MetaCorrection: Domain-aware Meta Loss Correction for Unsupervised Domain Adaptation in Semantic Segmentation

Xiaoqing Guo[1*] Chen Yang[1*] Baopu Li[2] Yixuan Yuan[1†]

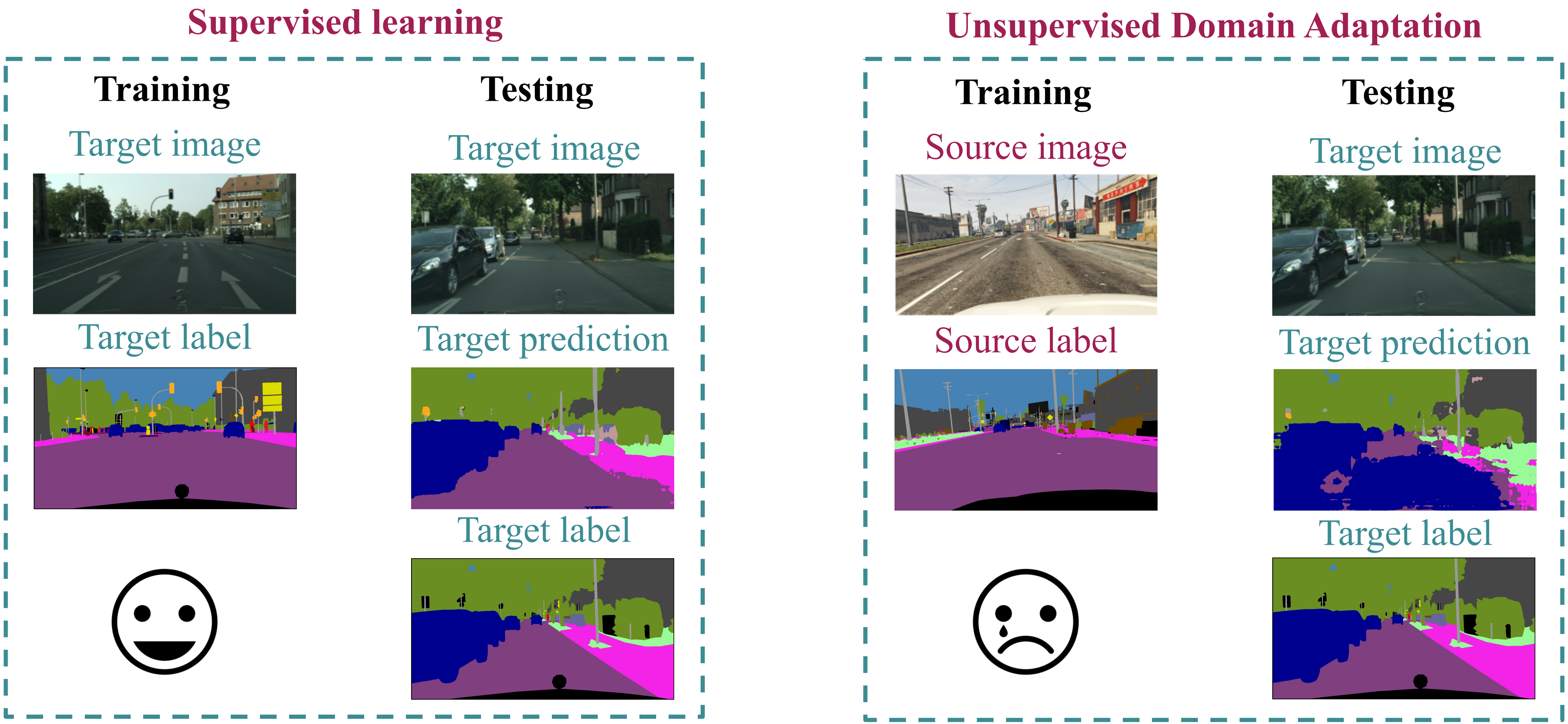
[1] City University of Hong Kong [2] Baidu USA

xqguo.ee@my.cityu.edu.hk cyang.eeg@my.cityu.edu.hk baopuli@baidu.com yxyuan.ee@cityu.edu.hk

* Xiaoqing Guo and Chen Yang contribute equally.
† Yixuan Yuan is the corresponding author

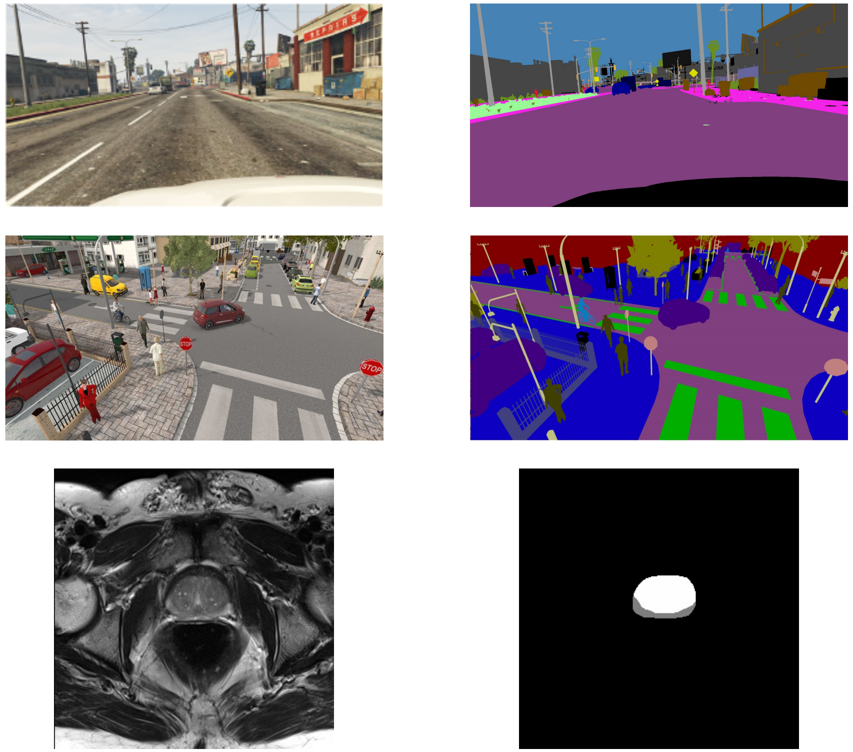
Introduction

➢ **Unsupervised Domain Adaptation (UDA) in Semantic Segmentation**



Introduction

- **Unsupervised Domain Adaptation (UDA) in Semantic Segmentation**



$X_{\mathcal{S}} = \{x_s \in \mathbb{R}^{H\times W\times 3}\}_{s\in\mathcal{S}}$ $Y_{\mathcal{S}} = \{y_s \in \{0,1\}^{H\times W\times C}\}_{s\in\mathcal{S}}$

Source Domain (Labeled)

Adaptation

$X_{\mathcal{T}} = \{x_t \in \mathbb{R}^{H\times W\times 3}\}_{t\in\mathcal{T}}$

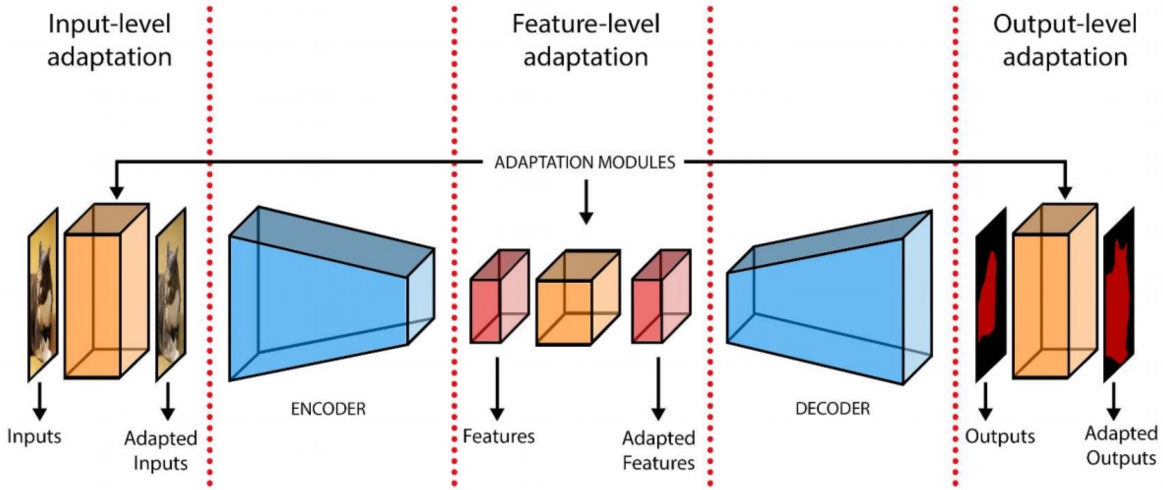
Target Domain (Unlabeled)

Introduction

Two major lines of approaches:

- **Adversarial learning**

  - Ignore the domain-specific knowledge
  - Could not guarantee the sufficient discriminative capability for the specific task.

