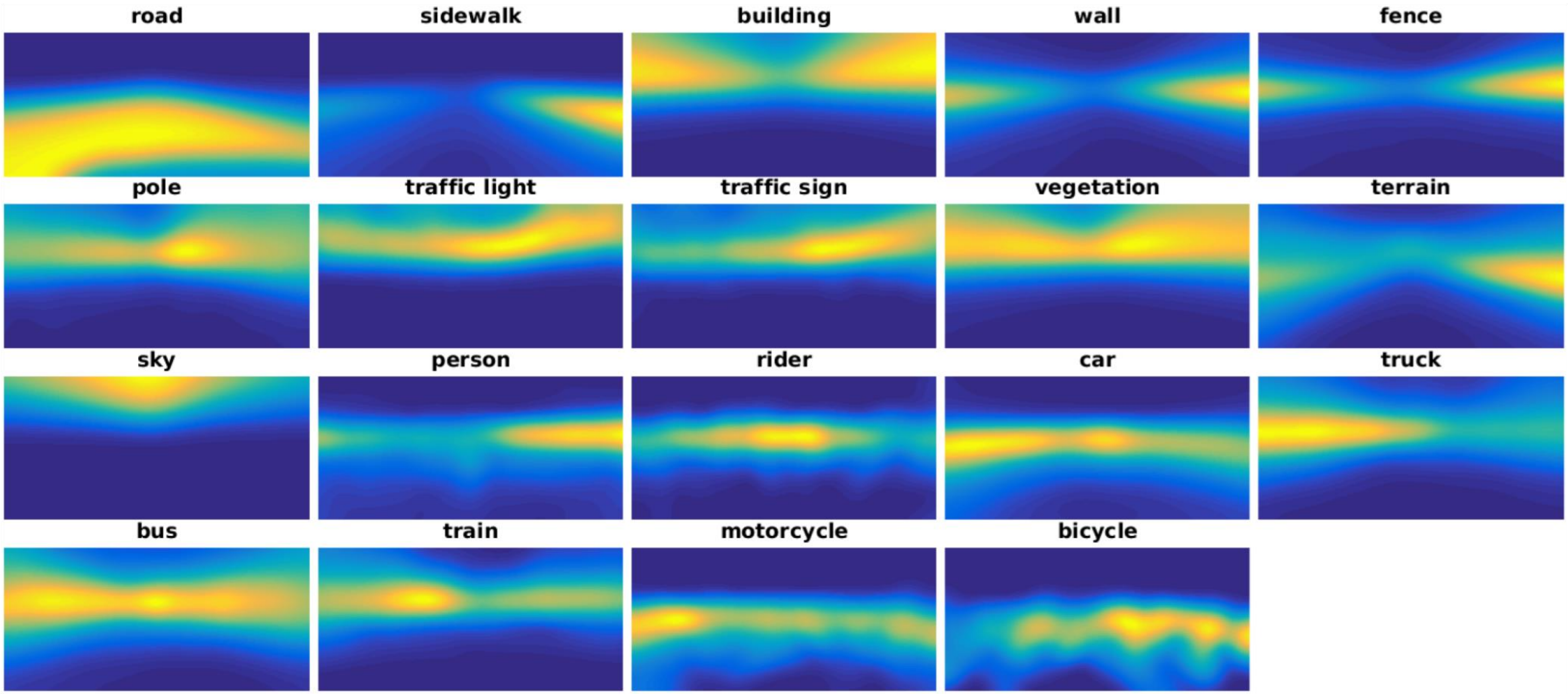
Self-Paced Learning Policy Design

The both k and k_c in ST and CBST can be easily determined with a single SPL policy parameter p:





Incorporating Spatial Priors (CBST-SP)



$$\min_{\mathbf{w},\hat{\mathbf{y}}} \mathcal{L}_{SP}(\mathbf{w},\hat{\mathbf{y}}) = -\sum_{s=1}^{S}\sum_{n=1}^{N} \mathbf{y}_{s,n}^{\top} \log(\mathbf{p}_n(\mathbf{w},\mathbf{I}_s)) - \sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{c=1}^{C} \left[\hat{y}_{t,n}^{(c)} \log(q_n(c) p_n(c|\mathbf{w},\mathbf{I}_t)) + k_c \hat{y}_{t,n}^{(c)}\right]$$

