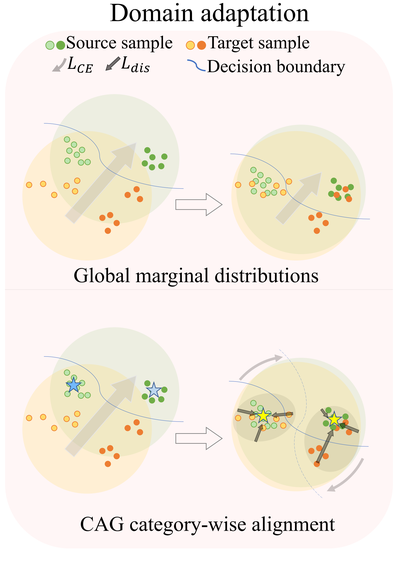
Category Anchor-Guided Unsupervised DomainAdaptation for Semantic Segmentation

Qiming Zhang[1] Jing Zhang[1] Wei Liu[2] Dacheng Tao[1]
[1]UBTECH Sydney AI Centre, School of Computer Science, Faculty of Engineering, The University of Sydney, Australia
[2]Tencent AI Lab, China

Problem definition

Despite the recent success in semantic segmentation, the difficulty in obtaining large-scale datasets hinders the further development, as annotating labels at the pixel level is prohibitively expensive and time-consuming. Unsupervised domain adaptation for semantic segmentation is thus proposed to bridge the domain gap between synthetic images (the source domain) and real images (the target domain), enabling to train neural networks only using synthetic images.

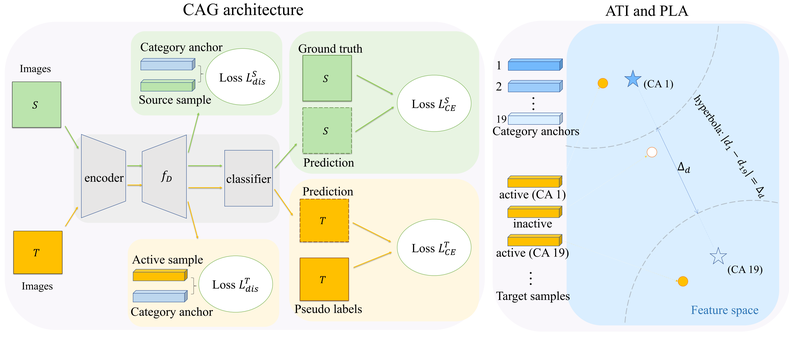
Motivation and adaptation process



- ***Motivation***
  Previous methods do not guarantee that samples of different categories in the target domain are properly separated, as shown in the upper part of the figure. Motivated by the clustering nature of features of the same category, we propose to use *category anchors* (CAs) to facilitate category-wise feature alignment.

- ***Adaptation process***
  The adaptation process is summarized as follows:
  1. The category anchors (CAs) are obtained by calculating the centroids of category-wise features in the source domain. (Stars in the lower figure.)
  2. Some features adjacent to one specific anchor are driven towards it. Then the other feature samples will follow the direction of the moving features due to the clustering nature.
  3. Finally the category-wise features are aligned, and a better segmentation result is achieved.

CAG-UDA method



- ***CAG architechture*** - given source domain image x^s and target domain image x^t, we have the features $f_D(Enc(x^s))$ and $f_D(Enc(x^t))$. Specifically, the feature at the specific index (i,j) is denoted as $f_D(Enc(x_i^s))|_j$. We then conduct *ATI* and *PLA* process for category-level feature alignment.

- ***Active target sample identification (ATI)*** - after obtaining category-wise anchors f_c^s, we identify active target feature samples for the following *PLA* process. The term 'active target samples' refers to those features near one category anchor and far away from all the other anchors. Given the distance between a target feature and the c^{th} anchor

$$d_{ijc}^t = \left\| f_c^s - f_D\left(Enc\left(x_i^t\right)\right)|_j \right\|_2, \tag{1}$$

we denote a_{ij}^t as the state of the target domain features. $\triangle_d$ is a pre-defined threshold.

$$a_{ij}^t = \begin{cases} 1, & d_{ijc^*}^t - \min_c(d_{ijc}^t) > \triangle_d, \forall c^* \neq \arg\min_c(d_{ijc}^t) \\ 0, & otherwise. \end{cases} \tag{2}$$

- ***Pseudo-label assignment (PLA)*** - in the *PLA* process, we assign pseudo-labels to those active target features according to the anchor index with the shortest distance $\arg\min_c(d_{ijc}^t)$.

- ***Objective functions*** - finally we update the model using two kinds of loss functions with available pseudo-labels. One is an L_2 norm distance loss ($Loss_{dis}^s$ and $Loss_{dis}^t$) which aims to reduce intra-category variance. The other is a segmentation loss ($Loss_{CE}^s$ and $Loss_{CE}^t$) which aims to increase inter-category variance and adjust the classifier for better alignment.

Experiments

Table 1: Quantitative results of the CAG-UDA model and SOTA methods (GTA5→Cityscapes).

	road	sidewalk	building	wall	fence	pole	light	sign	vege.	terrace	sky	person	rider	car	truck	bus	train	motor	bike	mIoU
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
AdaptSegNet[3]	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
CLAN[2]	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
BLF[1]	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
CAG-UDA	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**

Table 2: Quantitative results of the CAG-UDA model and SOTA methods (SYNTHIA→Cityscapes).

	road	sidewalk	building	wall	fence	pole	light	sign	vegetable	sky	person	rider	car	bus	motor	bike	mIoU
AdaptSegNet[3]	79.2	37.2	78.8	-	-	-	9.9	10.5	78.2	80.5	53.5	19.6	67.0	29.5	21.6	31.3	45.9
CLAN[2]	81.3	37.0	80.1	-	-	-	**16.1**	13.7	78.2	81.5	53.4	21.2	73.0	32.9	22.6	30.7	47.8
BLF[1]	86.0	46.7	80.3	-	-	-	14.1	11.6	79.2	81.3	54.1	27.9	73.7	**42.2**	**25.7**	45.3	51.4
CAG-UDA(13)	84.8	41.7	**85.5**	-	-	-	13.7	**23.0**	**86.5**	78.1	**66.3**	**28.1**	81.8	21.8	22.9	**49.0**	**52.6**

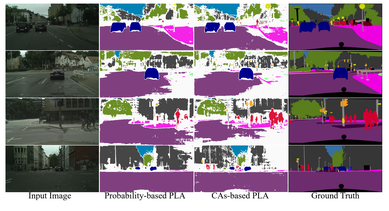


Figure 1: Comparison of PLA between the CAs-based and probability-based.

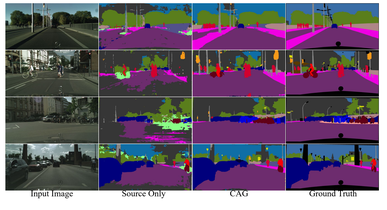


Figure 2: Qualitative results in the GTA5 → Cityscapes scenario.

Reference

[1] Y. Li, L. Yuan, and N. Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6936–6945, 2019. 2
[2] Y. Luo, L. Zheng, T. Guan, J. Yu, and Y. Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2507–2516, 2019. 2
[3] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7472–7481, 2018. 2