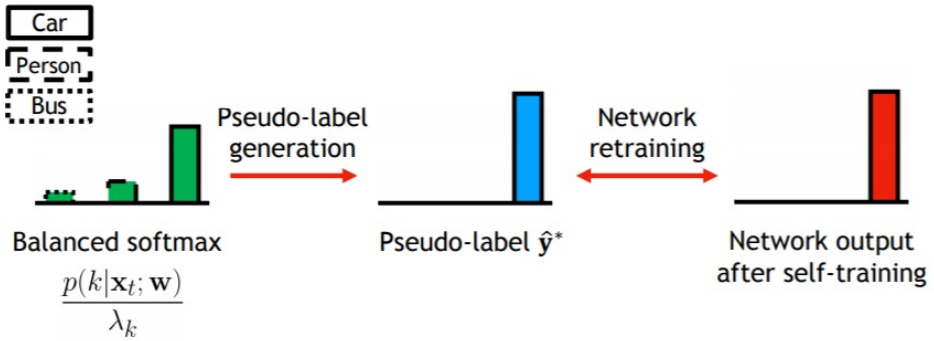


Adversarial learning for UDA in semantic segmentation [1]

- **Self-training**

  - The generated pseudo labels contain noises.

$$\mathcal{L}_{ST} = \mathcal{L}_{seg}^{\mathcal{S}}(X_{\mathcal{S}}, Y_{\mathcal{S}}) + \mathcal{L}_{seg}^{\mathcal{T}}(X_{\mathcal{T}}, \widehat{Y}_{\mathcal{T}}) = -\sum_{s\in\mathcal{S}} y_s \log f\left(x_s, \mathbf{w}\right) - \sum_{t\in\mathcal{T}} \widehat{y}_t \log f\left(x_t, \mathbf{w}\right).$$



Self-training for UDA in semantic segmentation [2]

[1] Toldo, Marco, et al. "Unsupervised domain adaptation in semantic segmentation: a review." Technologies 8.2 (2020): 35.
[2] Zou, Yang, et al. "Confidence regularized self-training." ICCV. 2019.

Motivation

- **Noise Transition Matrix (NTM)**
  - It aims to model the inter-class misclassification relationship of target data.
- **Domain-aware Meta-learning for Loss Correction (DMLC)**
  - Since ground truth in the target domain is not available, NTM can't be directly calculated, We try to estimate the NTM in a learning-to-learn fashion.

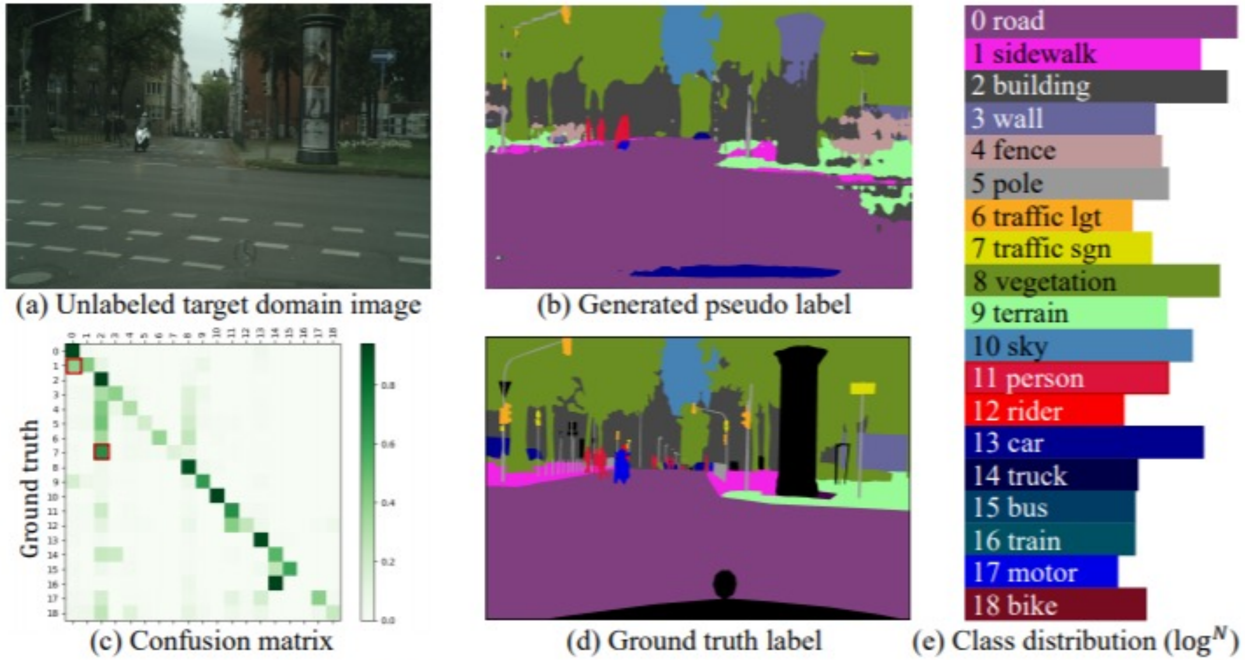
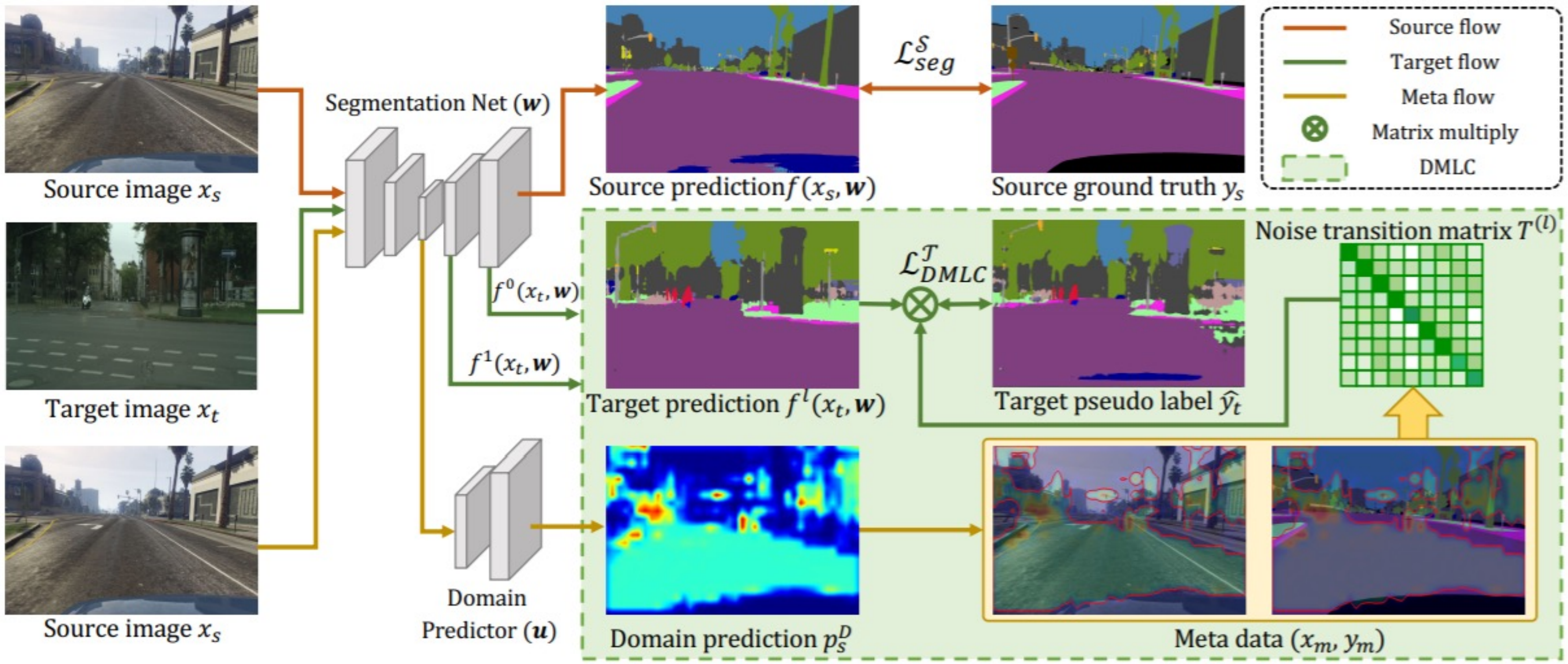


Figure 1. Sample of the noisy pseudo labels on Cityscapes [10]. The generated pseudo labels suffer from the data distribution biases in comparison to the ground truth.

