Cross-Domain Few-Shot Semantic Segmentation: Supplementary Material

Shuo Lei[1], Xuchao Zhang[2], Jianfeng He[1], Fanglan Chen[1], Bowen Du[3] *, and Chang-Tien Lu[1]

[1] Department of Computer Science, Virginia Tech, Falls Church, VA, USA
[2] NEC Laboratories America, Princeton, NJ, USA
[3] State Key Laboratory of Software Development Environment, Beihang University, Beijing, China

In this material, we provide more details about the proposed benchmark, the illustration of the proposed Task-adaptive Fine-tuning Inference (TFI) strategy and more experiment results in the paper. Specifically, in Section 1, we briefly introduce the selected datasets and analyze the task difficulty of each dataset in CD-FSS. Moreover, the DisFB measurement is presented, which is proposed to measure the similarity between foreground and background classes in a single image. In Section 2, we summarize the whole process of the TFI strategy. In Section 3, we provide the implementation details of transfer learning baselines. More details about the experiment results and some qualitative results are presented.

1 Task Difficulty Analysis on CD-FSS

To provide a better overview of the established benchmark, Table 1 in the paper summarizes the conceptual difference between PASCAL VOC [3] and the four cross-domain datasets. And the task difficulty for each domain is measured from two aspects: 1) domain shift (cross the datasets) and 2) class distinction in a single image (within the dataset). Different from the image classification task, the challenges of the segmentation task are decided by not only the domain shift but the discrimination between classes in a single image. For example, even if Chest X-ray data are far different from PASCAL VOC, predicting the mask in X-ray images is easier than the satellite images, as the foreground class is distinct from the background class.

We adopt Kullback-Leibler Divergence (KL-divergence) to measure the similarity between foreground and background classes in a single image, denoted as DisFB. Specifically, for each image, we first get the masked image for each class by multiplying the image and its corresponding segmentation map. Then the masked images are fed to the backbone network. Here, we adopt the final average pooling features in Inception network [8] pretrained in ImageNet [6]. Finally, the similarity between foreground and background classes is measured by calculating the KL-divergence between the probability distribution of foreground and background classes. The larger the DisFB is, the less the discrimination between classes in the dataset. Noted that DisFB is calculated in the same domain, not the domain shift measurement.

* Corresponding author.

2 Task-adaptive Fine-tuning Inference

For better understanding, the whole testing process for our method is summarized in Algorithm 1. Specifically, given each task, we extract a series of intermediate features of support and query images. Then all the intermediate features are transformed with the linear transformation matrices constructed by PATM. The originally predicted mask of the query image is obtained using transformed domain-agnostic features via hypercorrelation decoder [5]. Next, we compute the foreground class prototype with the predicted mask. The anchor layers are updated via $\mathcal{L}_{kl}$, which measures the similarity between the foreground class prototype of support and query sets. The final predicted mask of query images are obtained with the updated PATM.

Algorithm 1 Task-adaptive Fine-tuning Inference Strategy

Input: A testing set $\mathcal{D}_{test}$ generated from the target domain, feature extractor f_θ, Anchor Layers $\{[\mathbf{a}_{f,l}, \mathbf{a}_{b,l}]\}_l^3$, hypercorrelation decoder $g(\cdot)$, fine-tuning iteration K.

Output: Prediction masks of query images.