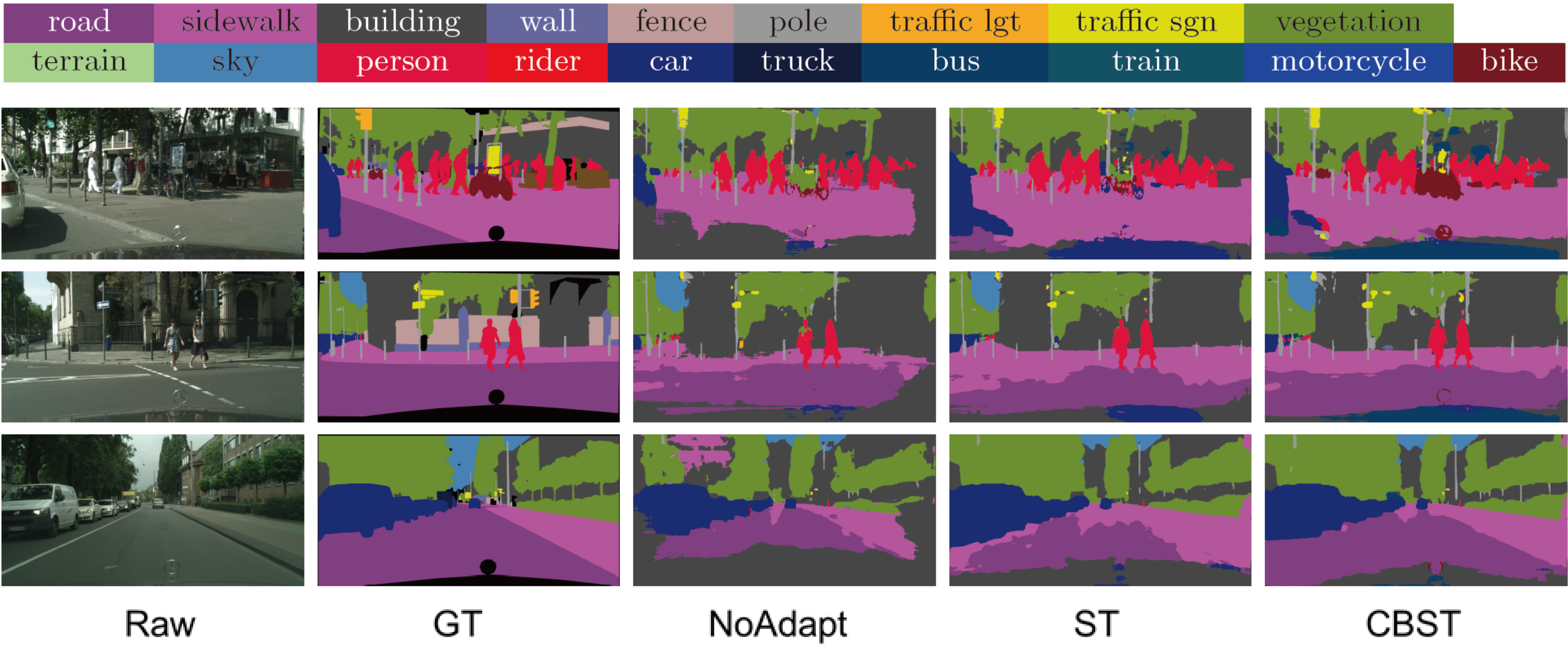
$$s.t.\ \hat{\mathbf{y}}_{t,n} \in \{\{\mathbf{e}|\mathbf{e} \in \mathbb{R}^C\} \cup \mathbf{0}\}, \forall t, n$$