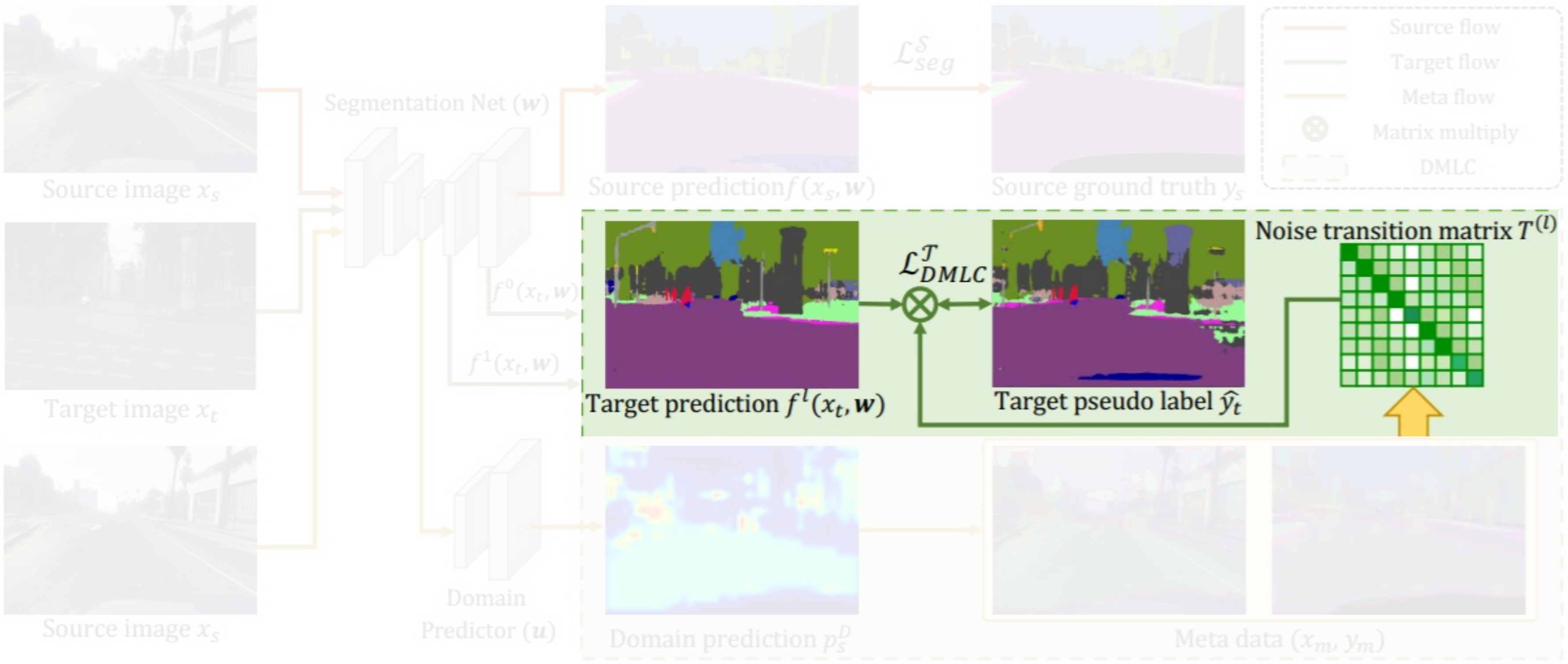
Method

- **MetaCorrection framework**
  - A learnable NTM formally models the noise distribution of pseudo labels in target domain.
  - DMLC strategy estimates NTM for loss correction in a data driven manner.
  - Provide matched and compatible supervision signals for different layers.



Method

- **Self-training with Loss Correction using NTM**
  - A learnable NTM formally models the noise distribution of pseudo labels in target domain.
  - DMLC strategy estimates NTM for loss correction in a data driven manner.
  - Provide matched and compatible supervision signals for different layers.



Method

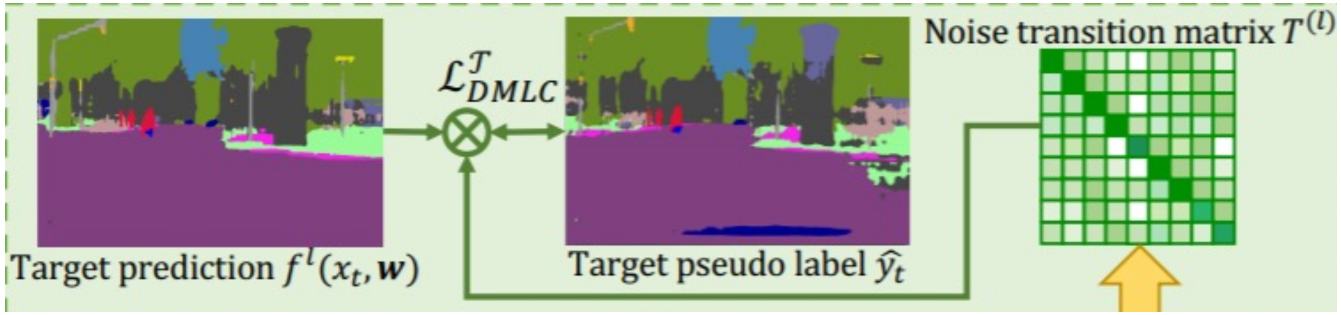
- **Self-training with Loss Correction using NTM**
  - NTM $T \in [0,1]^{C\times C}$: bridge pseudo labels $\widehat{Y}_t$ to the ground truth labels Y_t.
  - $T_{jk} = p(\hat{y}_t = k | y_t = j)$: the probability of ground truth label j flipping to noisy label k.

$$p(\widehat{y}_t = k \mid x_t, \mathbf{w}) = \sum_{j=1}^{C} T_{jk} p(y_t = j \mid x_t, \mathbf{w}),$$
$$\Rightarrow p(\widehat{y}_t \mid x_t, \mathbf{w}) = p(y_t \mid x_t, \mathbf{w})T.$$

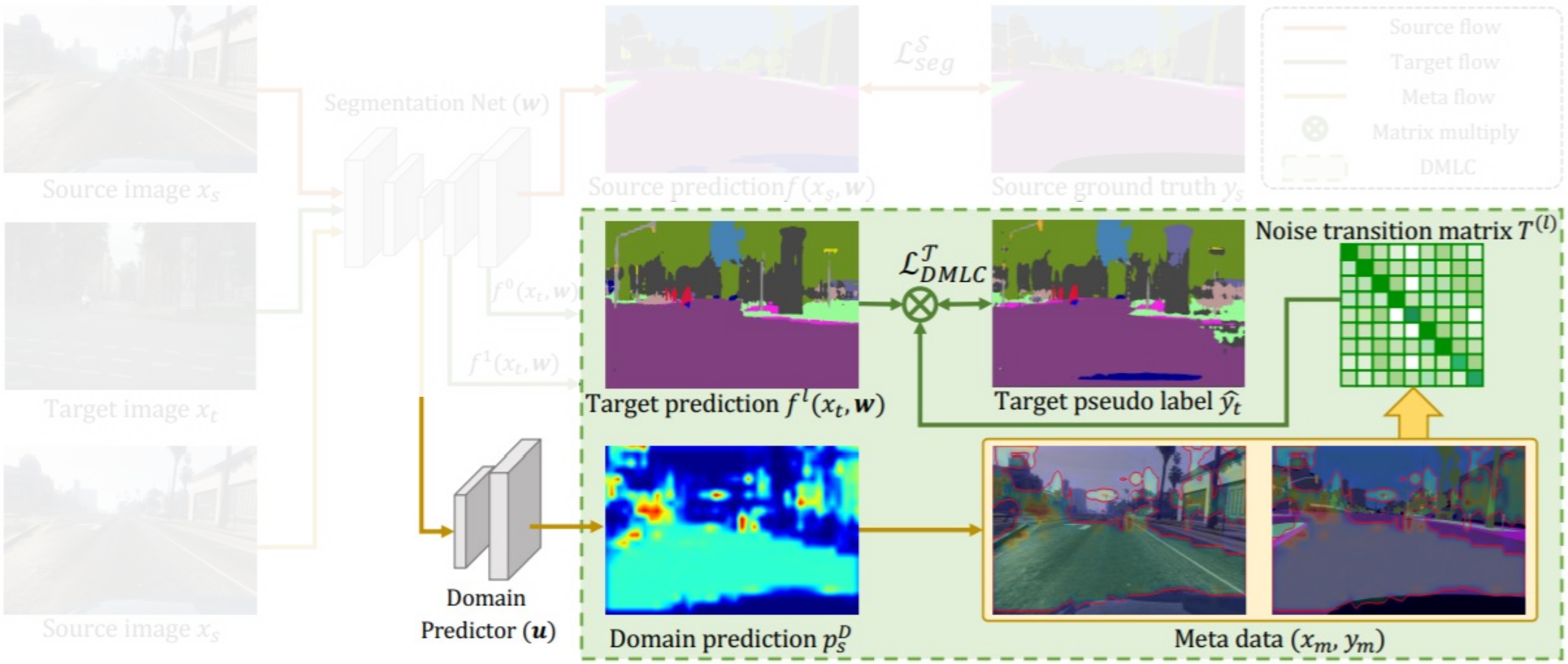
  - Correct supervision signal for target domain data via NTM

$$\begin{aligned}\mathcal{L}_{ST} &= \mathcal{L}^{\mathcal{S}}_{seg}(X_{\mathcal{S}}, Y_{\mathcal{S}}) + \mathcal{L}^{\mathcal{T}}_{seg}(X_{\mathcal{T}}, \widehat{Y}_{\mathcal{T}}) \\ &= -\sum_{s\in\mathcal{S}} y_s \log f(x_s, \mathbf{w}) \boxed{-\sum_{t\in\mathcal{T}} \widehat{y}_t \log f(x_t, \mathbf{w})}.\end{aligned}$$

$$\Rightarrow \quad \mathcal{L}^{\mathcal{T}}_{LC}(X_{\mathcal{T}}, \widehat{Y}_{\mathcal{T}}) = -\sum_{t\in\mathcal{T}} \widehat{y}_t \log[f(x_t, \mathbf{w})T].$$



- **Domain-aware Meta Loss Correction (DMLC)**
  - A learnable NTM formally models the noise distribution of pseudo labels in target domain.
  - DMLC strategy estimates NTM for loss correction in a data driven manner.
  - Provide matched and compatible supervision signals for different layers.



