Supplementary
FREDOM: Fairness Domain Adaptation Approach to Semantic Scene Understanding

Thanh-Dat Truong[1], Ngan Le[1], Bhiksha Raj[2], Jackson Cothren[3], Khoa Luu[1]

[1]CVIU Lab, University of Arkansas, USA [2]Carnegie Mellon University, USA

[3]Dep. of Geosciences, University of Arkansas, USA

{tt032, thile, jcothre, khoaluu}@uark.edu, bhiksha@cs.cmu.edu

1. Proof Of Eqn. (8)

Taking the logarithm of Eqn. (6) and Eqn. (7), the optimization formula can be derived as follows:

$$
\begin{aligned}
\theta^* &= \arg\min_{\theta} \left[\mathbb{E}_{\mathbf{x}_s \sim p_s(\mathbf{x}_s), \hat{\mathbf{y}}_\mathbf{s} \sim p_s(\hat{\mathbf{y}}_s)} \mathcal{L}_s(\mathbf{y}_s, \hat{\mathbf{y}}_s) \frac{p'_s(\mathbf{y}_s)}{p_s(\mathbf{y}_s)} + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \mathcal{L}_t(\mathbf{y}_t) \frac{p'_s(\mathbf{y}_t)}{p_s(\mathbf{y}_t)} \right] \\
&\simeq \arg\min_{\theta} \left[\mathbb{E}_{\mathbf{x}_s \sim p_s(\mathbf{x}_s), \hat{\mathbf{y}}_\mathbf{s} \sim p_s(\hat{\mathbf{y}}_s)} \log \left(\mathcal{L}_s(\mathbf{y}_s, \hat{\mathbf{y}}_s) \frac{p'_s(\mathbf{y}_s)}{p_s(\mathbf{y}_s)} \right) + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \log \left(\mathcal{L}_t(\mathbf{y}_t) \frac{p'_s(\mathbf{y}_t)}{p_s(\mathbf{y}_t)} \right) \right] \\
&= \arg\min_{\theta} \left[\mathbb{E}_{\mathbf{x}_s \sim p_s(\mathbf{x}_s), \hat{\mathbf{y}}_\mathbf{s} \sim p_s(\hat{\mathbf{y}}_s)} \left(\log \mathcal{L}_s(\mathbf{y}_s, \hat{\mathbf{y}}_s) + \log \frac{p'_s(\mathbf{y}_s)}{p_s(\mathbf{y}_s)} \right) + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \left(\log \mathcal{L}_t(\mathbf{y}_t) + \log \frac{p'_s(\mathbf{y}_t)}{p_s(\mathbf{y}_t)} \right) \right] \\
&= \arg\min_{\theta} \frac{1}{N} \Bigg[\mathbb{E}_{\mathbf{x}_\mathbf{s} \sim p_s(\mathbf{x}_s), \hat{\mathbf{y}}_\mathbf{s} \sim p_s(\hat{\mathbf{y}}_s)} \log \mathcal{L}_s(\mathbf{y}_s, \hat{\mathbf{y}}_s) + \mathbb{E}_{\mathbf{x}_\mathbf{s} \sim p_s(\mathbf{x}_s)} \frac{1}{N} \sum_{k=1}^{N} \log \left(\frac{p'_s(y^k_s) p'_s(\mathbf{y}_s^{\setminus k} | y^k_s)}{p_s(y^k_s) p_s(\mathbf{y}_s^{\setminus k} | y^k_s)} \right) \\
&\qquad + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \log \mathcal{L}_t(\mathbf{y}_t) + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \frac{1}{N} \sum_{k=1}^{N} \log \left(\frac{p'_s(y^k_t) p'_s(\mathbf{y}_t^{\setminus k} | y^k_t)}{p_s(y^k_t) p_s(\mathbf{y}_t^{\setminus k} | y^k_t)} \right) \Bigg] \\
&= \arg\min_{\theta} \frac{1}{N} \Bigg[\mathbb{E}_{\mathbf{x}_\mathbf{s} \sim p_s(\mathbf{x}_s), \hat{\mathbf{y}}_\mathbf{s} \sim p_s(\hat{\mathbf{y}}_s)} \log \mathcal{L}_s(\mathbf{y}_s, \hat{\mathbf{y}}_s) + \mathbb{E}_{\mathbf{x}_\mathbf{t} \sim p_t(\mathbf{x}_t))} \log \mathcal{L}_t(\mathbf{y}_t) \\
&\qquad + \frac{1}{N} \sum_{KBk=1}^{N} \left(\mathbb{E}_{\mathbf{x}_s \sim p_s(\mathbf{x}_s)} \log \left(\frac{p'_s(y^k_s)}{p_s(y^k_s)} \right) + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \log \left(\frac{p'_s(y^k_t)}{p_s(y^k_t)} \right) \right) \\
&\qquad + \frac{1}{N} \sum_{k=1}^{N} \left(\mathbb{E}_{\mathbf{x}_s \sim p_s(\mathbf{x}_s)} \log \left(\frac{p'_s(\mathbf{y}_s^{\setminus k} | y^k_s)}{p_s(\mathbf{y}_s^{\setminus k} | y^k_s)} \right) + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \log \left(\frac{p'_s(\mathbf{y}_t^{\setminus k} | y^k_t)}{p_s(\mathbf{y}_t^{\setminus k} | y^k_t)} \right) \right) \Bigg]
\end{aligned}
$$

