[E-DEEP] 2021-12-20 논문리뷰

Prototypical Pseudo Label Denoising and Target Structure Learning for Domain Adaptive Semantic Segmentation (CVPR2021)

저자 : Pan Zhang1 *, Bo Zhang2 , Ting Zhang2 , Dong Chen2 , Yong Wang1 , Fang Wen2 1University of Science and Technology of China 2Microsoft Research Asia

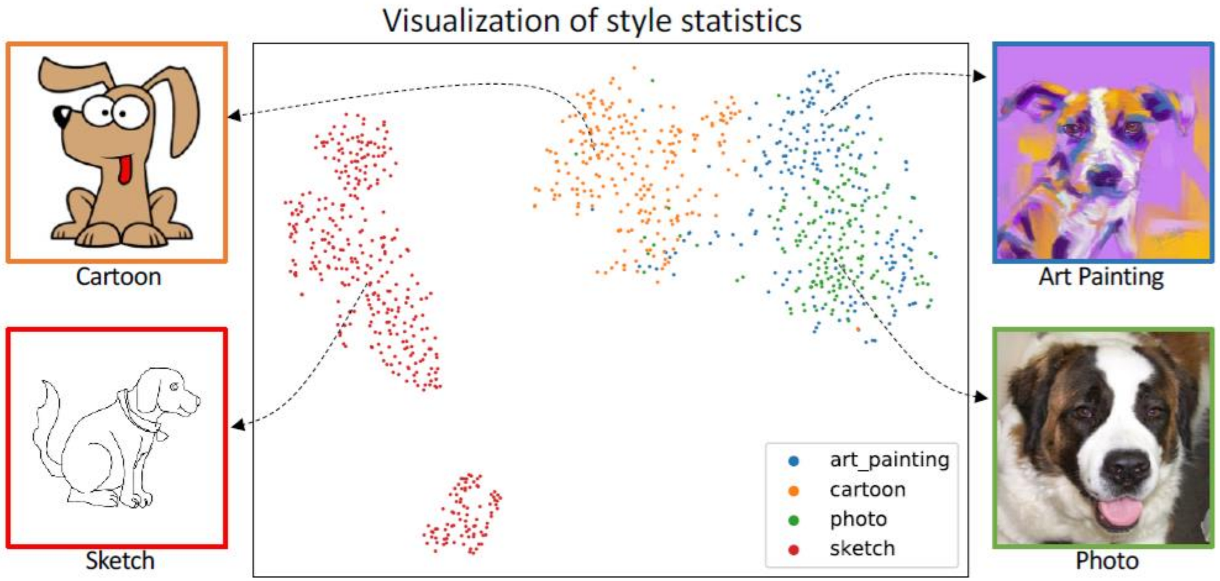
이유정

논문 링크 : https://arxiv.org/pdf/2101.10979.pdf

About Topic

- **Domain Adaptation :** Update data distribution in simulations to fit real-world environments

- **Source domain** : The environment in which we can adapt to all characteristics.
- **Target domain** : The environment in which you want to transfer the model.



Setting & Tasks

SETTING : Unsupervised Domain Adaptation

- **Source domain** : GTA5 dataset, Synthia dataset (label o)
- **Target domain** : Cityscapes dataset (label x)

