Domain Adaptation in Semantic Segmentation

李鑫

2021.05.12

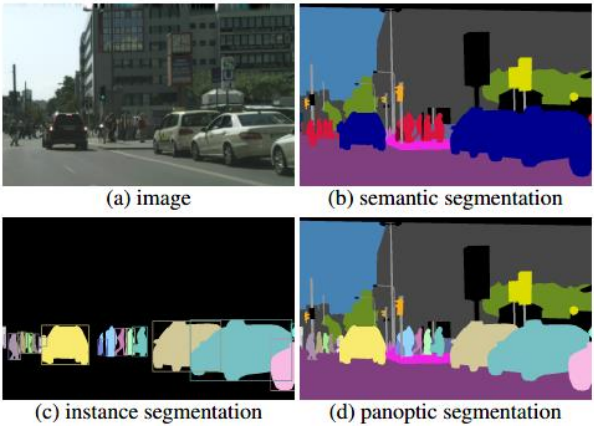
Outline

- Introduction
- Adversarial training based methods
- Self-training based methods
- Multi-source methods

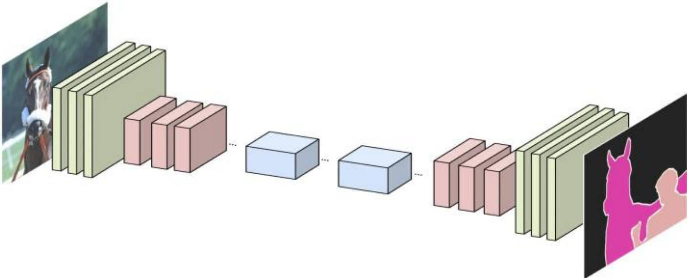
Outline

- Introduction
- Adversarial training based methods
- Self-training based methods
- Multi-source methods

Image Segmentation

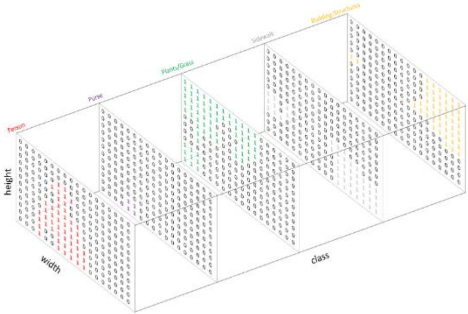


Semantic Segmentation

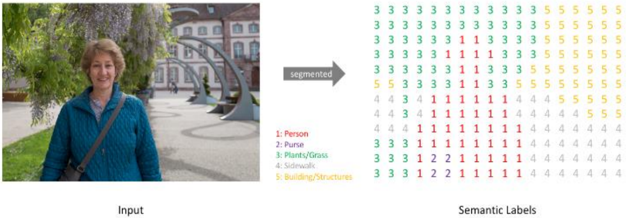


Base Framework

Semantic Segmentation



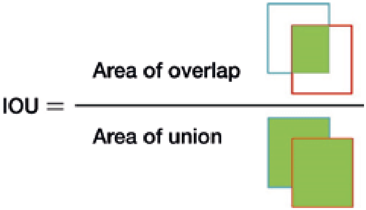
Prediction



Metrics for Segmentation Models

Intersection over Union (IoU):

$$\mathrm{IoU} = J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

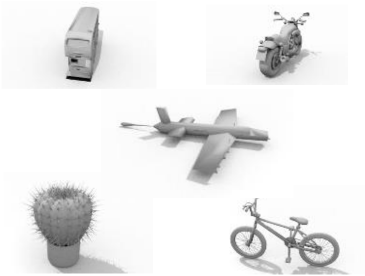


- Also called Jaccard Index
- The most commonly used metrics in semantic segmentation. (mean-IoU/mIoU)
- A denotes the ground truth and B denotes the predicted segmentation maps.

Cross-Domain Prediction

- **The distribution of test data is different that of training data**

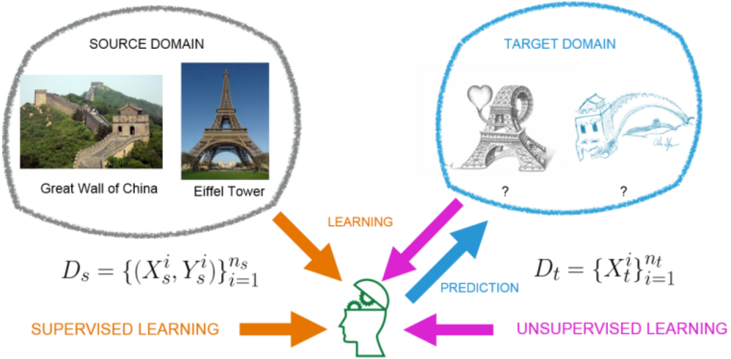
  Style, layout, shape, context, illumination, etc.



Training data

Test data

Domain adaptation (DA)



$$D_s = \{(X_s^i, Y_s^i)\}_{i=1}^{n_s}$$

$$D_t = \{X_t^i\}_{i=1}^{n_t}$$

Leveraging labeled **source domain**, to learn a model for the **target domain**.

Domain adaptation (DA)

The setting of domain adaptation

- Target distribution is different from the source one
- Same task (shared label sets)
- Large amounts of labeled source data and unlabeled target data

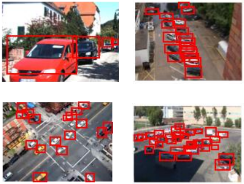
Why do we need domain adaptation?

- Costly to label large amounts of in-domain data
- Unrealistic to collect and annotate a dataset covering all the domain variations
- The knowledge of the task can be potentially reuse/shared across domains

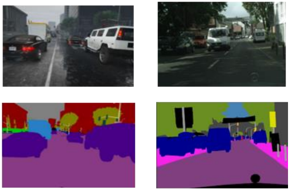
Example scenarios

Recognition

Detection

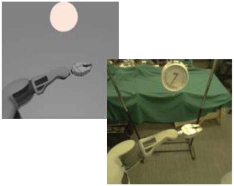


Segmentation



Re-identification

Control



Visual localization

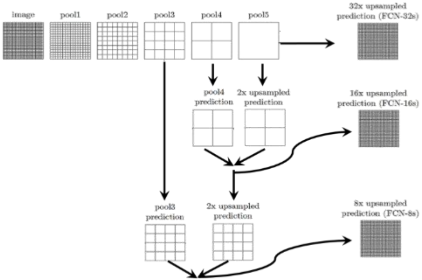
Fully Convolutional Networks (FCN)

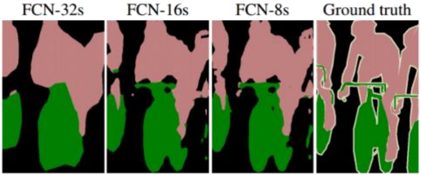
- Fully Convolutional
- Deconvolution
- Skip connections

  Combine coarse, high-level information and fine, low-level information

Limitations:

- Not fast enough for real-time inference.
- Does not take into account the global context information efficiently.
- Not easily transferable to 3D images.



