Supplementary Materials for DODA: Data-oriented Sim-to-Real Domain Adaptation for 3D Semantic Segmentation

Runyu Ding[1*], Jihan Yang[1*], Li Jiang[2], and Xiaojuan Qi[1✉]

[1] The University of Hong Kong
[2] MPI for Informatics
{ryding,jhyang,xjqi}@eee.hku.hk lijiang@mpi-inf.mpg.de

Outline

This supplementary document is arranged as follows:

- Sec. S1 elaborates the visible range design in occlusion simulation of VSS;
- Sec. S2 illustrates visualization and analysis of TACM and other data-mixing methods;
- Sec. S3 presents the implementation details of benchmark setup for sim-to-real settings and cross-site settings;
- Sec. S4 presents the per-class results of tail cuboid over-sampling in TACM;
- Sec. S5 benchmarks DODA with other popular UDA methods on cross-site settings;
- Sec. S6 investigates DODA performance on 3D-FRONT → NYU-Depth V2, which focuses on the adaptation from simulation 3D to real RGBD;
- Sec. S7 analyzes the pseudo-label quality with VSS and TACM.
- Sec. S8 presents the qualitative results of S3DIS and ScanNet on sim-to-real settings.

S1 Visible Range Design

In this section, we elaborate the visible range design. Given the camera position v and the point of interest h, the maximum visible range R_v is determined by FOV configurations encompassing the horizontal viewing angle α_h, the vertical viewing angle α_v and the viewing mode η. Specifically, the horizontal visible range $R_v[xy]$ is determined by α_h as Eq. (1):

$$R_v[xy] = \left\{ p \mid \frac{(p_{xy} - v_{xy})^T (h_{xy} - v_{xy})}{||p_{xy} - v_{xy}||_2 ||h_{xy} - v_{xy}||_2} > \cos \frac{\alpha_h}{2} \right\}, \tag{1}$$

where the subscript xy stands for the coordinate vector projected onto the X-Y plane. As for the vertical visible range $R_v[z]$, it depends on α_v and η as shown in Fig. S1. Specifically, for the simplest fixed mode (η = fixed), it selects the visible

[*] equal contribution
[✉] corresponding author

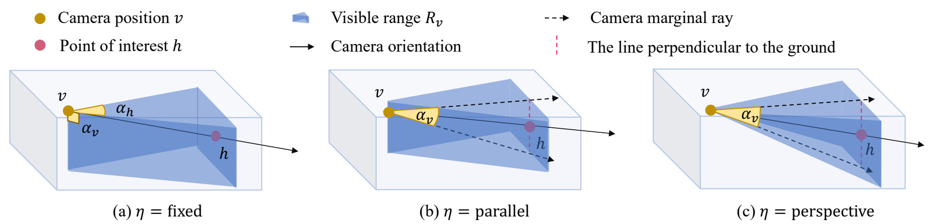

Fig. S1. An illustration of visible range with different viewing modes η. Note that for three modes, the definition of α_h is the same thus we only show it in the fixed mode.

range lower than the horizontal plane passing through camera v if the camera look downwards (see Fig. S1 (a)); otherwise range above the horizontal plane through v will be selected. In this regard, α_v is fixed at 90°. More flexibly, the parallel mode (η =parallel) decides the upper and lower bound of vertical visible range as the intersections of marginal rays and the line through h perpendicular to the ground (See Fig. S1 (b)). The perspective mode (η =perspective) further constrains the visible range into a rectangular pyramid bounded by the camera marginal rays (see Fig. S1 (c)), which is the most sophisticated and realistic camera projection process. Formally, the vertical range $R[v]$ with different viewing modes can be expressed as Eq. (2).