Task : Semantic Segmentation

Introduction

UDA methods

Source Image → Network → Source Prediction ↔ Source label

Target Image → Network → Target Prediction ↔ Target Pseudo label

Problems

1. Strict Confidence Threshold 를 사용해서 PGT를 만드는 것은 퀄리티를 보장하지 않음
2. Source와 Target의 분산이 너무 달랐다면 제대로 작동하지 않음.

Introduction

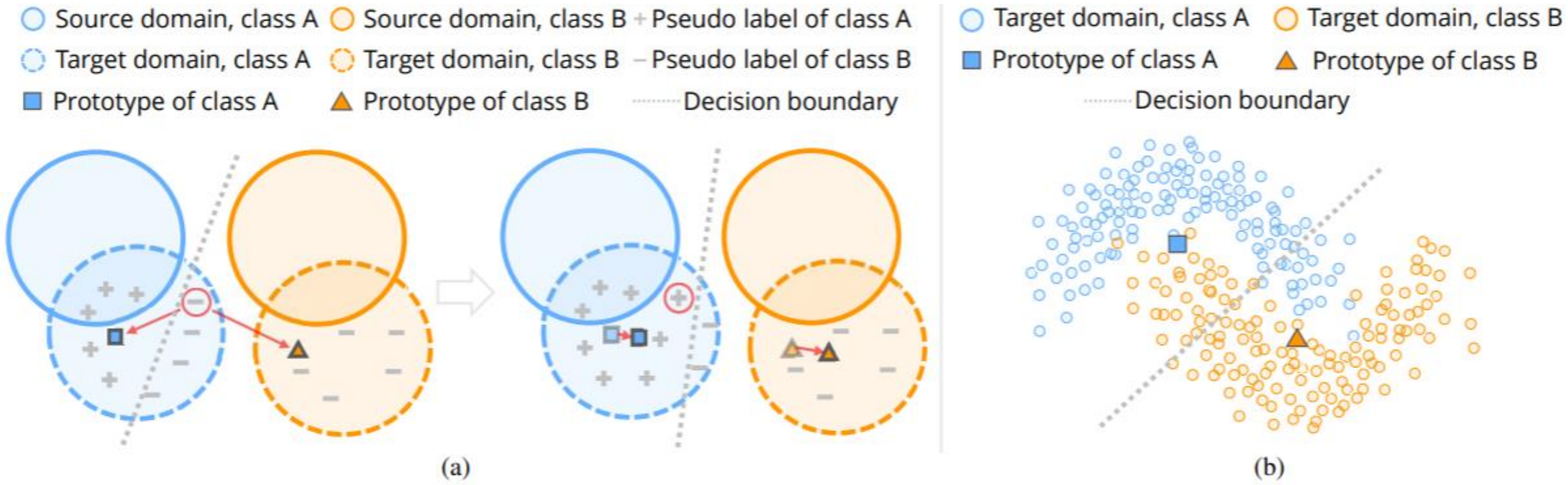


Figure 1: **We illustrate the existing issues of self-training by visualizing the feature space.** (a) The decision boundary (dashed line) crosses the distribution of the target data and induces incorrect pseudo label predictions. This is because the network is unaware of the target distribution when generating pseudo labels. To solve this, we calculate the prototypes of each class on-the-fly and rely on these prototypes to online correct the false pseudo labels. (b) The network may induce dispersed feature distribution in the target domain which is hardly differentiated by a linear classifier.

1. Strict Confidence Threshold 를 사용해서 PGT를 만드는 것은 퀄리티를 보장하지 않음
   → **soft pseudo label을 online correction**

2. Source와 Target의 분산이 너무 달랐다면 제대로 작동하지 않음.
   → **soft prototypical assignment를 augmented view를 학습**

Preliminary

Loss function & PGT

$$\ell_{ce}^{t} = -\sum_{i=1}^{H\times W}\sum_{k=1}^{K}\hat{y}_t^{(i,k)}\log(p_t^{(i,k)}), \qquad (1)$$

$$\hat{y}_t^{(i,k)} = \begin{cases} 1, & \text{if } k = \arg\max_{k'} p_t^{(i,k')} \\ 0, & \text{otherwise} \end{cases}$$

Method

PGT Generation & Updates

$$\hat{y}_t^{(i,k)} = \xi(\omega_t^{(i,k)} p_{t,0}^{(i,k)}), \tag{3}$$

where $\omega_t^{(i,k)}$ is the weight we propose for modulating the probability and changes as the training proceeds. The

$$\omega_t^{(i,k)} = \frac{\exp(-\|\tilde{f}(x_t)^{(i)} - \eta^{(k)}\|/\tau)}{\sum_{k'} \exp(-\|\tilde{f}(x_t)^{(i)} - \eta^{(k')}\|/\tau)}, \tag{4}$$

where $\tilde{f}$ denotes the momentum encoder [24] of the feature extractor f, as we desire a reliable feature estimation for x_t, and τ is the softmax temperature empirically set to $\tau = 1$. In this equation, $\omega_t^{(i,k)}$ actually approximates the trust confidence of $x_t^{(i)}$ belonging to the kth class. Note that

Prototype computation

$$\eta^{(k)} = \frac{\sum_{x_t \in \mathcal{X}_t} \sum_i f(x_t)^{(i)} * \mathbb{1}(\hat{y}_t^{(i,k)} == 1)}{\sum_{x_t \in \mathcal{X}_t} \sum_i \mathbb{1}(\hat{y}_t^{(i,k)} == 1)}, \tag{5}$$

$$\eta^{(k)} \leftarrow \lambda \eta^{(k)} + (1 - \lambda) \eta'^{(k)}, \tag{6}$$