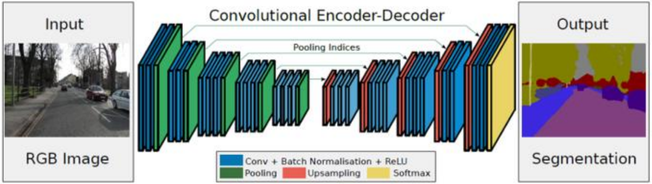


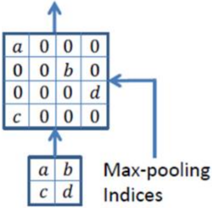
Jonathan Long, et al. Fully Convolutional Networks for Semantic Segmentation CVPR 2015

Encoder-Decoder-Based Models

SegNet

Encoder: 13 convolutional layers (VGG16) + 3 fully convolutional layers.
Eliminates the need for learning to up-sample.

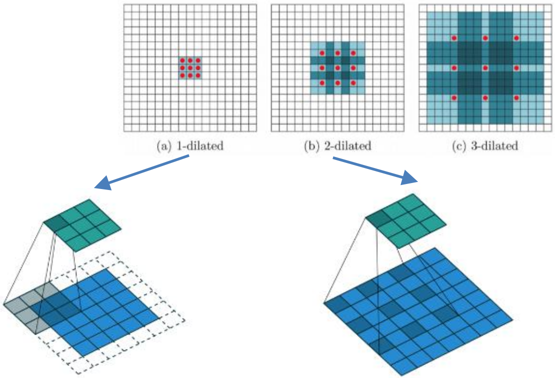




Vijay Badrinarayanan, et al; SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation PAMI 2016

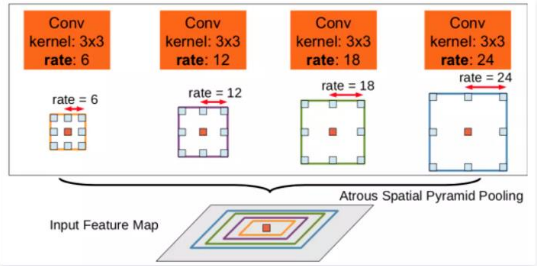
Dilated Convolutional Models and DeepLab Family

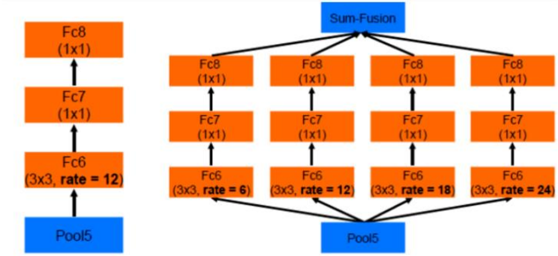
- Also called "Atrous Convolution".
- Enlarging the receptive field with no increase in computational cost.
- The DeepLab family, densely connected Atrous Spatial Pyramid Pooling (ASPP), **DeepLabv3+**：high mIoU of 89.0% on PASCAL VOC 2012.



Dilated Convolutional Models and DeepLab Family

ASPP (Atrous Spatial Pyramid Pooling)

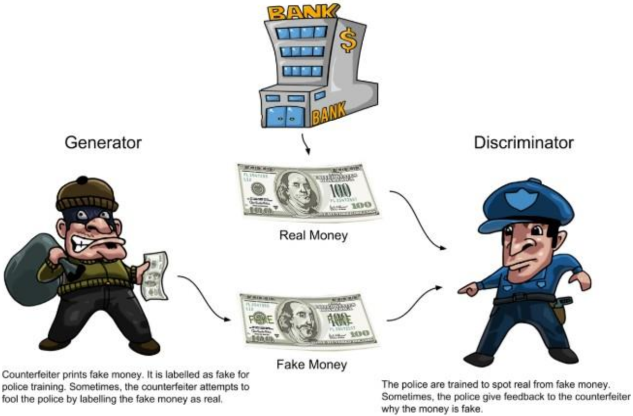




(a) DeepLab-LargeFOV (b) DeepLab-ASPP

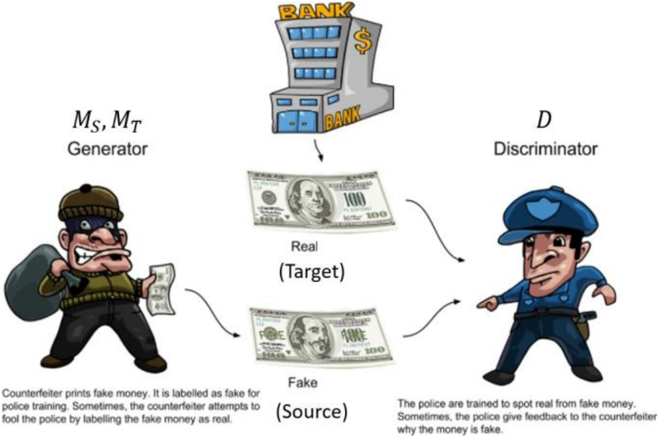
Adversarial learning

Principles of GAN

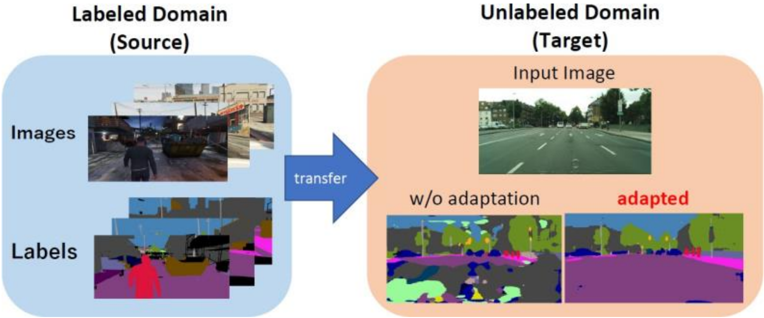


Goodfellow, et al. “Generative adversarial nets” NeurIPS 2014.

Domain adversarial training



Domain Adaptive Semantic Segmentation



From Synthetic to real data

- Easy to obtain pixel level annotation
- Poor labeling due to domain shift

Datasets

Synthetic

GTA5

GTA5 contains 24,966 training images with the resolution of 1914x1052 and we use its 19 categories shared with Cityscapes.

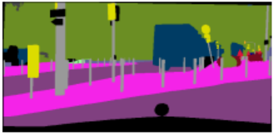
SYNTHIA

SYNTHIA dataset contains 9,400 1280x760 images and we use its 16 common categories with Cityscapes.