In addition, minimizing $\log \mathcal{L}_s(\mathbf{y}_s, \hat{\mathbf{y}}_s)$ and $\log \mathcal{L}_t(\mathbf{y}_t)$ is equivalent to minimizing $\mathcal{L}_s(\mathbf{y}_s, \hat{\mathbf{y}}_s)$ and $\mathcal{L}_t(\mathbf{y}_t)$. Therefore, the above formula can be further derived as follows:

$$
\begin{aligned}
\theta^* \simeq \arg\min_{\theta} \Bigg[& \mathbb{E}_{\mathbf{x}_\mathbf{s} \sim p_s(\mathbf{x}_s), \hat{\mathbf{y}}_\mathbf{s} \sim p_s(\hat{\mathbf{y}}_s)} \mathcal{L}_s(\mathbf{y}_s, \hat{\mathbf{y}}_s) + \mathbb{E}_{\mathbf{x}_\mathbf{t} \sim p_t(\mathbf{x}_t))} \mathcal{L}_t(\mathbf{y}_t) \\
& + \frac{1}{N} \sum_{k=1}^{N} \left(\mathbb{E}_{\mathbf{x}_s \sim p_s(\mathbf{x}_s)} \log \left(\frac{p'_s(y^k_s)}{p_s(y^k_s)} \right) + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \log \left(\frac{p'_s(y^k_t)}{p_s(y^k_t)} \right) \right) \\
& + \frac{1}{N} \sum_{k=1}^{N} \left(\mathbb{E}_{\mathbf{x}_s \sim p_s(\mathbf{x}_s)} \log \left(\frac{p'_s(\mathbf{y}_s^{\setminus k} | y^k_s)}{p_s(\mathbf{y}_s^{\setminus k} | y^k_s)} \right) + \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \log \left(\frac{p'_s(\mathbf{y}_t^{\setminus k} | y^k_t)}{p_s(\mathbf{y}_t^{\setminus k} | y^k_t)} \right) \right) \Bigg]
\end{aligned}
$$

2. Implementation Details

Our proposed FREDOM approach is implemented based on the implementation of SAC [1], DAFormer [6], and ImageGPT [2]. In particular, we borrow the implementation of DeepLab-V2 of SAC [1]. We use DeepLab-V2 with ResNet-101 backbone. The Atrous Spatial Pyramid Pooling with a sampling rate of $\{6, 12, 18, 24\}$ has been implemented in our work. The predicted segmentation is computed from the output of layer *conv5*. For the Transformer backbone, we borrow the implementation of DAFormer[2]. Our Transformer architecture also uses the MiT-B5 encoder as the backbone to produce a feature pyramid with dimensions of $\{64, 128, 320, 512\}$. To output the predicted segmentation, we use the decoder with a dilation rate of $\{1, 6, 12, 18\}$. Both DeepLab-V2 and Transformer of segmentation networks are pretrained on ImageNet-1K. For the Conditional Structure Network, we adopt the implementation of Image-GPT[3]. The structure of our conditional network structure is designed based on the base version of Image GPT [2], which is a variant of GPT-2 [8]. The Conditional Structure Network G is trained on the segmentation labels of the source domain.