$$k_c > 0, \forall c$$

Experiment: Cityscapes → NTHU

City	Method	Road	SW	Build	TL	TS	Veg.	Sky	PR	Rider	Car	Bus	Motor	Bike	Mean
Rome	Source Dilation-Frontend [10]	77.7	21.9	83.5	0.1	10.7	78.9	88.1	21.6	10.0	67.2	30.4	6.1	0.6	38.2
	GCAA [10]	79.5	29.3	84.5	0.0	22.2	80.6	82.8	29.5	13.0	71.7	37.5	25.9	1.0	42.9
	DeepLab-v2 [36]	83.9	34.3	87.7	13.0	41.9	84.6	92.5	37.7	22.4	80.8	38.1	39.1	5.3	50.9
	MAA [36]	83.9	34.2	88.3	**18.8**	40.2	**86.2**	**93.1**	47.8	21.7	80.9	**47.8**	48.3	8.6	**53.8**
	Source Resnet-38	86.0	21.4	81.5	14.3	47.4	82.9	59.8	30.8	20.9	83.1	20.2	40.0	5.6	45.7
	ST	85.9	20.2	84.3	15.0	46.4	84.9	73.5	**48.5**	21.6	84.6	17.6	46.2	6.7	48.9
	CBST	**87.1**	**43.9**	**89.7**	14.8	**47.7**	85.4	90.3	45.4	**26.6**	**85.4**	20.5	**49.8**	**10.3**	53.6
Rio	Source Dilation-Frontend [10]	69.0	31.8	77.0	4.7	3.7	71.8	80.8	38.2	8.0	61.2	38.9	11.5	3.4	38.5
	GCAA [10]	74.2	43.9	79.0	2.4	7.5	77.8	69.5	39.3	10.3	67.9	**41.2**	27.9	10.9	42.5
	DeepLab-v2 [36]	76.6	47.3	82.5	12.6	22.5	77.9	86.5	43.0	19.8	74.5	36.8	29.4	16.7	48.2
	MAA [36]	76.2	44.7	84.6	9.3	25.5	**81.8**	**87.3**	55.3	**32.7**	74.3	28.9	**43.0**	**27.6**	51.6
	Source Resnet-38	80.6	36.0	81.8	**21.0**	33.1	79.0	64.7	36.0	21.0	73.1	33.6	22.5	7.8	45.4
	ST	80.1	41.4	83.8	19.1	**39.1**	80.8	71.2	**56.3**	27.7	**79.9**	32.7	36.4	12.2	50.8
	CBST	**84.3**	**55.2**	**85.4**	19.6	30.1	80.5	77.9	55.2	28.6	79.7	33.2	37.6	11.5	**52.2**
Tokyo	Source Dilation-Frontend [10]	81.2	26.7	71.7	8.7	5.6	73.2	75.7	39.3	14.9	57.6	19.0	1.6	33.8	39.2
	GCAA [10]	83.4	**35.4**	72.8	12.3	12.7	77.4	64.3	42.7	21.5	64.1	**20.8**	8.9	40.3	42.8
	DeepLab-v2 [36]	83.4	35.4	72.8	12.3	12.7	77.4	64.3	42.7	21.5	64.1	**20.8**	8.9	40.3	42.8
	MAA [36]	81.5	26.0	77.8	**17.8**	26.8	82.7	**90.9**	55.8	**38.0**	72.1	4.2	24.5	**50.8**	**49.9**
	Source Resnet-38	83.8	26.4	73.0	6.5	27.0	80.5	46.6	35.6	22.8	71.3	4.2	10.5	36.1	40.3
	ST	83.1	27.7	74.8	7.1	29.4	**84.4**	48.5	**57.2**	23.3	**73.3**	3.3	22.7	45.8	44.6
	CBST	**85.2**	33.6	**80.4**	8.3	**31.1**	83.9	78.2	53.2	28.9	72.7	4.4	**27.0**	47.0	48.8
Taipei	Source Dilation-Frontend [10]	77.2	20.9	76.0	5.9	4.3	60.3	81.4	10.9	11.0	54.9	32.6	15.3	5.2	35.1
	GCAA [10]	78.6	28.6	80.0	13.1	7.6	68.2	82.1	16.8	9.4	60.4	34.0	26.5	9.9	39.6
	DeepLab-v2 [36]	78.6	28.6	80.0	13.1	7.6	68.2	82.1	16.8	9.4	60.4	34.0	26.5	9.9	39.6
	MAA [36]	81.7	29.5	**85.2**	**26.4**	15.6	**76.7**	**91.7**	31.0	12.5	71.5	**41.1**	47.3	27.7	49.1
	Source Resnet-38	84.9	26.0	80.1	8.3	**28.0**	73.9	54.4	18.9	26.8	71.6	26.0	48.2	14.7	43.2
	ST	83.1	23.5	78.2	9.6	25.4	74.8	35.9	**33.2**	27.3	75.2	32.3	52.2	28.8	44.6
	CBST	**86.1**	**35.2**	84.2	15.0	22.2	75.6	74.9	22.7	**33.1**	**78.0**	37.6	**58.0**	**30.9**	**50.3**