1: **for** each episode $(\mathcal{S}_i, \mathcal{Q}_i) \in \mathcal{D}_{test}$ **do**
2: **while** $k \leq K$ **do**
3: get $\{\mathbf{F}_l^s\}_{l=1}^L$, $\{\mathbf{F}_l^q\}_{l=1}^L$ using the extractor f_θ
4: **for** l in layer set in $\{1, \ldots, L\}$ **do**
5: $\mathbf{p}_{f,l}^s \leftarrow \frac{\sum_i \mathbf{F}_{l,i}^s \zeta_l(\mathbf{M}^s)_i}{\sum_i \zeta_l(\mathbf{M}^s)_i}$, $\mathbf{p}_{b,l}^s \leftarrow \frac{\sum_i \mathbf{F}_{l,i}^s \zeta_l(\sim\mathbf{M}^s)_i}{\sum_i \zeta_l(\sim\mathbf{M}^s)_i}$
6: $\mathbf{P}_l^s \leftarrow [\frac{\mathbf{p}_{f,l}^s}{\|\mathbf{p}_{f,l}^s\|}, \frac{\mathbf{p}_{b,l}^s}{\|\mathbf{p}_{b,l}^s\|}]$
7: $\mathbf{A}_l \leftarrow [\frac{\mathbf{a}_{f,l}}{\|\mathbf{a}_{f,l}\|}, \frac{\mathbf{a}_{b,l}}{\|\mathbf{a}_{b,l}\|}]$, using the corresponding Anchor Layer $\mathbb{R}^{C_l \times C_l}$
8: $\mathbf{W}_l = \mathbf{A}_l\{\mathbf{P}_l^{sT}\mathbf{P}_l^s\}^{-1}\mathbf{P}_l^{sT}$
9: $\mathbf{C}_l(i,j) = \text{ReLU}\left(\frac{\mathbf{W}_l\mathbf{F}_l^q(i)\cdot\mathbf{W}_l\hat{\mathbf{F}}_l^s(j)}{\|\mathbf{W}_l\mathbf{F}_l^q(i)\|\|\mathbf{W}_l\hat{\mathbf{F}}_l^s(j)\|}\right)$
10: **end for**
11: $\hat{\mathbf{M}} \leftarrow g(\{\mathbf{C}_l\}_l^L)$
12: $\mathcal{L}_{kl} \leftarrow \sum_{l=1}^L D_{KL}(\mathbf{p}_{f,l}^s || \hat{\mathbf{p}}_{f,l}^q)$, where $\hat{\mathbf{p}}_{f,l}^q \leftarrow \frac{\sum_i \mathbf{F}_{l,i}^q \hat{\mathbf{M}}_{l,i} \mathbb{1}[\hat{\mathbf{M}}_{l,i} \geqslant \tau]}{\sum_i \hat{\mathbf{M}}_{l,i}}$
13: Update Anchor Layers with $\mathcal{L}_{kl}$ using Adam optimizer.
14: $k = k + 1$
15: **end while**
16: Predict the final mask of query images following step 3-11 with the updated Anchor Layers.
17: **end for**

3 Experiments

3.1 Baseline Implementation Details

We provide the implementation details of the transfer learning based baselines compared in the Table 2 in the paper. Since CD-FSS is a new task, we adapt previous work to realize the idea of transfer learning for CD-FSS:

- **Fixed Feature Extractor for FCN**: We fine-tune FCN-32s [4] pretrained on all the classes in PASCAL VOC. During testing, we extract dense pixel-level features from both images in the support set and the query image. Then we train the classifier, 1-NN and logistic regression, to map dense fc-7 features from the support set to their corresponding labels and use it to generate the predicted mask.
- **Fixed Feature Extractor for DeeplabV3**: We also adopt DeeplabV3 [2] pretrained on all the classes in PASCAL VOC [3] as a feature extractor. Similarly, we train the classifier (e.g. 1NN and logistic regression) based on the dense features generated by the ASPP module and use it to produce the predicted mask in the query set.
- **Fine-tuning last-k layers for FCN (Ft Last-k)**: We adopt same testing strategy as [1, 7], for each test iteration we fine-tune the trained segmentation network on examples in the support set and test on the query image. Here, we adopt FCN-32s pretrained on all the classes in PASCAL VOC. In this paper, we consider fine-tuning the last-1 (fc-8), last-2 (fc-7, fc-8), last-3 (fc-6, fc-7, fc-8) fully connected layers.
- **Fine-tuning last-k layers for DeeplabV3 (Ft Last-k)**: For a fair comparison, we also fine-tune DeeplabV3 pretrained on all the classes in PASCAL VOC. To avoid overfitting, we only fine-tune the last-2 and the last-1 convolutional layers following the ASPP module.