For the adaptation loss in the target domain $\mathcal{L}_t$, we also use the self-supervised loss with pseudo labels [1]. Our training procedure and augmentation methods are implemented based on the implementation of SAC [1] and DAFormer [6]. We also adopt the data sampling technique of SAC [1] in our training. For the mask sampling technique, during training the conditional structure network, for each image in each iteration, we randomly generate a binary mask within three cases, as mentioned in the main paper.

3. Additional Experimental Results

3.1. Ablation Study

Table 1. **Effectiveness of our Mask Sampling Approach to Fairness Improvement Using DeepLab-V2 (DL-V2).**

	$\mathcal{L}_{Class}$	$\mathcal{L}_{Cond}$			Majority Group						Minority Group													mIoU	STD
		M1	M2	M3	Road	Build.	Veget.	Car	S.Walk	Sky	Pole	Person	Terrain	Fence	Wall	Sign	Bike	Truck	Bus	Train	Tr.Light	Rider	M.bike		
DL-V2	✗	✗	✗	✗	90.3	87.2	88.1	88.6	53.5	87.3	44.4	67.3	42.2	28.5	41.1	50.1	54.4	52.5	56.9	33.7	48.9	33.1	42.6	57.4	20.9
$+\mathcal{L}_{NG}$	✗	✗	✗	✗	90.4	87.1	88.0	88.6	53.6	87.2	44.7	67.4	42.3	28.4	41.2	49.8	54.9	53.0	57.2	37.8	48.8	33.0	42.5	57.7	20.7
$+\mathcal{L}_{CB}$	✗	✗	✗	✗	90.5	87.2	88.2	88.7	53.5	87.3	44.8	67.6	42.2	28.5	41.6	51.6	53.8	54.3	57.6	37.5	49.2	33.6	43.5	58.0	20.6
DL-V2	✓	✗	✗	✗	90.6	87.3	88.1	88.8	53.7	87.4	44.9	67.7	42.3	28.6	41.9	52.9	57.6	55.2	57.5	47.6	50.8	36.9	44.9	59.2	19.8
DL-V2	✓	✓	✗	✗	90.6	87.3	88.2	88.8	53.7	87.5	44.9	67.8	42.2	29.0	41.9	53.0	57.7	55.2	57.7	48.8	50.8	37.5	45.1	59.4	19.7
DL-V2	✓	✓	✓	✗	90.8	87.5	88.5	88.9	54.0	87.6	45.1	68.4	42.3	30.4	42.1	53.6	57.8	55.3	58.7	53.7	50.8	39.5	46.1	60.1	19.3
DL-V2	✓	✓	✓	✓	**90.9**	**87.8**	**88.6**	**89.7**	**54.1**	**89.5**	**45.2**	**68.8**	**42.6**	**32.6**	**44.1**	**57.1**	**58.1**	**58.4**	**62.6**	**55.3**	**51.4**	**40.0**	**47.7**	**61.3**	**19.1**