Real

Cityscape

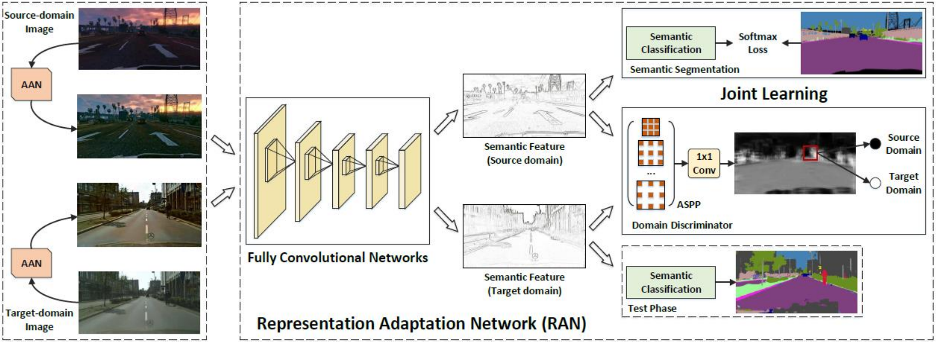
Cityscapes dataset contains 2,975 training images and 500 images for validation with the resolution of 2048x1024.



Outline

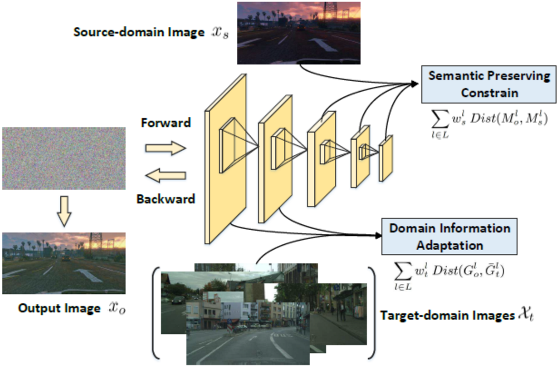
- Introduction
- Adversarial training based methods
- Self-training based methods
- Multi-source methods

Adversarial training based methods



Yiheng Zhang, et al. Fully Convolutional Adaptation Networks for Semantic Segmentation CVPR'18

Adversarial training based methods



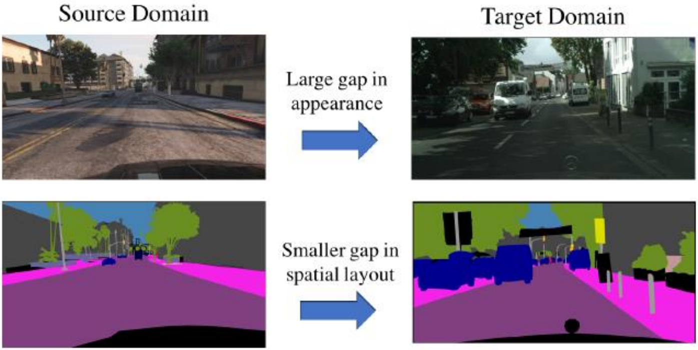
$$\min_{x_o} \sum_{l \in L} w_s^l \, Dist(M_o^l, M_s^l)$$

$$\min_{x_o} \sum_{l \in L} w_t^l \, Dist(G_o^l, \bar{G}_t^l)$$

$$\mathcal{L}_{AAN}(x_o) = \sum_{l \in L} w_s^l \, Dist(M_o^l, M_s^l) + \alpha \sum_{l \in L} w_t^l \, Dist(G_o^l, \bar{G}_t^l)$$

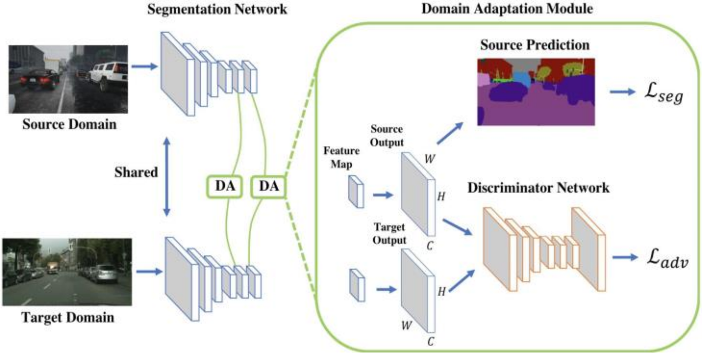
Yiheng Zhang, et al. Fully Convolutional Adaptation Networks for Semantic Segmentation CVPR 2018

Adversarial training based methods



Tsai, Yi-Hsuan, et al. "Learning to adapt structured output space for semantic segmentation." CVPR. 2018.

Adversarial training based methods

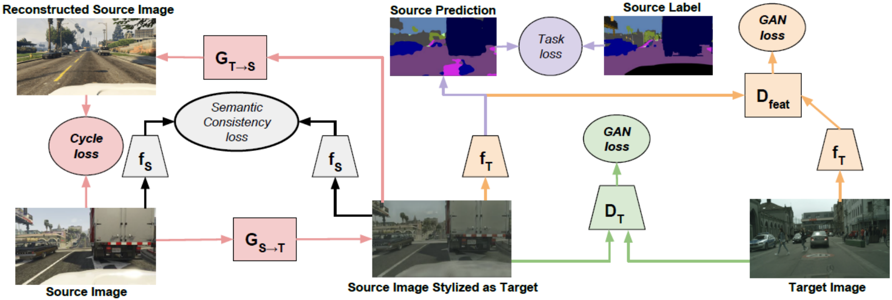


$$\mathcal{L}_d(P) = -\sum_{h,w}(1-z)\log(\mathbf{D}(P)^{(h,w,0)}) + z\log(\mathbf{D}(P)^{(h,w,1)}),$$

$$\mathcal{L}_{seg}(I_s) = -\sum_{h,w}\sum_{c\in C} Y_s^{(h,w,c)}\log(P_s^{(h,w,c)})$$

Tsai, Yi-Hsuan, et al. "Learning to adapt structured output space for semantic segmentation." CVPR. 2018.