Method

Department of Electrical Engineering
香港城市大學
City University of Hong Kong

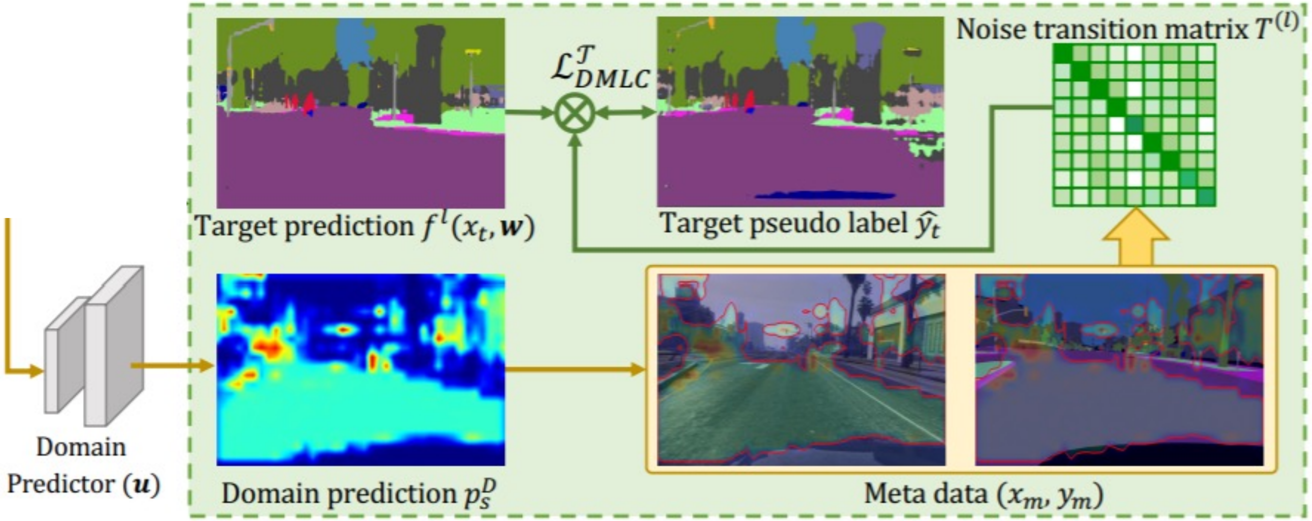
- **Domain-aware Meta Loss Correction (DMLC)**
  - Aim to heuristically explore the inter-class noise transition probabilities.
  - DMLC estimates T by minimizing the empirical risk on the domain-invariant meta data with clean labels
  - DMLC optimizes the segmentation net with loss corrected by T^* on the unlabeled target data.

$$T^* = \underset{T \in [0,1]^{c \times c}}{\arg\min} - \sum_{m \in \mathcal{M}} y_m \log f(x_m, \mathbf{w}(T)^*),$$

$$where\ \mathbf{w}(T)^* = \underset{\mathbf{w}}{\arg\min} - \sum_{t \in \mathcal{T}} \hat{y}_t \log[f(x_t, \mathbf{w})\, T],$$

$$\mathcal{L}_{DMLC} = \mathcal{L}_{seg}^{\mathcal{S}}(X_{\mathcal{S}}, Y_{\mathcal{S}}) + \mathcal{L}_{DMLC}^{\mathcal{T}}(X_{\mathcal{T}}, \widehat{Y}_{\mathcal{T}})$$

$$= - \sum_{s \in \mathcal{S}} y_s \log f(x_s, \mathbf{w}) - \sum_{t \in \mathcal{T}} \hat{y}_t \log[f(x_t, \mathbf{w})\, T^*].$$



Method

➢ Domain-aware Meta Loss Correction (DMLC)

- Virtual Optimization

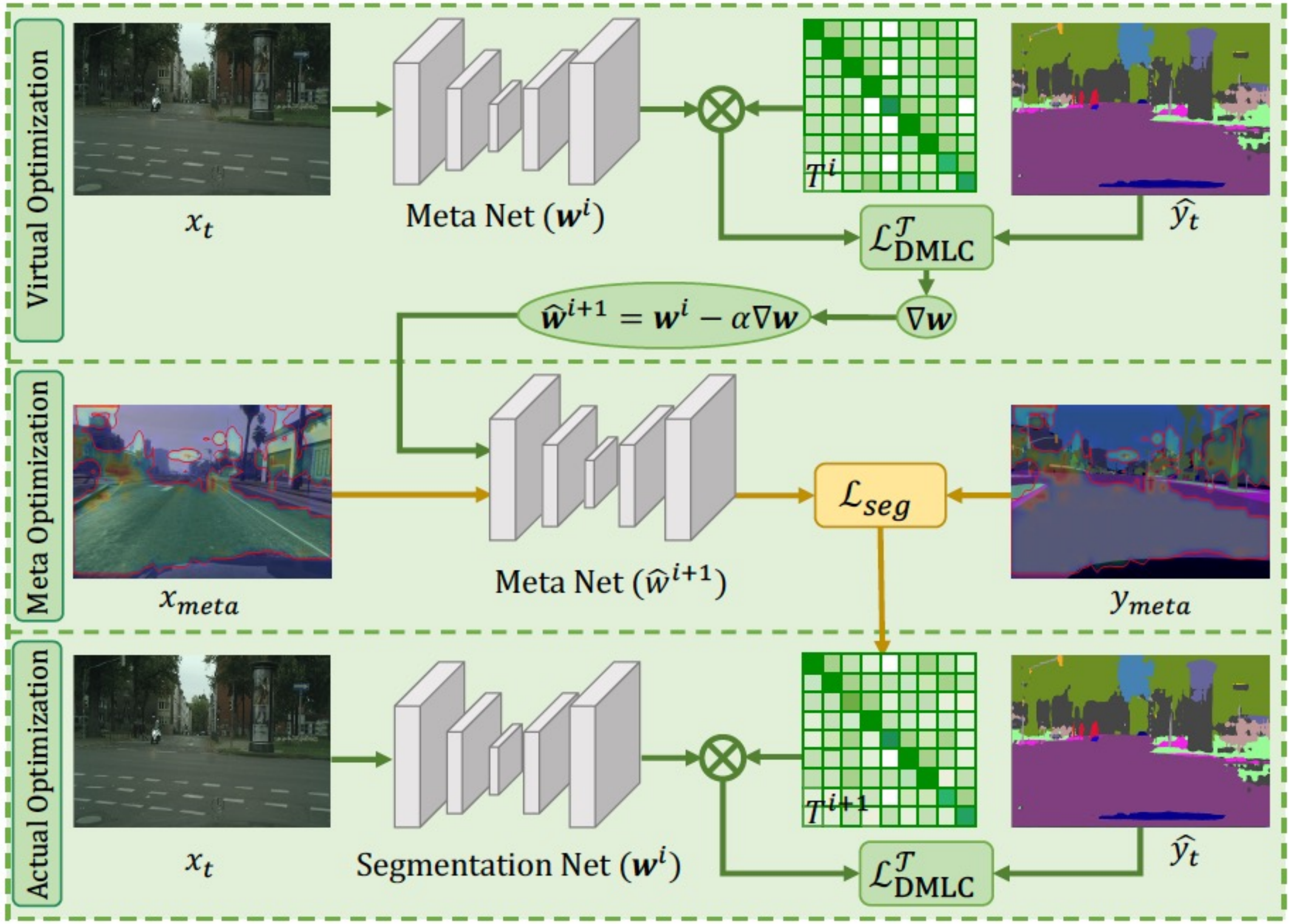
$$\hat{\mathbf{w}}^{i+1}(T^i) = \mathbf{w}^i + \gamma_v \nabla_{\mathbf{w}} \sum_{t\in\mathcal{T}} \hat{y}_t \log[f(x_t, \mathbf{w}^i)\, T^i]$$

- Meta Optimization

$$\widetilde{T}^{i+1} = T^i + \gamma_m \nabla_T \sum_{m\in\mathcal{M}} y_m \log f(x_m, \hat{\mathbf{w}}^{i+1}(T^i))$$