Effectiveness of Mask Sampling Approach To further illustrate the effectiveness, we conduct additional ablation studies using DeepLabV2 (DL-V2) under the domain adaptation setting trained on the GTA5 $\rightarrow$ Cityscapes benchmark. We consider experiments of the pre-defined weight-balancing different classes [13] ($\mathcal{L}_{CB}$), normalizing gradients ($\mathcal{L}_{NG}$). In addition, we evaluate the impact of the mask sampling approaches to the fairness improvement. There are three different strategies of binary mask samplings that will be evaluated, i.e., (1) If $\mathbf{m}$ contains only one unmasked pixel (denoted as **M1**), G learns to capture structural information of segmentation conditioned on a given pixel, (2) If $\mathbf{m}$ contains more than one unmasked pixel (denoted as **M2**), it increases the flexibility of G on learning segmentation structures conditioned on unmasked pixels, (3) If $\mathbf{m}$ does not contain any unmasked pixels (denoted as **M3**), it is equivalent to learning the log-likelihood of segmentation maps. The experimental results in Table 1 show the advantages of our method. We found that $\mathcal{L}_{NG}$ stabilizes the training procedure and $\mathcal{L}_{CB}$ brings a minor improvement. Also, while $\mathcal{L}_{Cond}$ with simple binary masks sampled as M1 is not powerful enough to model the conditional structures, combining three strategies of mask samplings brings a significant performance improvement of the segmentation model, especially in classes of the minority group, and promotes fairness in the model.

3.2. Quantitative Results

Table 2 reports our experimental results using DeepLab-V2 and Transformer networks compared to prior unsupervised domain adaptation methods. The table includes the results of both benchmarks, i.e., SYNTHIA $\rightarrow$ Cityscapes and GTA5 $\rightarrow$ Cityscapes. Overall, our proposed approach has achieved State-of-the-Art performance on both benchmarks and significantly improved the IoU results of classes in the minority which means promoting fairness of the model predictions.

[1] https://github.com/visinf/da-sac
[2] https://github.com/lhoyer/DAFormer
[3] https://github.com/teddykoker/image-gpt

Table 2. Comparison of Semantic Segmentation Performance with UDA Methods Using DeepLab-V2 (**DL-V2**) and Transformer (**Trans.**).