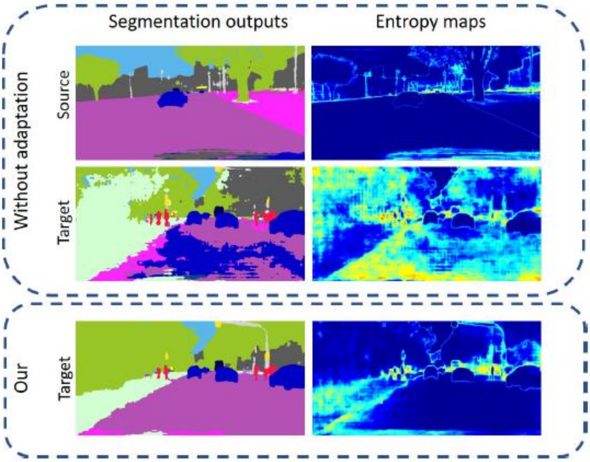
Adversarial training based methods



- image-level GAN loss (green)
- feature level GAN loss (orange)
- source and target semantic consistency losses(black)
- source cycle loss (red)
- source task loss (purple)

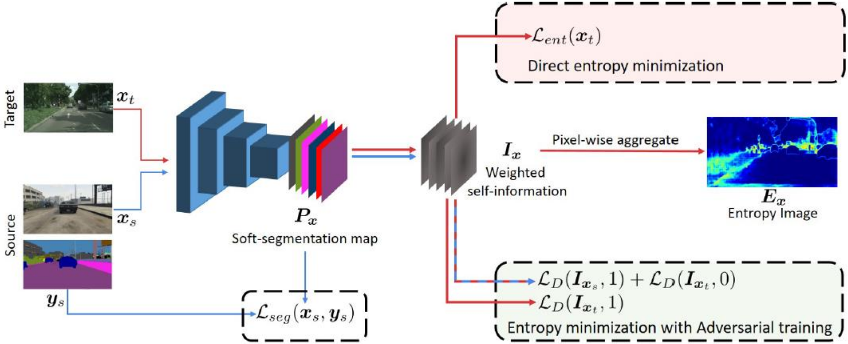
,Hoffman et al. "CyCADA: Cycle-Consistent Adversarial Domain Adaptation" ICML. 2018.

Adversarial training based methods



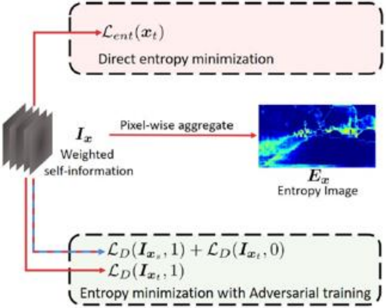
Tuan-Hung Vu, et al. ADVENT: Adversarial Entropy Minimization for Domain Adaptation in Semantic Segmentation, CVPR 2019

Adversarial training based methods



Tuan-Hung Vu, et al. ADVENT: Adversarial Entropy Minimization for Domain Adaptation in Semantic Segmentation, CVPR 2019

Adversarial training based methods



$$\boldsymbol{E}_{\boldsymbol{x}_t}^{(h,w)} = \frac{-1}{\log(C)} \sum_{c=1}^{C} \boldsymbol{P}_{\boldsymbol{x}_t}^{(h,w,c)} \log \boldsymbol{P}_{\boldsymbol{x}_t}^{(h,w,c)}$$

$$\mathcal{L}_{ent}(\boldsymbol{x}_t) = \sum_{h,w} \boldsymbol{E}_{\boldsymbol{x}_t}^{(h,w)}$$

$$\min_{\theta_F} \frac{1}{|\mathcal{X}_s|} \sum_{\boldsymbol{x}_s} \mathcal{L}_{seg}(\boldsymbol{x}_s, \boldsymbol{y}_s) + \frac{\lambda_{ent}}{|\mathcal{X}_t|} \sum_{\boldsymbol{x}_t} \mathcal{L}_{ent}(\boldsymbol{x}_t)$$

$$\min_{\theta_D} \frac{1}{|\mathcal{X}_s|} \sum_{\boldsymbol{x}_s} \mathcal{L}_D(\boldsymbol{I}_{\boldsymbol{x}_s}, 1) + \frac{1}{|\mathcal{X}_t|} \sum_{\boldsymbol{x}_t} \mathcal{L}_D(\boldsymbol{I}_{\boldsymbol{x}_t}, 0)$$

$$\min_{\theta_F} \frac{1}{|\mathcal{X}_t|} \sum_{\boldsymbol{x}_t} \mathcal{L}_D(\boldsymbol{I}_{\boldsymbol{x}_t}, 1)$$

$$\min_{\theta_F} \frac{1}{|\mathcal{X}_s|} \sum_{\boldsymbol{x}_s} \mathcal{L}_{seg}(\boldsymbol{x}_s, \boldsymbol{y}_s) + \frac{\lambda_{adv}}{|\mathcal{X}_t|} \sum_{\boldsymbol{x}_t} \mathcal{L}_D(\boldsymbol{I}_{\boldsymbol{x}_t}, 1)$$

Tuan-Hung Vu, et al. ADVENT: Adversarial Entropy Minimization for Domain Adaptation in Semantic Segmentation, CVPR 2019

Adversarial training based methods

(b) SYNTHIA → Cityscapes

Models	Appr.	road	sidewalk	building	wall	fence	pole	light	sign	veg	sky	person	rider	car	bus	mbike	bike	mIoU	mIoU*
FCNs in the Wild [15]	Adv	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
Adapt-SegMap [41]	Adv	78.9	29.2	75.5	-	-	-	0.1	4.8	72.6	76.7	43.4	8.8	71.1	16.0	3.6	8.4	-	37.6
Self-Training [51]	ST	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
Self-Training + CB [51]	ST	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
Ours (MinEnt)	Ent	37.8	18.2	65.8	2.0	0.0	15.5	0.0	0.0	76	73.9	45.7	11.3	66.6	13.3	1.5	13.1	27.5	32.5
Ours (MinEnt + CP)	Ent	45.9	19.6	65.8	5.3	0.2	20.7	2.1	8.2	74.4	76.7	47.5	12.2	71.1	22.8	4.5	9.2	30.4	35.4
Ours (AdvEnt + CP)	Adv	67.9	29.4	71.9	6.3	0.3	19.9	0.6	2.6	74.9	74.9	35.4	9.6	67.8	21.4	4.1	15.5	31.4	36.6
Adapt-SegMap [41]	Adv	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
Adapt-SegMap* [41]	Adv	81.7	39.1	78.4	11.1	0.3	25.8	6.8	9.0	79.1	80.8	54.8	21.0	66.8	34.7	13.8	29.9	39.6	45.8
Ours (MinEnt)	Ent	73.5	29.2	77.1	7.7	0.2	**27.0**	7.1	11.4	76.7	82.1	57.2	21.3	69.4	29.2	12.9	27.9	38.1	44.2
Ours (AdvEnt)	Adv	**87.0**	**44.1**	**79.7**	9.6	**0.6**	24.3	4.8	7.2	80.1	83.6	56.4	23.7	72.7	32.6	12.8	**33.7**	40.8	47.6
Ours (AdvEnt+MinEnt)	A+E	85.6	42.2	79.7	8.7	0.4	25.9	5.4	8.1	80.4	**84.1**	**57.9**	**23.8**	**73.3**	**36.4**	14.2	33.0	**41.2**	**48.0**