$$R_v[z] = \begin{cases} \{p \mid p_z > v_z\} \text{ if } h_z > v_z, \text{ otherwise } \{p \mid p_z < v_z\}, & \eta = \text{fixed}, \\ \{p \mid ||v_{xy} - h_{xy}|| \tan(\theta - \frac{\alpha_v}{2}) < (p_z - v_z) < ||v_{xy} - h_{xy}|| \tan(\theta + \frac{\alpha_v}{2})\}, & \eta = \text{parallel}, \\ \{p \mid ||v_{xy} - p_{xy}|| \tan(\theta - \frac{\alpha_v}{2}) < (p_z - v_z) < ||v_{xy} - p_{xy}|| \tan(\theta + \frac{\alpha_v}{2})\}, & \eta = \text{perspective}, \end{cases} \tag{2}$$

where θ is the camera pitch angle defined as $\arcsin(\frac{v_z - h_z}{||v-h||_2})$ and $||\cdot||$ denotes the L_2 distance. Finally we obtain the visible range R_v as the intersection of $R_v[xy]$ and $R_v[z]$.

S2 Visualization Comparison and Analysis between TACM and Other Data-mixing Methods

Even though we already present experimental results in Table 8 in the main paper, to better demonstrate the priority of our TACM among other data-mixing methods, we also show some visualization examples here. As shown in Fig S2, when scenes are mixed in Mix3D [8], it leads to ambiguity and loss of semantic cues since the neighboring relationship in local areas has been disrupted by mixed points from two domains. As for Copy-paste [4] and CutMix [16], they perturb a local area with randomly sampled patches or instances, which break the local context while introducing no disruptions of the broader context. In contrast, our TACM mixes scenes with the cuboid as the smallest unit, which preserves the local context while also bringing diversity to the global context by different cuboid combinations.

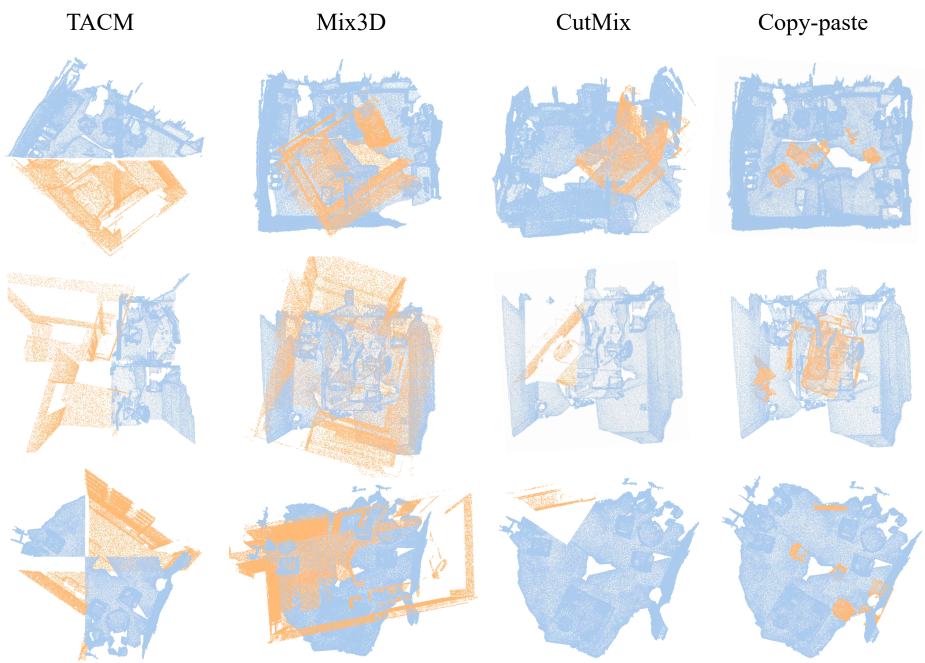

Fig. S2. An illustration of TACM examples along with other data-mixing methods. The yellow points are from source scenes and the blue points are from target scenes.

S3 Benchmark Setup

S3.1 Comparison of Large-scale Simulation Datasets.