3.2 Comparison with Few-Shot Cross-Domain Classification Method

Our work focuses on the few-shot *segmentation* task which is different and more complicated from the few-shot *classification* task in [9]. Due to different focused issues, the proposed model in [9] cannot be applied in the task directly. Thus, we adapt the method in [9] to address the few-shot semantic segmentation task by combining the feature-wise transformation layer (FWT) proposed in [9] with PANet. As seen from the table, FWT only slightly improves the performance in FSS-1000 and leads to performance reductions in other three datasets. In contrast, our model performs effectively in all datasets, surpassing PANet+FWT by 9.73% and 6.89% average accuracies in 1-shot and 5-shot settings, respectively.

Table 1. Comparison with the adapted few-shot cross-domain classification method on 1-way 1-shot and 5-shot few-shot semantic segmentation task on CD-FSS.

Methods	Deepglobe		ISIC		Chest X-ray		FSS-1000		Average	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
PANet	36.55	**45.43**	25.29	33.99	57.75	69.31	69.15	71.68	47.19	55.10
PANet+FWT	35.35	44.94	23.02	32.54	57.10	69.23	69.85	72.90	46.33	54.90
PATNet	**37.89**	42.97	**41.16**	**53.58**	**66.61**	**70.20**	**78.59**	**81.23**	**56.06**	**61.99**

3.3 Additional Qualitative Results

As done in the main paper, we show the support labels and ground truth labels in blue. Predicted segmentation for query images are in plum. From these results, we can observe that the quality of segmentation is improved from 1-shot to 5-shot.

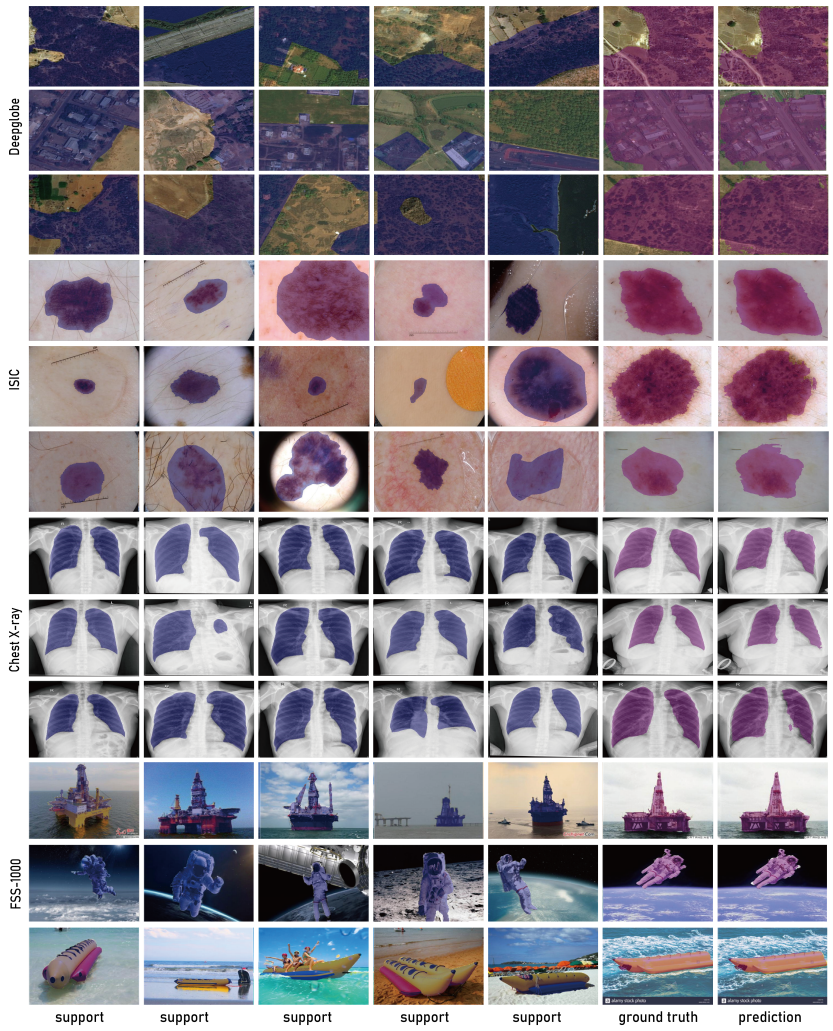

Fig. 1. Qualitative results of our model in 1-way 5-shot segmentation.