- Actual Optimization

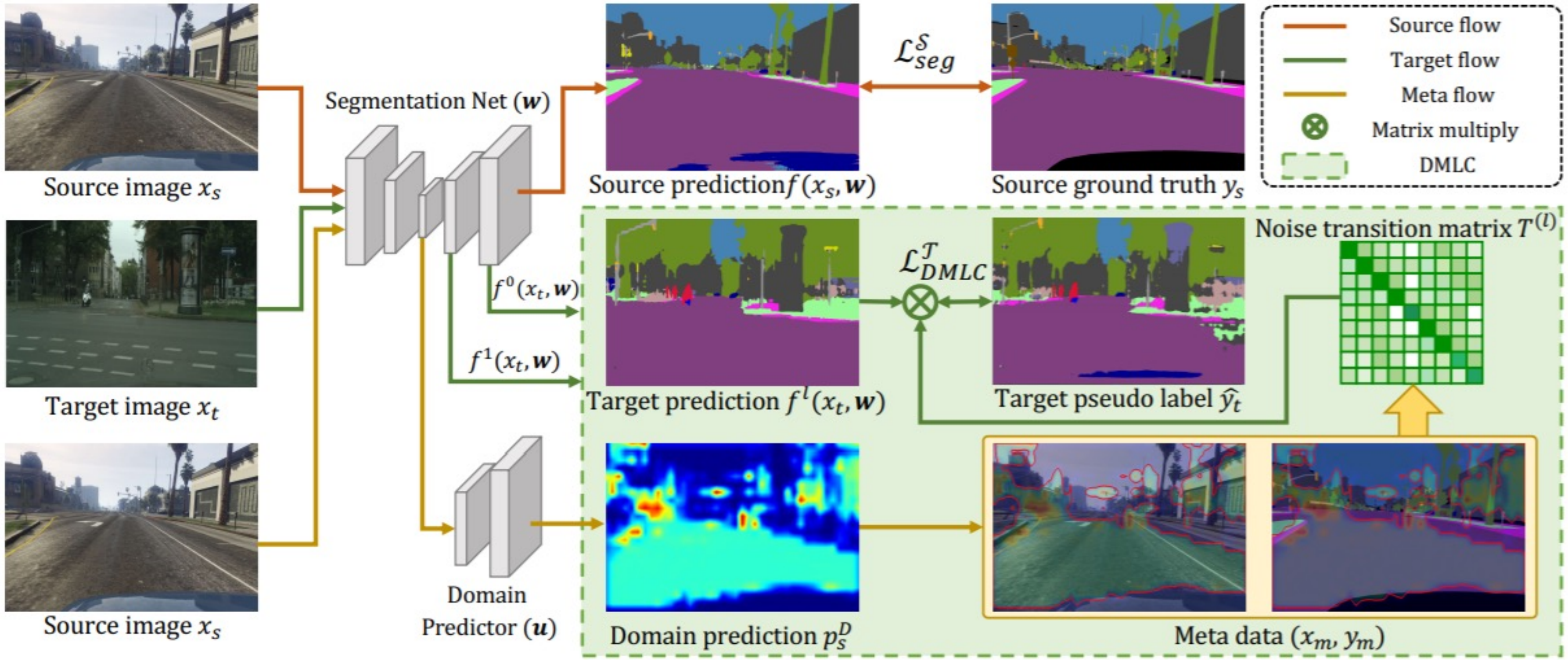
$$\mathbf{w}^{i+1} = \mathbf{w}^i + \gamma_a \nabla_{\mathbf{w}} \sum_{s\in\mathcal{S}} y_s \log f(x_s, \mathbf{w}) + \gamma_a \nabla_{\mathbf{w}} \sum_{t\in\mathcal{T}} \hat{y}_t \log[f(x_t, \mathbf{w})\, T^{i+1}]$$



Department of Electrical Engineering
香港城市大學
City University of Hong Kong

- **MetaCorrection framework**
  - A learnable NTM formally models the noise distribution of pseudo labels in target domain.
  - DMLC strategy estimates NTM for loss correction in a data driven manner.
  - Provide matched and compatible supervision signals for different layers.

$$\mathcal{L}_{MC} = \mathcal{L}^{S}_{seg}(X_S, Y_S) + \sum_{l} \alpha_l \mathcal{L}^{T(l)}_{DMLC}(X_T, \hat{Y}_T).$$



Experiments

- **Dataset of GTA5→CityScapes**
  - **GTA5**: 24,966 images captured from a <u>virtual video game</u>.
  - **CityScapes**: a real-world dataset collected in <u>driving scenarios</u>.

    Training set: 2,975 unlabeled images   Test set: 500 images

GTA5

CityScapes

Experiments

➢ Results on GTA5→CityScapes

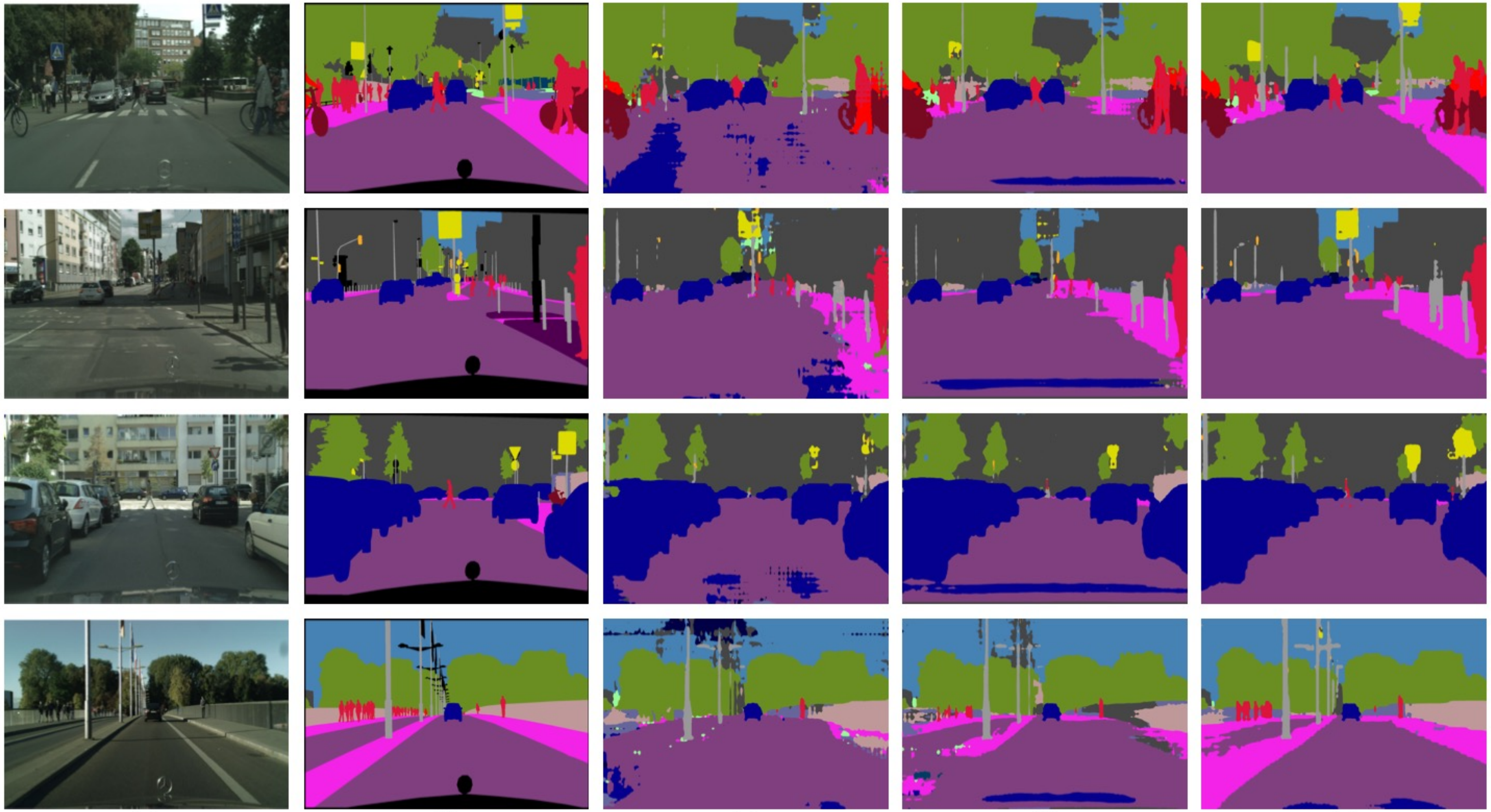
Table 1. Results of adapting GTA5 to CityScapes. The mechanism 'AL' and 'ST' stand for adversarial learning and self-training.