In our sim-to-real adaptation benchmark, we select 3D-FRONT [2] as the source domain which contains 18,968 professionally designed rooms with 13,151 CAD 3D furniture objects from 3D-FUTURE [3]. Regarding other large-scale synthetic datasets, SUNCG [10] is not publicly available. Structured3D [17] does not provide interior 3D furniture objects that populate the scenes, which cannot be used as a source dataset without instance classes and layouts. OpenRoom [6] only contains 2.5K CAD models as the objects, which constrains its diversity. Hence, 3D-FRONT is a favorable choice with adequate scenes as well as professional layouts.

S3.2 Label Mapping.

Due to the different label spaces of datasets, we need to condense common categories for each adaptation task. We manually determine the category mapping relations according to the class names and representative shapes for each class in different datasets. The selected common classes and mapping relations for 3D-FRONT $\rightarrow$ ScanNet, 3D-FRONT $\rightarrow$ S3DIS, 3D-FRONT $\rightarrow$ NYU-Depth V2 and ScanNet $\leftrightarrows$ S3DIS are shown in Table S1, S2, S3 and S4, respectively.

S3.3 Implementation Details

Network Details. We validate DODA on the sparse-convolution-based U-Net backbone [5, 1], which is a popular and high-performance network on 3D segmentation tasks. The voxel size for point cloud voxelization is set to 2cm.
Training Details. In the pretrain stage, we train source data for 11k iterations with 32 batch size on 8 GPUs. SGD optimizer is employed with 0.9 momentum and 0.0001 weight decay. The learning rate is initialized as 0.005 without decay. For pseudo label generation, we set the pseudo label confidence threshold T to 0.7 for ScanNet and to per-class 30% for S3DIS, to achieve the highest performance. In the self-training stage, we fine-tune the pertrain model for 3.8k iterations on ScanNet and 0.6k iterations on S3DIS. The initial learning rate is set as 0.005 and decayed following the polynomial policy with 0.9 power. The same batch size and optimizer are utilized as in the pretrain stage. The loss trade-off factor λ is set as 0.5. During the two stages, commonly used augmentations are applied, in terms of rotation along vertical axis, flip, elastic distortion, jittering and point shuffling. All experiments are conducted on 8 NVIDIA GTX 2080 Ti GPUs.

For the hyper-parameters of VSS, the number of cameras n_v is set to 4 by default. We set the α_h as 180°, α_v as 90° and η as fixed for FOV configuration. The point jittering intensity δ_p is set as 0.01. For cuboid mixing in TACM, the permutation probability ρ_s and domain mixing probability ρ_m are both set as 0.5. The number of partitions (n_x, n_y, n_z) is set to (2,2,1) with partition perturbations δ_ϕ as 0.1. Thus a total of 4 cuboids are partitioned for each scene.

As for tail cuboid over-sampling, we typically set the tail cuboid queue size N_q as 256 and the number of tail classes n_r as 2. The least number of tail cuboids per scene u is set as 2.

Table S1. Label mapping for 3D-FRONT → ScanNet.

Class	ScanNet	3D-FRONT
wall	wall	wallInner; wallOuter; baseboard; wallTop; customizedBackgroundModel; wallbottom; customizedFeatureWall; extrusionCustomizedBackgroundModel
floor	floor	floor
cabinet	cabinet	children cabinet; wardrobe; sideboard/side cabinet/ console table; wine cabinet; wardrobe; TV stand; drawer chest/corner cabinet
bed	bed	king-size bed; bunk bed; bed frame; single bed; kids bed
chair	chair	dining chair; lounge chair/cafe chair/office chair; dressing chair; classic Chinese chair; barstool
sofa	sofa	three-seat/multi-seat sofa; armchair; loveseat sofa; L-shapped sofa; lazy sofa; chaise longue sofa
table	table	coffee table; round end table; dressing table; dining table
door	door	door; pocket
window	window	window; baywindow
bookshelf	bookshelf	bookcase/jewelry armoire
desk	desk	desk

S3.4 UDA Baselines.