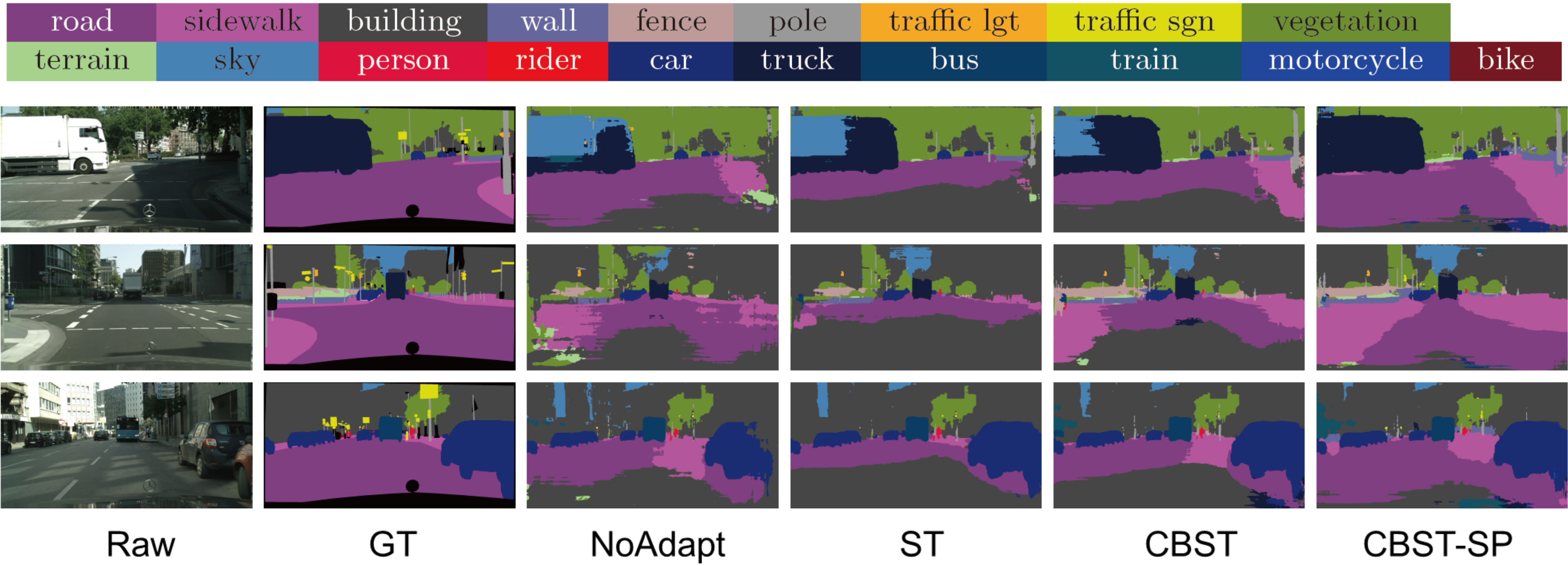
Experiment: SYNTHIA → Cityscapes



Experiment: SYNTHIA → Cityscapes

Method	Base Net	Road	SW	Build	Wall*	Fence*	Pole*	TL	TS	Veg.	Sky	PR	Rider	Car	Bus	Motor	Bike	mIoU	mIoU*
Source only [18]	Dilation-Frontend	6.4	17.7	29.7	1.2	0.0	15.1	0.0	7.2	30.3	66.8	51.1	1.5	47.3	3.9	0.1	0.0	17.4	20.2
FCN wild [18]	[43]	11.5	19.6	30.8	4.4	0.0	20.3	0.1	11.7	42.3	68.7	51.2	3.8	54.0	3.2	0.2	0.6	20.2	22.1
Source only [45]	FCN8s-VGG16	5.6	11.2	59.6	8.0	**0.5**	21.5	8.0	5.3	72.4	75.6	35.1	9.0	23.6	4.5	0.5	18.0	22.0	27.6
Curr. DA [45]	[21]	65.2	26.1	74.9	0.1	**0.5**	10.7	3.5	3.0	76.1	70.6	47.1	8.2	43.2	20.7	0.7	13.1	29.0	34.8
Source only	FCN8s-VGG16	24.1	19.1	68.5	0.9	0.3	16.4	5.7	10.8	75.2	76.3	43.2	15.2	26.7	15.0	5.9	8.5	25.7	30.3
GAN DA	[21]	79.1	31.1	77.1	3.0	0.2	22.8	6.6	15.2	77.4	78.9	47.0	14.8	67.5	16.3	6.9	13.0	34.8	40.8
Source only	DeepLab-v2 [36]	55.6	23.8	74.6	–	–	–	6.1	12.1	74.8	79.0	55.3	19.1	39.6	23.3	13.7	25.0	–	38.6
MAA	[36]	**84.3**	**42.7**	**77.5**	–	–	–	4.7	7.0	77.9	**82.5**	54.3	**21.0**	72.3	**32.2**	**18.9**	32.3	–	46.7
Source only	FCN8s-VGG16	17.2	19.7	47.3	1.1	0.0	19.1	3.0	9.1	71.8	78.3	37.6	4.7	42.2	9.0	0.1	0.9	22.6	26.2
ST	[21]	0.2	14.5	53.8	1.6	0.0	18.9	0.9	7.8	72.2	80.3	48.1	6.3	67.7	4.7	0.2	4.5	23.9	27.8
CBST		69.6	28.7	69.5	12.1	0.1	25.4	11.9	13.6	82.0	81.9	49.1	14.5	66.0	6.6	3.7	32.4	35.4	36.1
Source only	ResNet-38	32.6	21.5	46.5	4.8	0.1	26.5	14.8	13.1	70.8	60.3	56.6	3.5	74.1	20.4	8.9	13.1	29.2	33.6
ST	[41]	38.2	19.6	70.2	3.9	0.0	31.9	17.6	17.2	82.4	68.3	63.1	5.3	78.4	11.2	0.8	7.5	32.2	36.9
CBST		53.6	23.7	75.0	**12.5**	0.3	**36.4**	**23.5**	**26.3**	**84.8**	74.7	**67.2**	17.5	**84.5**	28.4	15.2	**55.8**	**42.5**	**48.4**