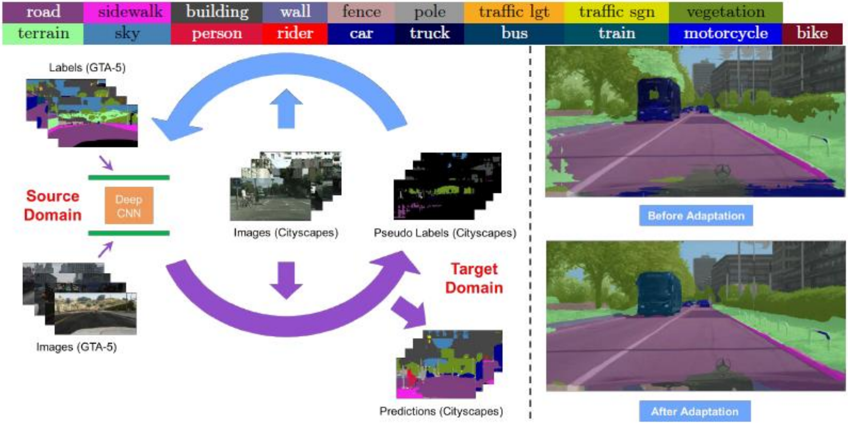
Tuan-Hung Vu, et al. ADVENT: Adversarial Entropy Minimization for Domain Adaptation in Semantic Segmentation, CVPR 2019

Outline

- Introduction
- Adversarial training based methods
- Self-training based methods
- Multi-source methods

Self-training learning

CBST: class-balanced self-training



Yang Zou et al, Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-Training ECCV 2018

Self-training learning based methods

Strategy1 ： self-paced self training learning

easy-to-hard

$$
\begin{aligned}
\min_{\mathbf{w},\hat{\mathbf{y}}} \mathcal{L}_{ST}(\mathbf{w},\hat{\mathbf{y}}) = & -\sum_{s=1}^{S}\sum_{n=1}^{N} \mathbf{y}_{s,n}^{\top} \log(\mathbf{p}_n(\mathbf{w},\mathbf{I}_s)) \\
& -\sum_{t=1}^{T}\sum_{n=1}^{N} [\hat{\mathbf{y}}_{t,n}^{\top} \log(\mathbf{p}_n(\mathbf{w},\mathbf{I}_t)) + \boxed{k|\hat{\mathbf{y}}_{t,n}|_1]} \\
& s.t.\ \hat{\mathbf{y}}_{t,n} \in \{\{\mathbf{e}^{(i)}|\mathbf{e}^{(i)} \in \mathbb{R}^C\} \cup \mathbf{0}\}, \forall t,n \\
& \quad k > 0
\end{aligned}
$$

Algorithm 1: Determination of k in ST

Input : Neural network $P(\mathbf{w})$, all target images $\mathbf{I}_t$, portion p of selected pseudo-labels

Output: k

for t=1 *to* T **do**

  $\mathrm{P}_{\mathbf{I}_t} = \mathrm{P}(\mathbf{w},\mathbf{I}_t)$

  $\mathrm{MP}_{\mathbf{I}_t} = \max(\mathrm{P}_{\mathbf{I}_t}, \mathrm{axis}=0)$

  $\mathrm{M} = [\mathrm{M}, \mathrm{matrix\_to\_vector}(\mathrm{MP}_{\mathbf{I}_t})]$

end

M = sort(M,order=descending)

$\mathrm{len}_{th} = \mathrm{length}(\mathrm{M}) \times \mathrm{p}$

$\mathrm{k} = -\log(\mathrm{M}[\mathrm{len}_{th}])$

return k

Yang Zou et al, Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-Training ECCV 2018

Self-training learning based methods

Strategy2： Class-Balanced Self-Training

$$
\begin{aligned}
\min_{\mathbf{w},\hat{\mathbf{y}}} \mathcal{L}_{CB}(\mathbf{w},\hat{\mathbf{y}}) = & -\sum_{s=1}^{S}\sum_{n=1}^{N} \mathbf{y}_{s,n}^{\top} \log(\mathbf{p}_n(\mathbf{w},\mathbf{I}_s)) \\
& -\sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{c=1}^{C} \left[\hat{y}_{t,n}^{(c)} \log(p_n(c|\mathbf{w},\mathbf{I}_t)) + \boxed{k_c \hat{y}_{t,n}^{(c)}}\right] \\
s.t.\ \hat{\mathbf{y}}_{t,n} = & \left[\hat{y}_{t,n}^{(1)},...,\hat{y}_{t,n}^{(C)}\right] \in \{\{\mathbf{e}^{(i)}|\mathbf{e}^{(i)} \in \mathbb{R}^C\} \cup \mathbf{0}\}, \forall t,n \\
& k_c > 0, \forall c
\end{aligned}
$$

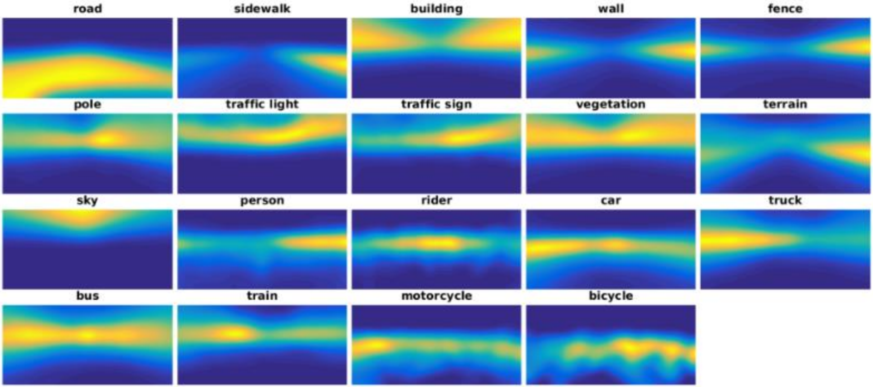
Algorithm 2: Determination of k_c in CBST

```
Input  : Neural network f(w), all target images I_t, portion p of selected
         pseudo-labels
Output: k_c
for t=1 to T do
    P_{I_t} = P(w,I_t)
    LP_{I_t} = argmax(P,axis=0)
    MP_{I_t} = max(P,axis=0)
    for c=1 to C do
        MP_{c,I_t} = MP_{I_t}(LP_{I_t} == c)
        M_c = [M_c, matrix_to_vector(MP_{c,I_t})]
    end
end
for c=1 to C do
    M_c = sort(M_c,order=descending)
    len_{c,th} = length(M_c) × p
    k_c = -log(M_c[len_{c,th}])
end
return k_c
```

Yang Zou et al, Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-Training ECCV 2018