We reproduce 7 popular 2D UDA methods and 1 3D outdoor UDA method as our baselines, encompassing MCD [9], AdaptSegNet [11], CBST [18], MinEnt [12], AdvEnt [12], Noisy Student [14] APO-DA [15] and SqueezeSegV2 [13]. Similar to DODA, for each baseline, we adopt a sparse-convolution-based U-Net backbone [5, 1] and a linear fully-connected point-wise classification head as the overall segmentation network. Besides, some modifications are made for adapting to the 3D vision task as below. For MCD, the U-Net is used as the generator and the point-wise classification head is used as two-branch classifiers. For AdaptSegNet, we employ its single-level adversarial training performed on the output space. Since the output of the segmentation network is the point-wise predictions, we implement the discriminator as a PointNet-like neural network with 3-layer shared MLP and point random downsampling. For MinEnt, we perform point-wise entropy minimization on target data. For AdvEnt, the same discriminator is utilized as in AdaptSegNet to discriminate outputs from different domains. For APO-DA, we also use the UNet as the generator and only attack the linear

Table S2. Label mapping for 3D-FRONT → S3DIS.

Class	S3DIS	3D-FRONT
wall	wall	wallInner; wallOuter; baseboard; wallTop; customizedBackgroundModel; wallBottom; customizedFeatureWall; extrusionCustomizedBackgroundModel
floor	floor	floor
chair	chair	dining chair; lounge chair/cafe chair/office chair; dressing chair; classic Chinese chair; barstool
sofa	sofa	three-seat/multi-seat sofa; armchair; loveseat sofa; L-shapped sofa; lazy sofa; chaise longue sofa
table	table	coffee table; round end table; dressing table; dining table; desk
door	door	door; pocket
window	window	window; baywindow
bookcase	bookshelf	bookcase/jewelry armoire
ceiling	ceiling	customizedCeiling; smartCustomizedCeiling; ceiling; extrusionCustomizedCeilingModel
beam	beam	beam
column	column	column

Table S3. Label mapping for 3D-FRONT → NYU-Depth V2.

Class	NYU-Depth V2	3D-FRONT
wall	wall	wallInner; wallOuter; baseboard; wallTop; customizedBackgroundModel; wallBottom; customizedFeatureWall; extrusionCustomizedBackgroundModel
floor	floor	floor
cabinet	cabinet	children cabinet; wardrobe; sideboard/side cabinet/ console table; wine cabinet; wardrobe; TV stand; drawer chest/corner cabinet
bed	bed	king-size bed; bunk bed; bed frame; single bed; kids bed
chair	chair	dining chair; lounge chair/cafe chair/office chair; dressing chair; classic Chinese chair; barstool
sofa	sofa	three-seat/multi-seat sofa; armchair; loveseat sofa; L-shapped sofa; lazy sofa; chaise longue sofa
table	table	coffee table; round end table; dressing table; dining table
door	door	door; pocket
window	window	window; baywindow
bookshelf	bookshelf	bookcase/jewelry armoire
desk	desk	desk
ceiling	ceiling	customizedCeiling; smartCustomizedCeiling; ceiling; extrusionCustomizedCeilingModel

Table S4. Label mapping for ScanNet → S3DIS and S3DIS → ScanNet.

Class	ScanNet	S3DIS
wall	wall	wall
floor	floor	floor
chair	chair	chair
sofa	sofa	sofa
table	table	table
door	door	door
window	window	window
bookshelf	bookshelf	bookcase

classification head to generate point-wise adversarial features. As for the self-training pipeline including CBST and Noisy Student, no other modifications are needed. For the 3D baseline SqueezeSegV2, without official implementations, we self-implement the geodesic alignment and domain calibration modules for our indoor UDA task. The intensity rendering module is discarded since it is specified for outdoor data.