Experiment: GTA5 → Cityscapes

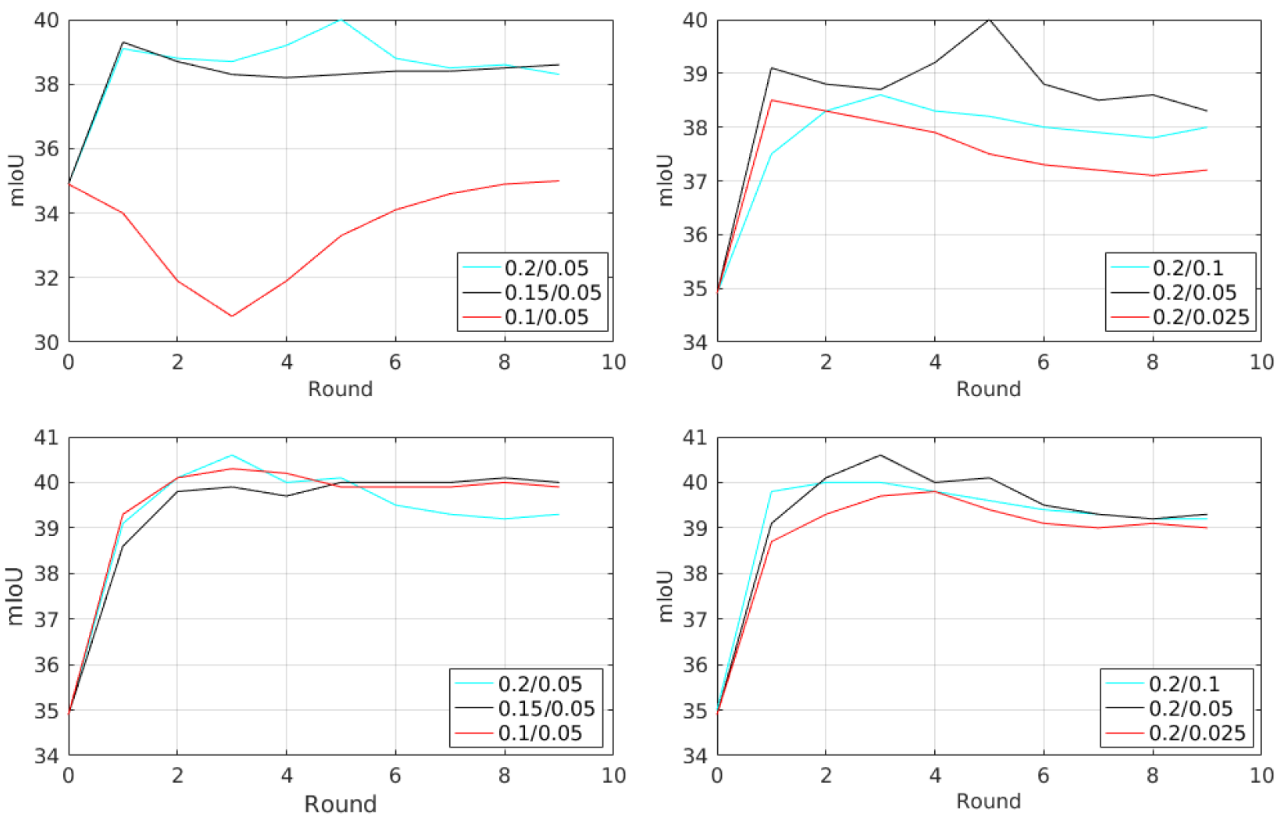


Experiment: GTA5 → Cityscapes

Method	Base Net	Road	SW	Build	Wall	Fence	Pole	TL	TS	Veg.	Terrain	Sky	PR	Rider	Car	Truck	Bus	Train	Motor	Bike	mIoU
Source only [18]	Dilation-Frontend	31.9	18.9	47.7	7.4	3.1	16.0	10.4	1.0	76.5	13.0	58.9	36.0	1.0	67.1	9.5	3.7	0.0	0.0	0.0	21.2
FCN wild [18]	[43]	70.4	32.4	62.1	14.9	5.4	10.9	14.2	2.7	79.2	21.3	64.6	44.1	4.2	70.4	8.0	7.3	0.0	3.5	0.0	27.1
Source only [45]	FCN8s-VGG16	18.1	6.8	64.1	7.3	8.7	21.0	14.9	16.8	45.9	2.4	64.4	41.6	17.5	55.3	8.4	5.0	6.9	4.3	13.8	22.3
Curr. DA [45]	[21]	74.9	22.0	71.7	6.0	11.9	8.4	16.3	11.1	75.7	13.3	66.5	38.0	9.3	55.2	18.8	18.9	0.0	16.8	16.6	28.9
Source only [17]	FCN8s-VGG16	26.0	14.9	65.1	5.5	12.9	8.9	6.0	2.5	70.0	2.9	47.0	24.5	0.0	40.0	12.1	1.5	0.0	0.0	0.0	17.9
CyCADA [17]	[21]	85.2	37.2	76.5	21.8	15.0	23.8	22.9	21.5	80.5	31.3	60.7	50.5	9.0	76.9	17.1	28.2	4.5	9.8	0.0	35.4
Source only [17]	Dilated ResNet-26	42.7	26.3	51.7	5.5	6.8	13.8	23.6	6.9	75.5	11.5	36.8	49.3	0.9	46.7	3.4	5.0	0.0	5.0	1.4	21.7
CyCADA [17]	[44]	79.1	33.1	77.9	23.4	17.3	32.1	33.3	31.8	81.5	26.7	69.0	62.8	14.7	74.5	20.9	25.6	6.9	18.8	20.4	39.5
Source only [30]	ResNet-50	64.5	24.9	73.7	14.8	2.5	18.0	15.9	0	74.9	16.4	72.0	42.3	0.0	39.5	8.6	13.4	0.0	0.0	0.0	25.3