Table 2. PATNet with ResNet50 backbone network. The number of intermediate features extracted from backbone network amounts to 13, i.e., L=13. The Anchor Layer is the main difference part from HSNet [5] in the training process. Generalizability can be greatly increased by introducing only 7.1K learnable parameters.

Layer	Input		Output		Operation	# params.
ResNet50 Backbone	I_q	(3,400,400)	$\{\mathbf{F}_l^q\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4	Series of 2D Convs	23.6M(frozen)
	I_s	(3,400,400)	$\{\mathbf{F}_l^s\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4		
Masking Layer	$\{\mathbf{F}_l^s\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4	$\{\mathbf{F}_l^s\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4	Bilinear Interpolation Hadamard Product	-
	$\mathbf{M}_s$	(1,400,400)				
Anchor Layer	$\{\mathbf{F}_l^s\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4	$\{\mathbf{F}_l^q\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4	Linear Transformation	7.17K
	$\{\mathbf{F}_l^s\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4	$\{\mathbf{F}_l^s\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4		
Correlation Layer	$\{\mathbf{F}_l^s\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4	$\{\mathbf{C}_p\}_{p=1}^{3}$	(3,13,13,13,13) (5,25,25,25,25) (4,50,50,50,50)	Cosine Similarity	-
	$\{\mathbf{F}_l^s\}_{l=1}^{13}$	(2048,13,13)×3 (1024,25,25)×6 (512,50,50)×4				
Squeezing Block f_3^{sqz}	$\mathbf{C}_3$	(3,13,13,13,13)	$\mathbf{C}_3^{sqz}$	(128,13,13,2,2)	(CP 4D conv Group Norm ReLU) × 3	168K
Squeezing Block f_2^{sqz}	$\mathbf{C}_2$	(6,25,25,25,25)	$\mathbf{C}_2^{sqz}$	(128,25,25,2,2)	(CP 4D conv Group Norm ReLU) × 3	172K
Squeezing Block f_1^{sqz}	$\mathbf{C}_1$	(4,50,50,50,50)	$\mathbf{C}_1^{sqz}$	(128,50,50,2,2)	(CP 4D conv Group Norm ReLU) × 3	203K
Mixing Block f_2^{mix}	$\mathbf{C}_3^{sqz}$	(128,13,13,2,2)	$\mathbf{C}_2^{mix}$	(128,25,25,2,2)	Bilinear Interpolation Element-wise Addition (CP 4D conv Group Norm ReLU) × 3	886K
	$\mathbf{C}_2^{sqz}$	(128,25,25,2,2)				
Mixing Block f_1^{mix}	$\mathbf{C}_2^{mix}$	(128,25,25,2,2)	$\mathbf{C}_1^{mix}$	(128,50,50,2,2)	Bilinear Interpolation Element-wise Addition (CP 4D conv Group Norm ReLU) × 3	886K
	$\mathbf{C}_1^{sqz}$	(128,50,50,2,2)				
Pooling Layer	$\mathbf{C}_1^{mix}$	(128,50,50,2,2)	$\mathbf{Z}$	(128,50,50)	Average-pooling	-
Decoder Layer	$\mathbf{Z}$	(128,50,50)	$\mathbf{M}_q$	(2,400,400)	Series of 2D Convs with Bilinear Interpolation	259K

References

1. Caelles, S., Maninis, K.K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L.: One-shot video object segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 221–230 (2017)
2. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
3. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**(2), 303–338 (2010)
4. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
5. Min, J., Kang, D., Cho, M.: Hypercorrelation squeeze for few-shot segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6941–6952 (2021)
6. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
7. Shaban, A., Bansal, S., Liu, Z., Essa, I., Boots, B.: One-shot learning for semantic segmentation. arXiv preprint arXiv:1709.03410 (2017)
8. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)
9. Tseng, H.Y., Lee, H.Y., Huang, J.B., Yang, M.H.: Cross-domain few-shot classification via learned feature-wise transformation. arXiv preprint arXiv:2001.08735 (2020)