Approach	Network	Majority Group						Minority Group													mIoU	STD
		Road	Build.	Veget.	Car	S.Walk	Sky	Pole	Person	Terrain	Fence	Wall	Sign	Bike	Truck	Bus	Train	Tr.Light	Rider	M.bike		
SYNTHIA → Cityscapes																						
CBST [15]	DL-V2	68.0	76.3	77.6	81.6	29.9	78.3	33.9	60.6	–	1.4	10.8	29.5	39.8	–	23.5	–	22.8	28.3	18.8	42.6	26.8
DACS [9]	DL-V2	80.6	81.9	83.7	82.9	25.1	90.8	37.2	67.6	–	2.9	21.5	24.0	47.6	–	38.9	–	22.7	38.3	28.5	48.3	28.4
CorDA [13]	DL-V2	93.3	85.3	84.9	85.6	61.6	90.4	37.8	69.7	–	5.1	19.6	42.8	53.9	–	38.4	–	36.6	41.8	32.6	55.0	27.3
AdvEnt [11]	DL-V2	87.0	79.7	80.1	72.7	44.1	83.6	24.3	56.4	–	0.6	9.6	7.2	33.7	–	32.6	–	4.8	23.7	12.8	40.8	31.4
DADA [12]	DL-V2	89.2	81.4	81.8	79.7	44.8	84.0	26.2	54.7	–	0.3	6.8	11.1	38.8	–	40.7	–	8.6	19.3	14.0	42.6	32.0
MaxSquare [3]	DL-V2	82.9	80.3	82.5	79.0	40.7	82.2	25.8	53.1	–	0.8	10.2	18.2	35.6	–	31.4	–	12.8	18.0	10.4	41.4	30.6
IntraDA [7]	DL-V2	84.3	79.5	80.0	78.0	37.7	84.1	24.9	57.2	–	0.4	5.3	8.4	36.5	–	38.1	–	9.2	23.0	20.3	41.7	31.0
BiMaL [10]	DL-V2	**92.8**	81.5	82.4	85.7	**51.5**	84.6	30.4	55.9	–	1.0	10.2	15.9	38.8	–	44.5	–	17.6	22.3	24.6	46.2	30.9
SAC [1]	DL-V2	89.3	85.6	87.1	87.0	47.3	89.1	43.1	63.7	–	1.3	26.6	32.0	52.8	–	35.6	–	45.6	25.3	30.3	52.6	27.9
ProDA [14]	DL-V2	87.8	**84.6**	**88.1**	88.2	45.7	84.4	44.0	**74.2**	–	0.6	**37.1**	37.0	45.6	–	51.1	–	**54.6**	24.3	40.5	55.5	26.4
FREDOM	DL-V2	86.0	**87.0**	87.1	87.1	46.3	**89.1**	**48.7**	71.2	–	**5.3**	33.3	**46.8**	**59.9**	–	**54.6**	–	53.4	**38.1**	**51.3**	**59.1**	**24.0**
TransDA-S [4]	Trans.	82.1	86.2	89.2	90.9	40.9	90.3	53.0	68.0	–	1.0	25.8	36.1	45.4	–	58.4	–	53.7	26.2	41.2	55.5	27.1
TransDA [4]	Trans.	**90.4**	86.4	**90.3**	**92.3**	**54.8**	93.0	53.8	71.2	–	1.7	31.1	37.1	49.8	–	66.0	–	61.1	25.3	44.4	59.3	27.3
ProCST [5]	Trans.	84.3	87.7	86.1	87.6	41.1	87.9	50.7	74.7	–	6.1	42.6	54.2	62.5	–	61.4	–	55.5	47.2	53.3	61.4	22.6
DAFormer [6]	Trans.	84.5	88.4	86.0	87.2	40.7	89.8	50.0	73.2	–	6.5	41.5	54.6	61.7	–	53.2	–	55.0	48.2	53.9	60.9	22.8
FREDOM	Trans.	89.4	**89.3**	89.9	90.5	50.8	**93.7**	**57.3**	**79.4**	–	**9.3**	**48.8**	**60.1**	**68.1**	–	**66.0**	–	**65.1**	**51.6**	**62.3**	**67.0**	**22.0**
GTA5 → Cityscapes																						
CBST [15]	DL-V2	91.8	80.5	83.9	82.7	53.5	80.9	34.0	53.1	34.2	21.0	32.7	20.4	42.8	30.3	35.9	16.0	28.9	24.0	25.9	45.9	25.4
DACS [9]	DL-V2	89.9	87.9	88.0	84.5	39.7	88.8	38.5	67.2	44.0	39.5	30.7	52.8	34.0	45.7	50.2	0.0	46.4	35.8	27.3	52.1	25.4
CorDA [13]	DL-V2	94.7	87.6	87.6	**90.2**	**63.1**	**89.7**	40.2	66.7	47.0	40.6	30.7	51.6	56.0	48.9	57.5	0.0	47.8	35.9	39.8	56.6	24.8
AdvEnt [11]	DL-V2	89.9	81.6	83.9	83.7	36.5	77.1	28.5	57.4	34.0	25.2	29.2	22.4	23.3	29.4	39.1	1.5	32.3	27.9	28.4	43.8	26.4
MaxSquare [3]	DL-V2	89.4	82.1	85.3	84.6	43.0	78.2	30.3	63.0	39.4	21.3	30.5	24.0	33.5	36.4	43.0	5.5	34.7	22.9	34.7	46.4	25.7
IntraDA [7]	DL-V2	90.6	82.6	85.2	86.4	36.1	80.2	27.6	59.3	39.3	21.3	29.5	23.1	37.6	33.6	53.9	0.0	31.4	29.4	32.7	46.3	26.7
BiMaL [10]	DL-V2	**91.2**	82.7	85.4	86.6	39.6	80.8	29.6	59.7	44.0	25.2	29.4	25.5	36.8	38.5	47.6	1.2	34.3	30.4	34.0	47.3	25.9
SAC [1]	DL-V2	90.3	86.6	87.5	88.5	53.9	86.0	45.1	67.6	40.2	27.4	42.5	42.9	45.1	49.0	54.6	9.8	48.6	29.7	26.6	53.8	24.2
ProDA [14]	DL-V2	87.8	79.7	88.6	88.8	56.0	82.1	**45.6**	**70.7**	**45.2**	**44.8**	**46.3**	**53.5**	56.4	45.5	59.4	1.0	53.5	39.2	48.9	57.5	21.7
FREDOM	DL-V2	90.9	**87.8**	**88.6**	89.7	54.1	89.5	45.2	68.8	42.6	32.6	44.1	**57.1**	**58.1**	**58.4**	**62.6**	**55.3**	51.4	**40.0**	**47.7**	**61.3**	**19.1**
TransDA-S [4]	Trans.	92.9	88.2	89.6	91.4	59.1	94.1	47.6	74.3	42.0	32.0	42.5	39.2	51.4	54.0	58.0	44.4	57.6	45.3	48.3	60.6	20.8
TransDA [4]	Trans.	94.7	89.2	90.4	92.5	64.2	93.7	50.1	76.7	50.2	45.8	48.1	40.8	55.4	56.8	60.1	47.6	60.2	47.6	49.6	63.9	19.1
ProCST [5]	Trans.	95.8	89.8	90.2	92.3	69.6	93.0	49.8	72.2	50.3	45.0	55.8	63.3	63.1	72.2	78.8	65.1	56.8	44.9	56.4	68.7	17.1
DAFormer [6]	Trans.	95.7	89.4	89.9	92.3	70.2	92.5	49.6	72.2	47.9	48.1	53.5	59.4	61.8	74.5	78.2	65.1	55.8	44.7	55.9	68.3	17.3
FREDOM	Trans.	**96.7**	**90.9**	**91.6**	**94.1**	**74.8**	**94.4**	**57.5**	**78.4**	**52.1**	**49.0**	**58.1**	**71.4**	**68.9**	**83.9**	**85.2**	**72.5**	**63.4**	**53.1**	**62.8**	**73.6**	**15.8**