Self-training learning based methods

Spatial Priors



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

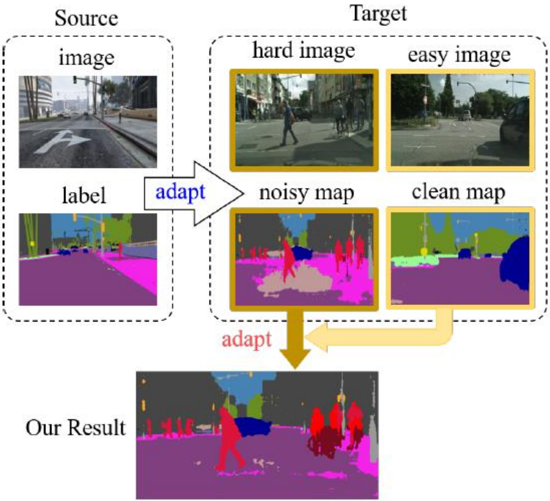
$$-\sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{c=1}^{C} [\hat{y}_{t,n}^{(c)} \log(q_n(c) p_n(c|\mathbf{w},\mathbf{I}_t)) + k_c \hat{y}_{t,n}^{(c)}]$$

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

$$k_c > 0, \forall c$$

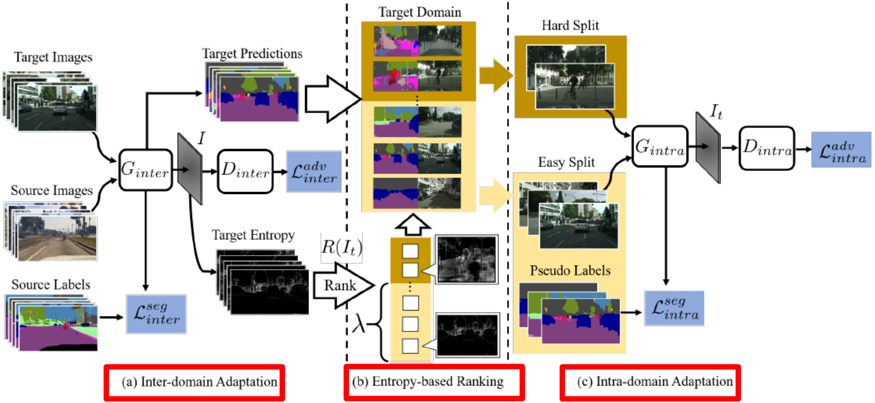
Yang Zou et al, Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-Training ECCV 2018

Self-training based method



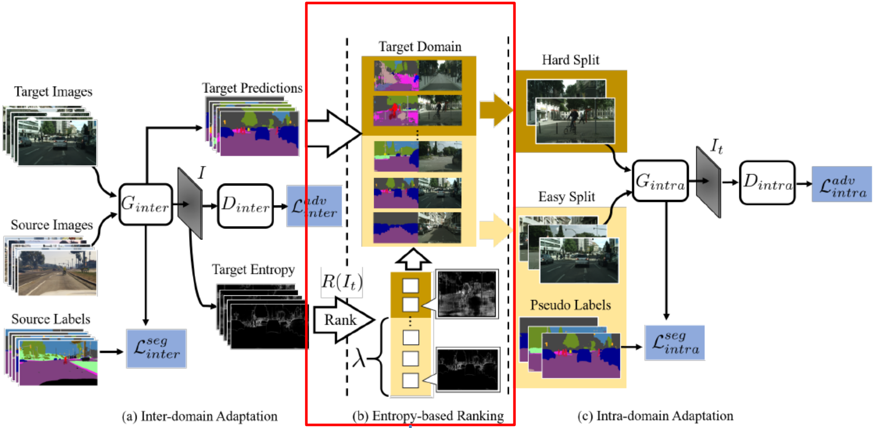
Fei Pan, et al. Unsupervised Intra-domain Adaptation for Semantic Segmentation through Self-Supervision CVPR 2020

Self-training based method



Fei Pan, et al. Unsupervised Intra-domain Adaptation for Semantic Segmentation through Self-Supervision CVPR 2020

Self-training based method



$$R(X_t) = \frac{1}{HW} \sum_{h,w} \sum_{c} I_t^{(h,w,c)}$$

Fei Pan, et al. Unsupervised Intra-domain Adaptation for Semantic Segmentation through Self-Supervision CVPR 2020

Self-training based method

Table 2: The ablation study on hyperparameter λ for separating the target domain into the easy and the hard split.

GTA5 → Cityscapes						
λ	0.0	0.5	0.6	0.67	0.7	1.0
mIoU	43.8	45.2	46.0	**46.3**	45.6	45.5

(a) GTA5 → Cityscapes

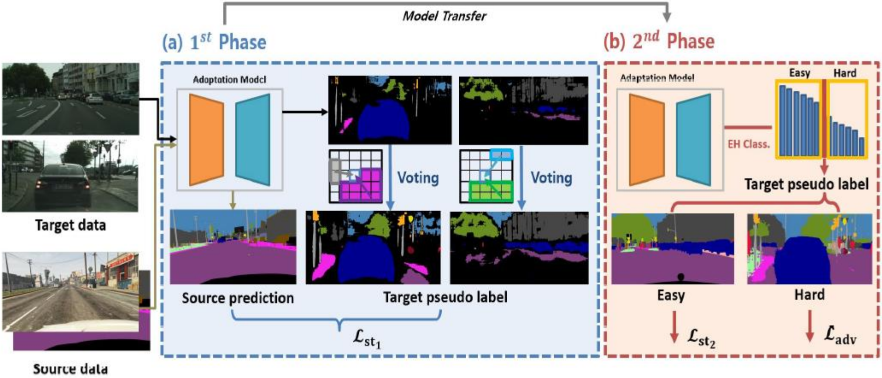
Method	road	sidewalk	building	wall	fence	pole	light	sign	veg	terrain	sky	person	rider	car	truck	bus	train	mbike	bike	mIoU
Without adaptation [26]	75.8	16.8	77.2	12.5	21.0	25.5	30.1	20.1	81.3	24.6	70.3	53.8	26.4	49.9	17.2	25.9	**6.5**	25.3	36.0	36.6
ROAD [5]	76.3	36.1	69.6	28.6	22.4	28.6	29.3	14.8	82.3	35.3	72.9	54.4	17.8	78.9	27.7	30.3	4.0	24.9	12.6	39.4
AdaptSegNet [26]	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
MinEnt [29]	84.2	25.2	77.0	17.0	23.3	24.2	33.3	**26.4**	80.7	32.1	78.7	57.5	30.0	77.0	**37.9**	44.3	1.8	**31.4**	**36.9**	43.1
AdvEnt [29]	89.9	36.5	81.6	29.2	**25.2**	28.5	32.3	22.4	83.9	34.0	77.1	57.4	27.9	83.7	29.4	39.1	1.5	28.4	23.3	43.8
Ours	**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**