Pseudo label training loss

Symmetric cross entropy loss

$$\ell_{sce}^t = \alpha \ell_{ce}(p_t, \hat{y}_t) + \beta \ell_{ce}(\hat{y}_t, p_t), \tag{7}$$

Method

Structure learning by enforcing consistency

$$z_{\mathcal{T}}^{(i,k)} = \frac{\exp(-\|\tilde{f}(\mathcal{T}(x_t))^{(i)} - \eta^{(k)}\|/\tau)}{\sum_{k'} \exp(-\|\tilde{f}(\mathcal{T}(x_t))^{(i)} - \eta^{(k')}\|/\tau)}, \quad (8)$$

where $\tau = 1$ by default. Likewise, the soft assignment $z_{\mathcal{T}'}$ for $\mathcal{T}'(x_t)$ can be obtained in the same manner except that we use the original trainable feature extractor f. Since z_t

$$\ell_{kl}^t = \mathrm{KL}\left(z_{\mathcal{T}} \| z_{\mathcal{T}'}\right). \quad (9)$$

$$\ell_{reg}^t = -\sum_{i=1}^{H\times W} \sum_{j=1}^{K} \log p_t^{(i,k)}. \quad (10)$$

$$\ell_{total} = \ell_{ce}^s + \ell_{sce}^t + \gamma_1 \ell_{kl}^t + \gamma_2 \ell_{reg}^t. \quad (11)$$

Distillation to self-supervised model

$$\ell_{\mathrm{KD}} = \ell_{ce}^s(p_s, y_s) + \ell_{ce}^t(p_t^\dagger, \xi(p_t)) + \beta\mathrm{KL}(p_t \| p_t^\dagger), \quad (12)$$

Results

	road	sideway	building	wall	fence	pole	light	sign	vege.	terrace	sky	person	rider	car	truck	bus	train	motor	bike	mIoU	gain
Source	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	+0.0
AdaptSeg [55]	86.5	25.9	79.8	22.1	20.0	23.6	33.1	21.8	81.8	25.9	75.9	57.3	26.2	76.3	29.8	32.1	7.2	29.5	32.5	41.4	+4.8
CyCADA [27]	86.7	35.6	80.1	19.8	17.5	38.0	39.9	41.5	82.7	27.9	73.6	64.9	19.0	65.0	12.0	28.6	4.5	31.1	42.0	42.7	+6.1
CLAN [37]	87.0	27.1	79.6	27.3	23.3	28.3	35.5	24.2	83.6	27.4	74.2	58.6	28.0	76.2	33.1	36.7	6.7	31.9	31.4	43.2	+6.6
APODA [68]	85.6	32.8	79.0	29.5	25.5	26.8	34.6	19.9	83.7	40.6	77.9	59.2	28.3	84.6	34.6	49.2	8.0	32.6	39.6	45.9	+9.3
PatchAlign [57]	**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	+9.9
ADVENT [58]	89.4	33.1	81.0	26.6	26.8	27.2	33.5	24.7	83.9	36.7	78.8	58.7	30.5	84.8	38.5	44.5	1.7	31.6	32.4	45.5	+8.9
BDL [35]	91.0	44.7	84.2	34.6	27.6	30.2	36.0	36.0	85.0	43.6	83.0	58.6	31.6	83.3	35.3	49.7	3.3	28.8	35.6	48.5	+11.9
FADA [61]	91.0	50.6	**86.0**	43.4	29.8	36.8	43.4	25.0	86.8	38.3	**87.4**	64.0	38.0	85.2	31.6	46.1	6.5	25.4	37.1	50.1	+13.5
CBST [75]	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	+9.3
MRKLD [76]	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	+10.5
CAG_UDA [69]	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	+13.6
Seg-Uncertainty [73]	90.4	31.2	85.1	36.9	25.6	37.5	48.8	48.5	85.3	34.8	81.1	64.4	36.8	86.3	34.9	52.2	1.7	29.0	44.6	50.3	+13.7
ProDA	87.8	**56.0**	79.7	**46.3**	**44.8**	**45.6**	**53.5**	**53.5**	**88.6**	**45.2**	82.1	**70.7**	**39.2**	**88.8**	**45.5**	**59.4**	1.0	**48.9**	**56.4**	**57.5**	**+20.9**

Table 1: Comparison results of GTA5→Cityscapes adaptation in terms of mIoU. The best score for each column is highlighted.