		GTA5 → CityScapes																			
Methods	mech.	road	sidewalk	building	wall	fence	pole	traffic lgt	traffic sgn	veg.	terrain	sky	person	rider	car	truck	bus	train	motor	bike	mIoU
AdaptSegNet [40]	AL	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
PatchAlign [41]	AL	92.3	51.9	82.1	29.2	25.1	24.5	33.8	33.0	82.4	32.8	82.2	58.6	27.2	84.3	33.4	46.3	2.2	29.5	32.3	46.5
LTIR [20]	AL	**92.9**	55.0	85.3	34.2	31.1	34.9	40.7	34.0	85.2	40.1	87.1	61.0	31.1	82.5	32.3	42.9	0.3	36.4	46.1	50.2
CBST [51]	ST	91.8	53.5	80.5	32.7	21.0	34.0	28.9	20.4	83.9	34.2	80.9	53.1	24.0	82.7	30.3	35.9	16.0	25.9	42.8	45.9
CRST [50]	ST	91.0	55.4	80.0	33.7	21.4	37.3	32.9	24.5	85.0	34.1	80.8	57.7	24.6	84.1	27.8	30.1	26.9	26.0	42.3	47.1
MaxSquare [4]	ST	89.4	43.0	82.1	30.5	21.3	30.3	34.7	24.0	85.3	39.4	78.2	**63.0**	22.9	84.6	36.4	43.0	5.5	34.7	33.5	46.4
MLSL [17]	ST	89.0	45.2	78.2	22.9	27.3	37.4	**46.1**	43.8	82.9	18.6	61.2	60.4	26.7	85.4	35.9	44.9	36.4	**37.2**	**49.3**	49.0
PyCDA [23]	ST	90.5	36.3	84.4	32.4	28.7	34.6	36.4	31.5	**86.8**	37.9	78.5	62.3	21.5	85.6	27.9	34.8	18.0	22.9	**49.3**	47.4
IntraDA [30]	ST	90.6	37.1	82.6	30.1	19.1	29.5	32.4	20.6	85.7	**40.5**	79.7	58.7	31.1	**86.3**	31.5	48.3	0.0	30.2	35.8	46.3
CAG-UDA [46]	ST	90.4	51.6	83.8	34.2	27.8	**38.4**	25.3	**48.4**	85.4	38.2	78.1	58.6	**34.6**	84.7	21.9	42.7	**41.1**	29.3	37.2	50.2
Source only	–	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
Ours (single DMLC)	ST	92.5	55.1	85.9	36.9	32.4	34.7	41.4	37.0	85.3	37.8	87.4	62.7	31.8	84.5	**36.8**	48.2	2.2	34.3	47.3	51.2
Ours (MetaCorrection)	ST	92.8	**58.1**	**86.2**	**39.7**	**33.1**	36.3	42.0	38.6	85.5	37.8	**87.6**	62.8	31.7	84.8	35.7	**50.3**	2.0	36.8	48.0	**52.1**

Experiments

Department of Electrical Engineering
香港城市大學
City University of Hong Kong

- Results on GTA5→CityScapes



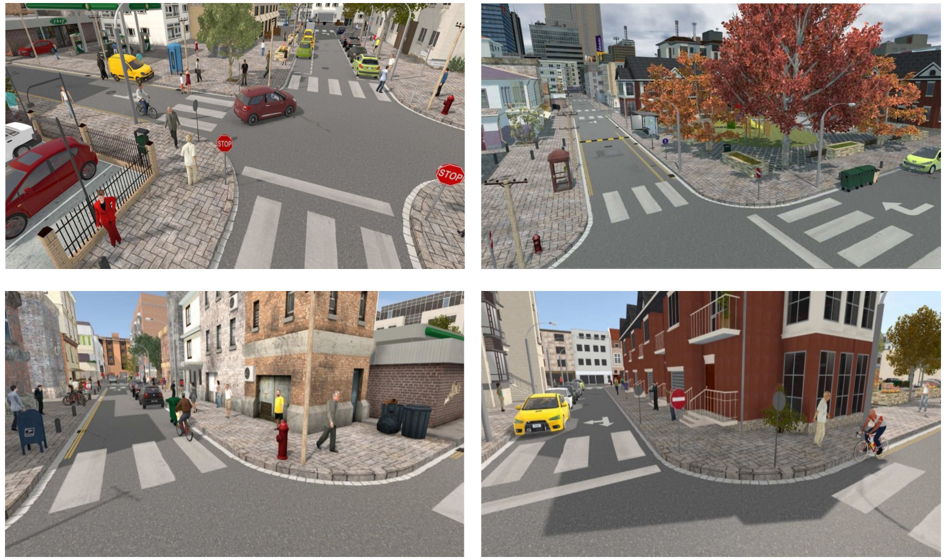
(a) (b) (c) (d) (e)

Figure 4. Qualitative results of UDA semantic segmentation in GTA5→CityScapes scenario. (a) Target image, (b) Ground truth, Predictions from (c) source only model, (d) self-training based MRENT model, (e) ours (MetaCorrection).

Experiments

- Dataset of SYNTHIA → CityScapes
  - **SYNTHIA**: 9,400 <u>synthetic</u> images.
  - **CityScapes**: a real-world dataset collected in <u>driving scenarios</u>.

    Training set: 2,975 unlabeled images   Test set: 500 images



SYNTHIA

CityScapes

Experiments

➢ Results on SYNTHIA → CityScapes

Table 2. Results of adapting SYNTHIA to CityScapes. mIoU* denotes the mean IoU of 13 classes, excluding the classes with ∗.

		SYNTHIA → CityScapes																	
Methods	mech.	road	sidewalk	building	wall*	fence*	pole*	light	sign	veg.	sky	person	rider	car	bus	mbike	bike	mIoU	mIoU*
AdaptSegNet [40]	AL	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
PatchAlign [41]	AL	82.4	38.0	78.6	8.7	0.6	26.0	3.9	11.1	75.5	84.6	53.5	21.6	71.4	32.6	19.3	31.7	40.0	46.5
LTIR [20]	AL	**92.6**	**53.2**	79.2	–	–	–	1.6	7.5	78.6	84.4	52.6	20.0	82.1	34.8	14.6	39.4	–	49.3
CBST [51]	ST	68.0	29.9	76.3	10.8	1.4	33.9	22.8	29.5	77.6	78.3	60.6	<u>28.3</u>	81.6	23.5	18.8	39.8	42.6	48.9
CRST [50]	ST	67.7	32.2	73.9	10.7	1.6	**37.4**	22.2	<u>31.2</u>	80.8	80.5	60.8	**29.1**	82.8	25.0	19.4	45.3	43.8	50.1
MaxSquare [4]	ST	82.9	40.7	<u>80.3</u>	10.2	0.8	25.8	12.8	18.2	82.5	82.2	53.1	18.0	79.0	31.4	10.4	35.6	41.4	48.2
MLSL [17]	ST	59.2	30.2	68.5	**22.9**	1.0	<u>36.2</u>	**32.7**	28.3	**86.2**	75.4	**68.6**	27.7	82.7	26.3	**24.3**	**52.7**	<u>45.2</u>	51.0
PyCDA [23]	ST	75.5	30.9	83.3	<u>20.8</u>	0.7	32.7	<u>27.3</u>	**33.5**	<u>84.7</u>	85.0	64.1	25.4	<u>85.0</u>	45.2	21.2	32.0	**46.7**	**53.3**
IntraDA [30]	ST	84.3	37.7	79.5	5.3	0.4	24.9	9.2	8.4	80.0	84.1	57.2	23.0	78.0	**38.1**	20.3	36.5	41.7	48.9
CAG-UDA [46]	ST	84.7	40.8	81.7	7.8	0.0	35.1	13.3	22.7	84.5	77.6	<u>64.2</u>	27.8	80.9	19.7	<u>22.7</u>	<u>48.3</u>	44.5	51.5
Source only	–	55.6	23.8	74.6	9.2	0.2	24.4	6.1	12.1	74.8	79.0	55.3	19.1	39.6	23.3	13.7	25.0	33.5	38.6
Ours (single DMLC)	ST	<u>92.3</u>	<u>53.0</u>	80.2	7.7	**2.8**	26.9	11.4	8.1	83.1	**85.2**	58.9	20.5	**85.5**	35.9	21.0	41.8	44.6	52.1
Ours (MetaCorrection)	ST	**92.6**	52.7	**81.3**	8.9	<u>2.4</u>	28.1	13.0	7.3	83.5	<u>85.0</u>	60.1	19.7	84.8	<u>37.2</u>	21.5	43.9	45.1	<u>52.5</u>