ADR [30]	[16]	87.8	15.6	77.4	20.6	9.7	19.0	19.9	7.7	82.0	31.5	74.3	43.5	9.0	77.8	17.5	27.7	1.8	9.7	0.0	33.3
Source only [24]	DenseNet	67.3	23.1	69.4	13.9	14.4	21.6	19.2	12.4	78.7	24.5	74.8	49.3	3.7	54.1	8.7	5.3	2.6	6.2	1.9	29.0
I2I Adapt [24]	[19]	85.8	37.5	80.2	23.3	16.1	23.0	14.5	9.8	79.2	**36.5**	**76.4**	53.4	7.4	82.8	19.1	15.7	2.8	13.4	1.7	35.7
Source only [36]	DeepLab-v2	75.8	16.8	77.2	12.5	21.0	25.5	30.1	20.1	81.3	24.6	70.3	53.8	26.4	49.9	17.2	25.9	6.5	25.3	36.0	36.6
MAA [36]	[19]	86.5	36.0	**79.9**	23.4	23.3	23.9	35.2	14.8	83.4	33.3	75.6	58.5	27.6	73.7	32.5	35.4	3.9	30.1	28.1	42.4
Source only	FCN8s-VGG16	64.0	22.1	68.6	13.3	8.7	19.9	15.5	5.9	74.9	13.4	37.0	37.7	10.3	48.2	6.1	1.2	1.8	10.8	2.9	24.3
ST	[18]	83.8	17.4	72.1	14.3	2.9	16.5	16.0	6.8	81.4	24.2	47.2	40.7	7.6	71.7	10.2	7.6	0.5	11.1	0.9	28.1
CBST		66.7	26.8	73.7	14.8	9.5	28.3	25.9	10.1	75.5	15.7	51.6	47.2	6.2	71.9	3.7	2.2	5.4	18.9	32.4	30.9
CBST-SP		**90.4**	50.8	72.0	18.3	9.5	27.2	28.6	14.1	82.4	25.1	70.8	42.6	14.5	76.9	5.9	12.5	1.2	14.0	28.6	36.1
Source only	ResNet-38	70.0	23.7	67.8	15.4	18.1	40.2	41.9	25.3	78.8	11.7	31.4	**62.9**	**29.8**	60.1	21.5	26.8	7.7	28.1	12.0	35.4