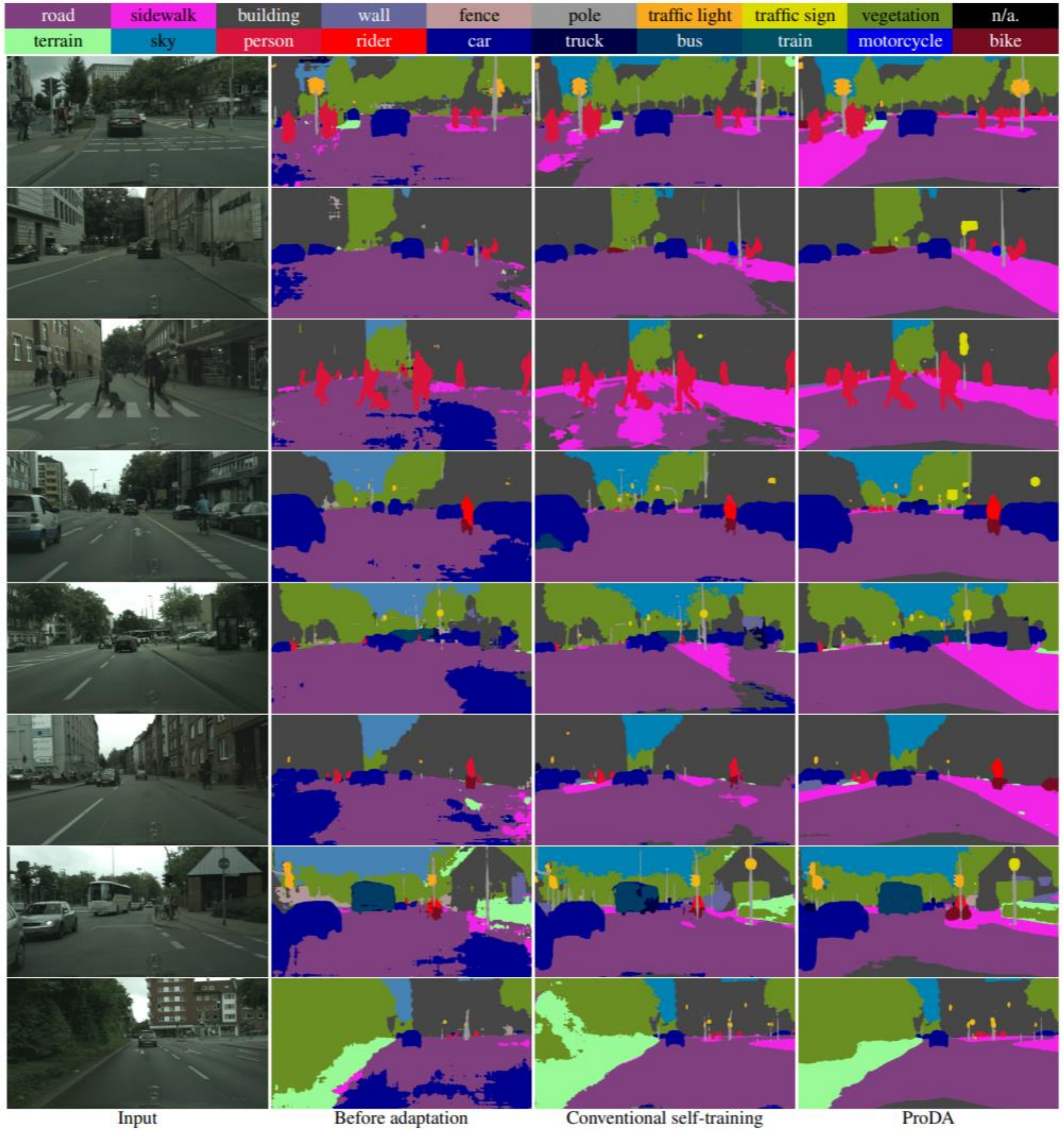


Figure 6: Qualitative results of semantic segmentation on the Cityscapes dataset. From left to right: input, before adaptation, conventional self-training, ProDA.