(b) SYNTHIA → Cityscapes

Method	road	sidewalk	building	wall*	fence*	pole*	light	sign	veg	sky	person	rider	car	bus	mbike	bike	mIoU	mIoU*
Without adaptation [26]	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
AdaptSegNet [26]	81.7	39.1	78.4	**11.1**	0.3	25.8	6.8	9.0	79.1	80.8	54.8	21.0	66.8	34.7	13.8	29.9	39.6	45.8
MinEnt [29]	73.5	29.2	77.1	7.7	0.2	**27.0**	7.1	11.4	76.7	82.1	**57.2**	21.3	69.4	29.2	12.9	27.9	38.1	44.2
AdvEnt [29]	**87.0**	**44.1**	**79.7**	9.6	**0.6**	24.3	4.8	7.2	**80.1**	83.6	56.4	**23.7**	72.7	32.6	12.8	33.7	40.8	47.6
Ours	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**

Fei Pan, et al. Unsupervised Intra-domain Adaptation for Semantic Segmentation through Self-Supervision CVPR 2020

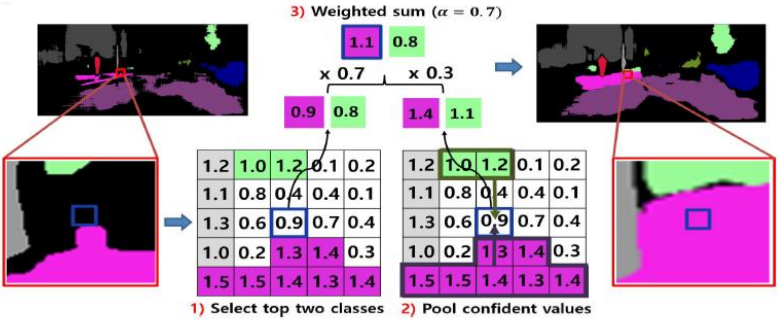
Self-training based method

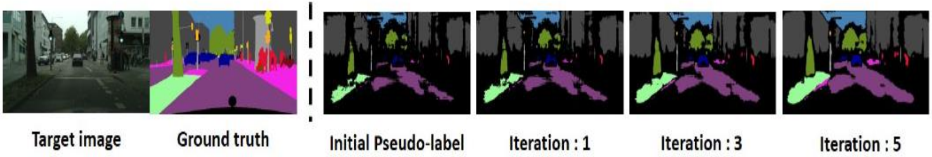


However, since only the confident predictions are taken as pseudo labels, existing self-training approaches inevitably produce sparse pseudo labels in practice.

Inkyu Shin, et al. Two-phase Pseudo Label Densi cation for Self-training based Domain Adaptation ECCV 2020

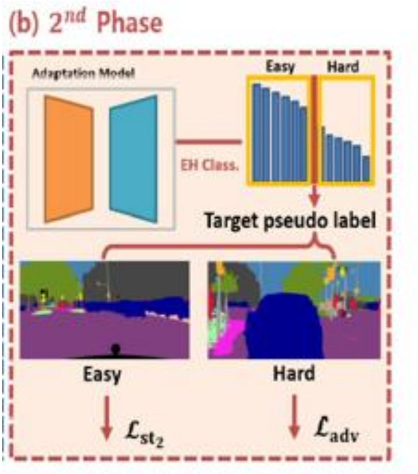
Self-training based method





Inkyu Shin, et al. Two-phase Pseudo Label Densi cation for Self-training based Domain Adaptation ECCV 2020

Self-training based method



$$conf_{\mathrm{t}} = \frac{1}{K'} \sum_{k=1}^{K'} \frac{N_{\mathrm{t}}^{k*}}{N_{\mathrm{t}}^{k}} \cdot \frac{1}{\lambda_k}$$

class-wise thresholding value

Picking up the top q portion as easy samples and consider the rest as hard samples for the training. We initially set q to 30% and add 5% in each round.

Inkyu Shin, et al. Two-phase Pseudo Label Densi cation for Self-training based Domain Adaptation ECCV 2020

Self-training based method

SYNTHIA → Cityscapes

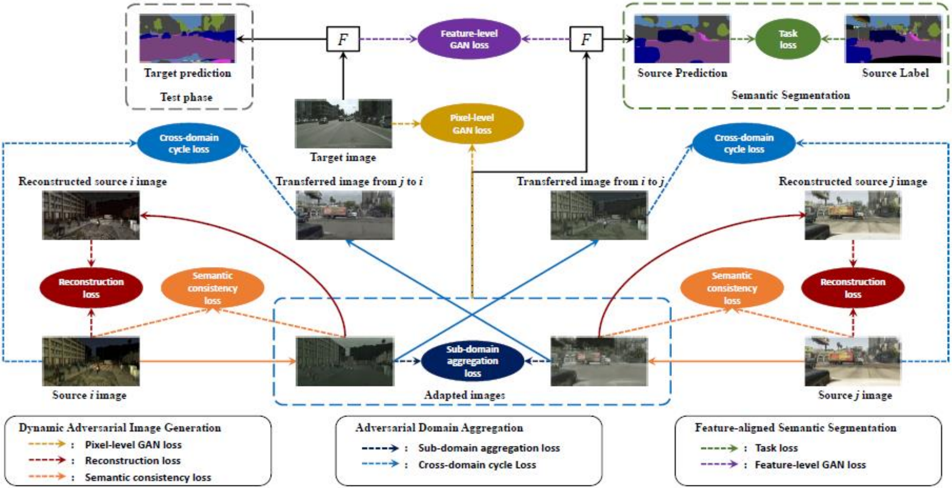
Method	Seg Model	Road	SW	Build	Wall*	Fence	Pole*	TL	TS	Veg.	Sky	PR	Rider	Car	Bus	Motor	Bike	mIoU	mIoU*	R-mIoU
Source	Deeplabv2-V	41.5	16.6	38.3	0.2	0.0	22.6	0.1	4.9	66.5	64.7	44.9	1.7	60.7	3.3	0.0	0.6	22.9	26.4	4.3
CBST [39]	Deeplabv2-V	75.7	32.3	70.2	3.5	0.0	28.6	1.4	9.0	79.8	65.6	52.9	13.7	65.8	**9.1**	**1.5**	36.4	34.1	39.5	11.5
CRST(MRKLD) [40]	Deeplabv2-V	75.1	33.5	70.8	5.6	0.0	28.7	**2.0**	**9.7**	78.9	**72.5**	51.7	11.6	63.4	7.3	1.4	38.6	34.4	39.7	11.7
CRST(MRKLD) + TPLD	Deeplabv2-V	**81.3**	**34.5**	**73.3**	11.9	0.0	26.9	0.2	6.3	**79.9**	71.2	**55.1**	**14.2**	**73.6**	5.7	0.5	**41.7**	**36.0**	**41.3**	**11.9**
Adapt-SegMap [36]	Deeplabv2-R	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	-
ADVENT [37]	Deeplabv2-R	**87.0**	44.1	79.7	9.6	0.6	24.3	4.8	7.2	80.1	**83.6**	56.4	23.7	72.7	32.6	12.8	33.7	40.8	47.6	16.6
CLAN [25]	Deeplabv2-R	81.3	37.3	80.1	-	-	-	16.1	13.7	78.2	81.5	53.4	21.2	73.0	32.9	22.6	30.7	-	47.8	-
Source	Deeplabv2-R	45.9	21.4	63.0	7.3	0.0	33.6	4.5	14.4	**81.6**	79.7	55.3	16.7	67.5	21.3	7.5	19.0	33.7	38.3	13.8
CBST [39]	Deeplabv2-R	68.0	29.9	76.3	10.8	1.4	33.9	**22.8**	29.5	77.6	78.3	60.6	28.3	81.6	23.5	18.8	39.8	42.6	48.9	23.2
CRST(MRKLD) [40]	Deeplabv2-R	67.7	32.2	73.9	10.7	1.6	37.4	22.2	**31.2**	80.8	80.5	**60.8**	**29.1**	82.8	25.0	19.4	**45.3**	43.8	50.1	24.7
CRST(MRKLD) + TPLD	Deeplabv2-R	80.9	**44.3**	**82.2**	19.9	0.3	40.6	20.5	30.1	77.2	80.9	60.6	25.5	**84.8**	**41.1**	**24.7**	43.7	**47.3**	**53.5**	**27.4**
Source	Deeplabv3-R	45.5	19.0	71.3	6.2	0.0	27.4	11.3	15.3	79.4	79.4	58.3	9.2	79.7	33.0	6.0	8.8	34.4	39.7	13.0
CBST [39]	Deeplabv3-R	45.2	19.4	**81.8**	15.7	0.2	33.3	**20.8**	24.9	85.0	82.2	64.6	26.7	**84.8**	48.8	22.9	43.9	43.8	50.1	26.4
CRST(MRKLD) [40]	Deeplabv3-R	52.3	21.9	80.0	17.2	0.8	32.4	17.9	**31.1**	84.8	83.5	63.7	28.5	83.1	37.2	19.1	52.5	44.1	50.4	26.3
CRST(MRKLD) + TPLD	Deeplabv3-R	**70.9**	**29.5**	80.6	18.4	0.4	26.6	19.9	30.9	**85.5**	**86.3**	**66.0**	**32.9**	84.4	**51.1**	**29.3**	**56.2**	**48.1**	**55.7**	**29.5**