3.3. Qualitative Results

Figure 1 illustrates additional qualitative results of the SYNTHIA → Cityscapes experiments. In particular, we compare our results with AdvEnt [11], BiMaL [10], SAC [1], and DAFormer [6]. Overall, our approach produces better quality compared to prior methods. The predictions of segmentation maps of classes in minority groups have been improved and well-segmented compared to other methods.

References

[1] Nikita Araslanov, , and Stefan Roth. Self-supervised augmentation consistency for adapting semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3, 4

[2] Mark Chen, Alec Radford, Rewon Child, Jeff Wu, Heewoo Jun, Prafulla Dhariwal, David Luan, and Ilya Sutskever. Generative pretraining from pixels. *ICML*, 2020. 2

[3] Minghao Chen, Hongyang Xue, and Deng Cai. Domain adaptation for semantic segmentation with maximum squares loss. In *ICCV*, 2019. 3

[4] Runfa Chen, Yu Rong, Shangmin Guo, Jiaqi Han, Fuchun Sun, Tingyang Xu, and Wenbing Huang. Smoothing matters: Momentum transformer for domain adaptive semantic segmentation. *CoRR*, 2022. 3

[5] Shahaf Ettedgui, Shady Abu-Hussein, and Raja Giryes. Procst: Boosting semantic segmentation using progressive cyclic style-transfer, 2022. 3

[6] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. DAFormer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, 2022. 2, 3, 4

[7] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *CVPR*, 2020. 3

[8] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. 2

Figure 1. **SYNTHIA → Cityscapes**. Qualitative Results of Our Proposed Approach. Columns 1-6 are the results of AdvEnt [11], BiMaL [10], SAC [1], DAFormer [6], our FREDOM approach, and Ground Truth (Best view in color and 2× zoom).

[9] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1379–1389, 2021. 3

[10] Thanh-Dat Truong, Chi Nhan Duong, Ngan Le, Son Lam Phung, Chase Rainwater, and Khoa Luu. Bimal: Bijective maximum likelihood approach to domain adaptation in semantic scene segmentation. In *ICCV*, 2021. 3, 4

[11] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019. 3, 4

[12] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Mathieu Cord, and Patrick Pérez. Dada: Depth-aware domain adaptation in semantic segmentation. In *ICCV*, 2019. 3

[13] Qin Wang, Dengxin Dai, Lukas Hoyer, Luc Van Gool, and Olga Fink. Domain adaptive semantic segmentation with self-supervised depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 3

[14] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. *arXiv preprint arXiv:2101.10979*, 2021. 3

[15] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018. 3