Experiments

➢ Results on SYNTHIA → CityScapes

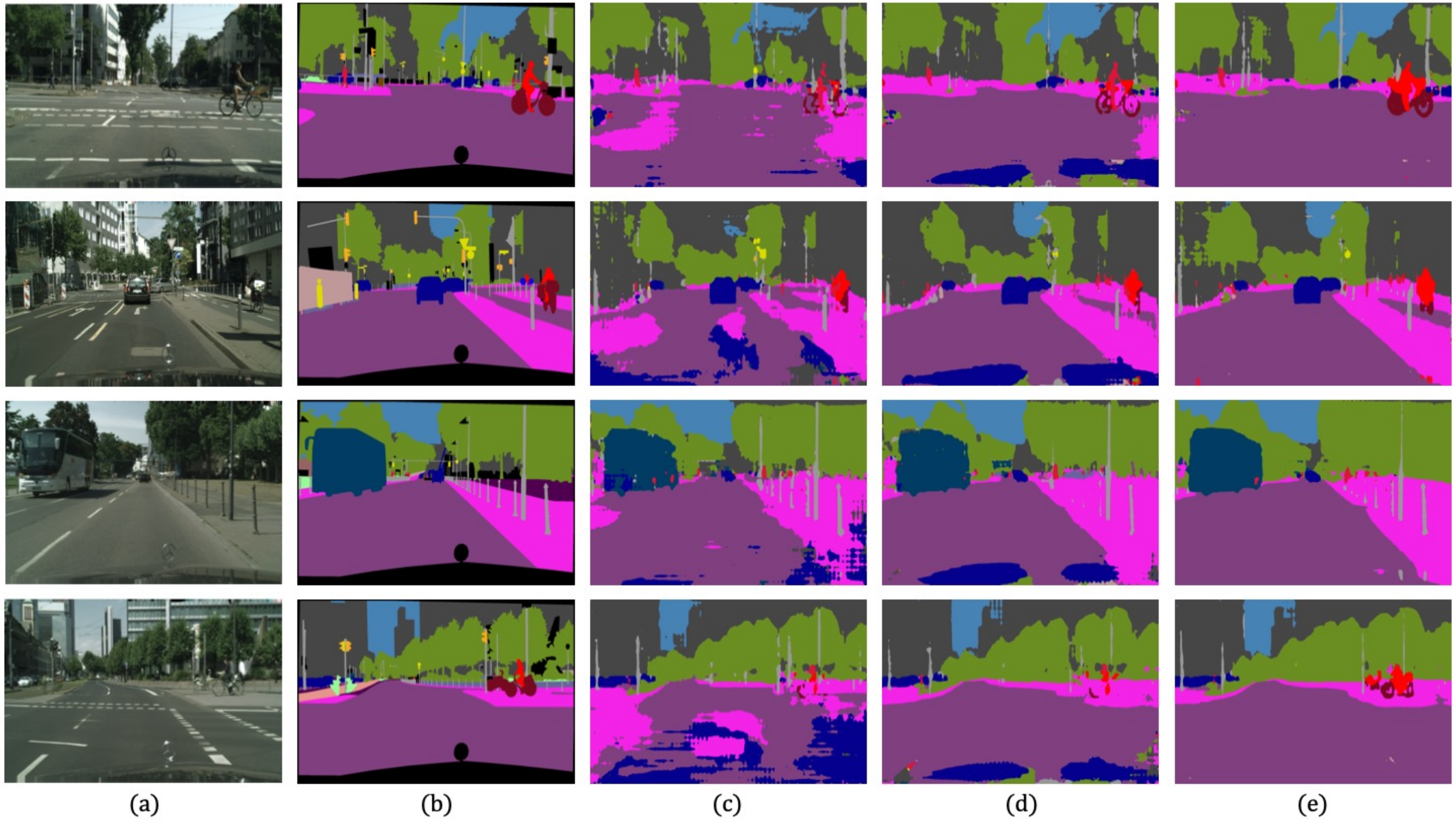
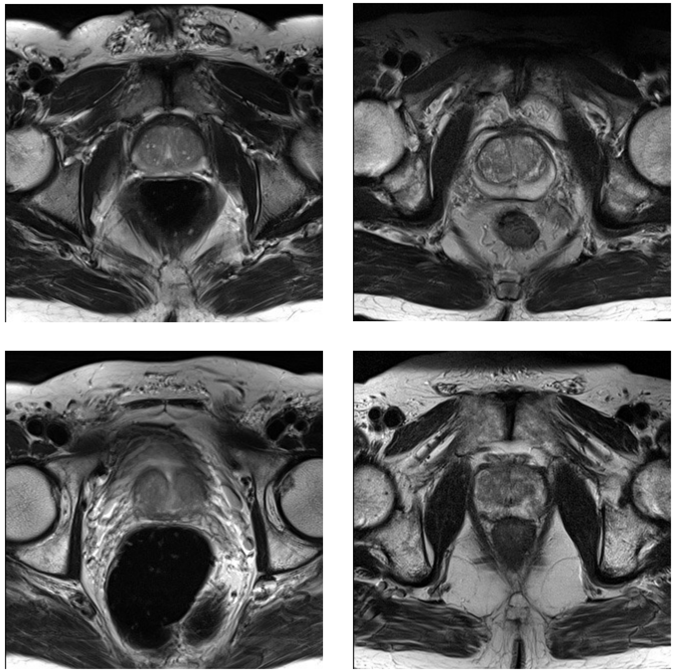


Figure 5. Qualitative results of UDA semantic segmentation in SYNTHIA→CityScapes scenario. (a) Target image, (b) Ground truth, Predictions from (c) source only model, (d) self-training based MRENT model, (e) ours (MetaCorrection).

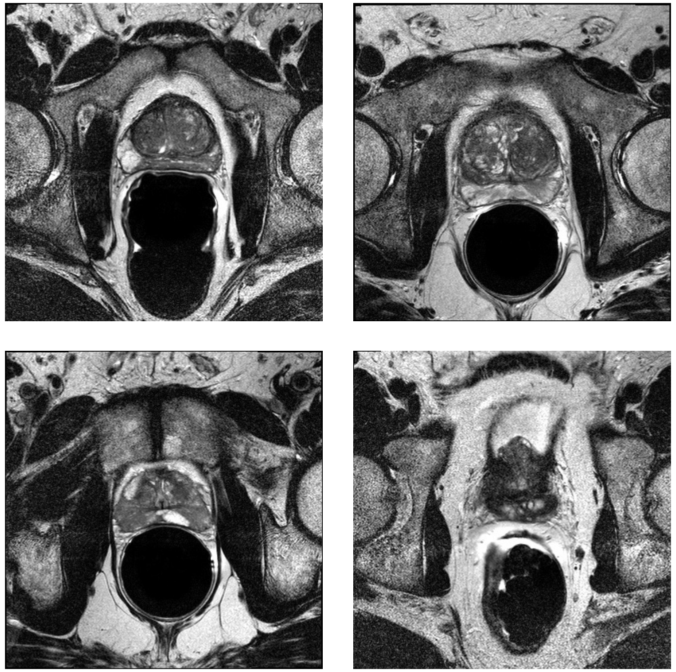
Experiments

- **Dataset of Decathlon → NCI-ISBI13**
  - **Decathlon**: 32 prostate MRIs obtained from 3T (Siemens TIM).
  - **NCI-ISBI13**: 40 prostate MRIs obtained from 1.5 T (Philips Achieva).

    Training set: 30 unlabeled 3D MRIs   Test set: 10 3D MRIs



Decathlon



NCI-ISBI13

Experiments

➢ **Results on Decathlon → NCI-ISBI13**

Table 3. Results of adapting Decathlon to NCI-ISBI13.

Method	mech.	PZ (*Dice*)	TZ (*Dice*)	WP (*Dice*)
CBST [51]	ST	38.22	70.14	64.31
MRENT [50]	ST	40.82	72.39	67.68
MaxSquare [4]	ST	37.45	69.61	63.34
Source only	–	28.48	52.57	47.56
Ours (single DMLC)	ST	42.03	74.09	69.38
Ours (MetaCorrection)	ST	**43.25**	**74.31**	**70.87**

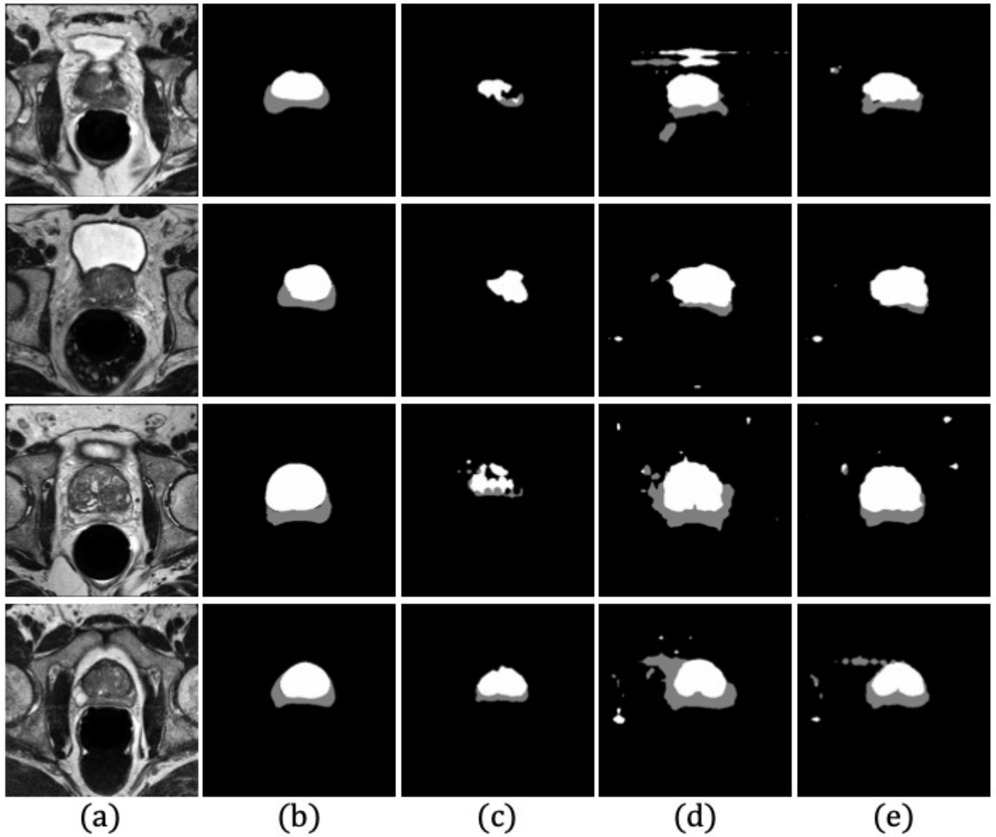


Figure 3. Qualitative results of UDA semantic segmentation in Decathlon→NCI-ISBI13 scenario. (a) Target image, (b) Ground truth, Predictions from (c) source only model, (d) self-training based MRENT model, (e) ours (MetaCorrection).