ST	[41]	90.1	56.8	77.9	28.5	23.0	41.5	45.2	39.6	84.8	26.4	49.2	59.0	27.4	82.3	39.7	45.6	**20.9**	**34.8**	**46.2**	41.5
CBST		86.8	46.7	76.9	26.3	**24.8**	42.0	46.0	38.6	80.7	15.7	48.0	57.3	27.9	78.2	24.5	49.6	17.7	25.5	45.1	45.2
CBST-SP		88.0	56.2	77.0	27.4	22.4	40.7	47.3	**40.9**	82.4	21.6	60.3	50.2	20.4	**83.8**	35.0	**51.0**	15.2	20.6	37.0	46.2
CBST-SP+MST		89.6	**58.9**	78.5	**33.0**	22.3	**41.4**	**48.2**	39.2	**83.6**	24.3	65.4	49.3	20.2	83.3	**39.0**	48.6	12.5	20.3	35.3	**47.0**

Experiment: GTA5 → BDD

Method	Road	SW	Build	Wall	Fence	Pole	TL	TS	Veg.	Terrain	Sky	PR	Rider	Car	Truck	Bus	Train	Motor	Bike	Mean
Source Resnet-38	76.7	34.1	53.8	10.2	28.3	29.1	34.1	33.9	73.4	17.5	60.8	52.8	15.2	63.8	40.78	28.8	0.0	21.3	2.6	35.0
ST	83.5	26.1	72.5	14.1	27.3	26.5	32.5	28.5	74.5	35.7	88.1	51.4	15.9	67.4	26.6	35.9	0.0	8.9	2.9	37.8
ST-SP	88.2	40.8	74.1	14.8	27.1	25.8	33.1	36.1	72.2	37.4	88.8	53.8	21.2	74.2	24.5	22.9	0.0	12.9	1.5	39.5
CBST	84.1	26.6	75.0	15.3	28.8	28.0	33.8	29.8	76.2	35.6	90.4	54.2	18.2	69.4	28.6	36.7	0.0	13.0	3.8	39.3
CBST-SP	89.9	39.3	73.9	14.9	28.0	28.7	34.1	35.6	76.7	34.9	89.6	57.4	19.8	77.3	27.1	28.1	0.0	13.8	1.7	40.6



p_0: Initial p value

Δp: Per round increment size

Legend: $p_0/\Delta p$

Thank You!