Inkyu Shin, et al. Two-phase Pseudo Label Densi cation for Self-training based Domain Adaptation ECCV 2020

Outline

- Introduction
- Adversarial training based methods
- Self-training based methods
- Multi-source methods

Multi-source methods



Sicheng Zhao, et al. Multi-source Domain Adaptation for Semantic Segmentation NuerIPS 2019

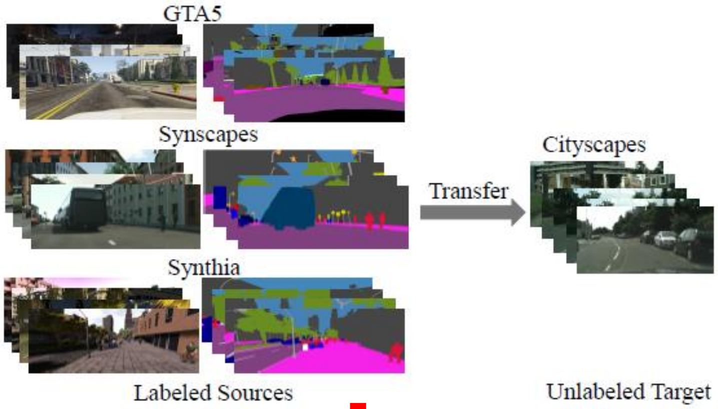
Multi-source methods

Table 1: Comparison of the proposed MADAN model with several state-of-the-art domain adaptation methods. The full names of each property from the second to the last columns are pixel-level alignment, feature-level alignment, semantic consistency, cycle consistency, multiple sources, domain aggregation, one task network, and fine-grained prediction, respectively.

	pixel	feat	sem	cycle	multi	aggr	one	fine
ADDA [25]	✗	✓	–	–	✗	–	✓	✓
CycleGAN [39]	✓	✗	✗	✓	✗	–	✓	✗
PiexlDA [37]	✓	✗	✗	✗	✗	–	✓	✓
SBADA [41]	✓	✗	✓	✓	✗	–	✓	✗
GTA-GAN [42]	✓	✓	✗	✗	✗	–	✓	✗
DupGAN [43]	✓	✓	✓	✗	✗	–	✓	✗
CyCADA [32]	✓	✓	✓	✓	✗	–	✓	✓
DCTN [68]	✗	✓	–	–	✓	✗	✗	✗
MDAN [69]	✗	✓	–	–	✓	✗	✓	✗
MMN [70]	✗	✓	–	–	✓	✗	✗	✗
MADAN (ours)	✓	✓	✓	✓	✓	✓	✓	✓

Sicheng Zhao, et al. Multi-source Domain Adaptation for Semantic Segmentation NuerIPS 2019

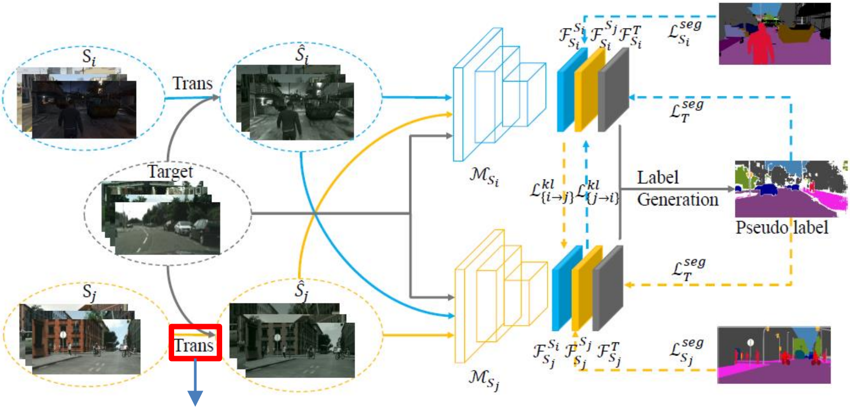
Multi-source methods



Collaborative Learning

Jianzhong He, et al. Multi-Source Domain Adaptation with Collaborative Learning for Semantic Segmentation CVPR 2021

Multi-source methods



$$\hat{\mathcal{X}}_S^{LAB} = \frac{(\mathcal{X}_S^{LAB} - \mu_S)}{\sigma_S} * \sigma_T + \mu_T$$

Jianzhong He, et al. Multi-Source Domain Adaptation with Collaborative Learning for Semantic Segmentation CVPR 2021