Experiments

- **Ablation Study**
  - Comparison with Self-training based UDA Models.
  - Robustness to Various Types of Noise.

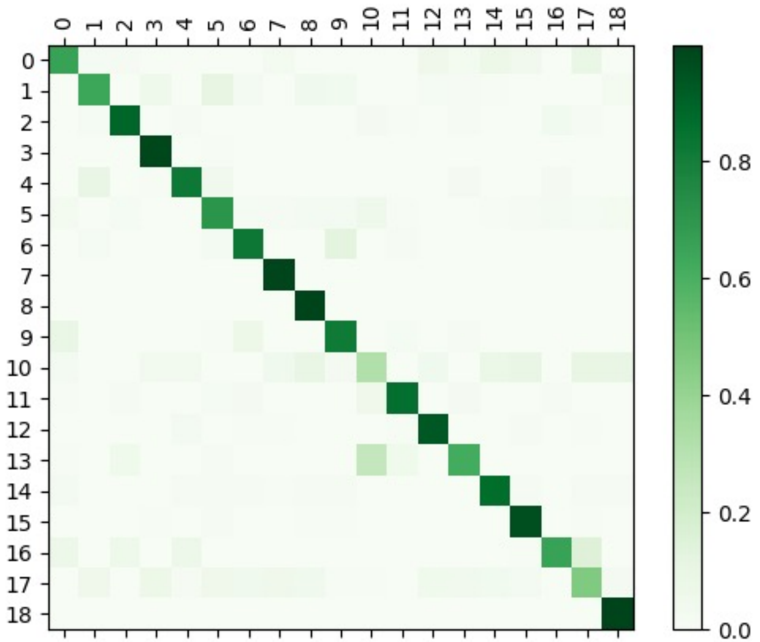
Table 4. Impact of different pseudo labels. 'Pseudo Label' denotes we employ pseudo labels generated by the corresponding model.

Method	Pseudo Label	GTA5 → CityScapes	Δ
AdaptSegNet [40]	—	42.4	–
Self-training (MRENT [50])	AdaptSegNet	45.1	2.7
Self-training (Threshold [51])	AdaptSegNet	44.4	2.0
Self-training (Ucertainty [48])	AdaptSegNet	46.1	3.7
Ours (single DMLC)	AdaptSegNet	45.9	3.5
Ours (MetaCorrection)	AdaptSegNet	**47.3**	**4.9**
LTIR [20]	—	50.2	–
Self-training (MRENT [50])	LTIR	50.6	0.4
Ours (single DMLC)	LTIR	51.2	1.0
Ours (MetaCorrection)	LTIR	**52.1**	**1.9**
Source only	—	36.6	–
Self-training (MRENT [50])	Source	39.6	3.0
Ours (single DMLC)	Source	43.8	7.2
Ours (MetaCorrection)	Source	**44.5**	**7.9**

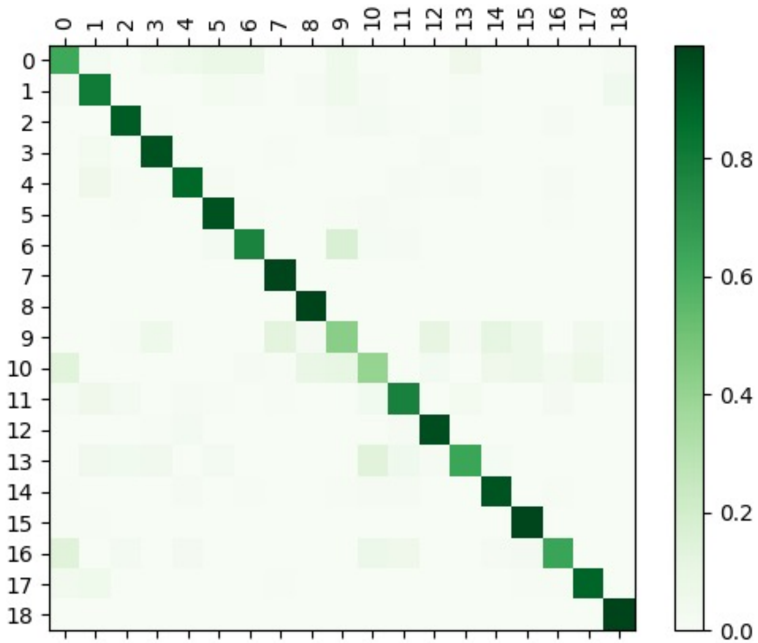
Experiments

➢ Visualization Results

- NTM Visualization: different level learns different noise transition matrix.



Noise Transition Matrix (T^0)

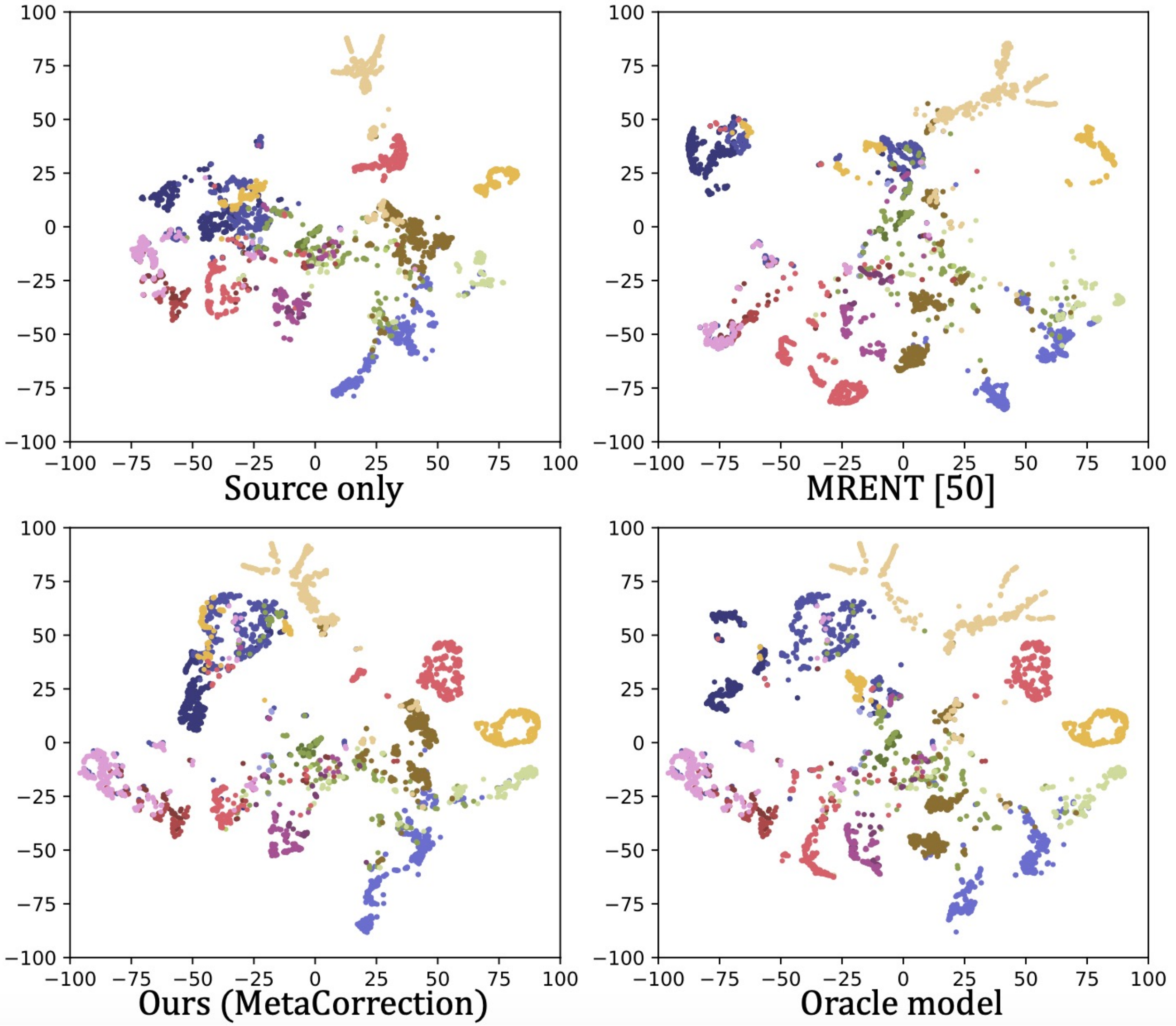


Noise Transition Matrix (T^1)

Experiments

➢ Visualization Results

- Feature Visualization: our feature distribution is more similar to the oracle method.



Conclusion

- We advance a **MetaCorrection** framework, where a Domain-aware Meta-learning strategy is devised to benefit Loss Correction (DMLC) for UDA semantic segmentation.
  - NTM: models the noise distribution of pseudo labels in target domain.
  - DMLC strategy: estimates NTM for loss correction in a data driven manner.
  - To accommodate the capacity gap between shallow and deep features, DMLC strategy is further incorporated to provide compatible supervision signals for low-level features.

Thanks!