S4 Per-class Results of Tail Cuboid Over-sampling

We present per-class results of Tail Cuboid Over-Sampling (TCOS) on 3D-FRONT → ScanNet in Table S5 to demonstrate that the performance gain mainly comes from boosting tail categories. From target pseudo label statistics, the tail classes for this setting are bookshelf and desk with sampling ratios around 25% and 75%, respectively. For desk, the significant improvements around 6% verifies the effectiveness of our method in addressing the long-tail issue in pseudo labels.

Table S5. Supplementary adaptation results of 3D-FRONT → ScanNet in terms of mIoU (%). We indicate the best adaptation results in **bold**. † denotes DODA results without tail cuboid over-sampling.

Method	mIoU	wall	floor	cab.	bed	chair	sofa	table	door	wind.	bksf.	desk
DODA w/o TCOS†	50.55	72.63	**93.98**	**28.11**	**65.88**	71.43	53.17	57.40	**08.53**	21.76	**57.10**	26.09
DODA	**51.42**	**72.71**	93.86	27.61	64.31	**71.64**	**55.30**	**58.43**	08.21	**24.95**	56.49	**32.06**

S5 Experimental Results on Cross-site Adaptation Tasks

In real-to-real cross-site adaptation tasks, scenes collected from different sites or room types suffer considerable domain discrepancies. To verify the effectiveness of TACM in bridging the real-world domain gaps, we compare DODA (only

TACM) with other popular UDA methods on ScanNet → S3DIS and S3DIS → ScanNet in Table S6 and Table S7, respectively. Results show that DODA (only TACM) outperforms other methods by a large margin around 6% ∼ 16% on ScanNet → S3DIS and 4% ∼ 18% on S3DIS → ScanNet. It verifies that our TACM can serve as a general module to eliminate source context bias through target cuboid-level contextual patterns complement.

Besides, to evaluate unsupervised domain adaptation methods, we argue that S3DIS is unsuitable as a source dataset since the per-class results of DODA on real-to-real S3DIS → ScanNet are even worse than its counterpart on the sim-to-real 3D-FRONT → ScanNet setting (see Table 1 of the main paper). Although real-to-real adaptation theoretically shows smaller domain gaps than sim-to-real settings, S3DIS is rather simple with a small sample size and limited diversity as its scenes are collected only in three buildings of mainly office and educational use, thus resulting in poor performance of adaptation. It illustrates the importance of carefully selecting real-world datasets as the source domain. Simulated datasets, on the other hand, can be a consistently appealing choice as a source domain with arbitrarily large size, diverse samples and free annotations.

Table S6. Adaptation results of ScanNet → S3DIS in terms of mIoU (%). We indicate the best adaptation result in **bold**. † denotes the self-training results with TACM based on CBST.

Method	mIoU	wall	floor	chair	sofa	table	door	wind.	bkcase.
Source Only	54.09	64.38	94.39	76.15	25.46	70.55	28.98	28.52	**44.31**
MCD [9]	49.83	61.38	95.47	73.51	32.04	**75.24**	36.95	08.01	16.02
AdaptSegNet [11]	50.28	67.75	94.47	69.13	24.77	67.71	36.32	13.54	28.57
CBST [18]	60.13	68.66	**96.02**	84.61	55.04	63.80	33.47	35.61	43.84
MinEnt [12]	55.31	71.31	94.70	68.10	39.86	68.23	35.98	22.03	42.24
AdvEnt [12]	49.86	68.83	93.87	67.37	20.77	68.11	32.67	13.74	33.50
Noisy student [14]	58.82	66.76	95.84	83.56	52.05	64.39	36.36	37.51	34.08
APO-DA [15]	53.47	68.70	95.62	76.69	43.01	70.53	26.22	11.63	35.37
DODA (only TACM)†	**66.52**	**73.81**	95.94	**85.82**	**70.71**	64.64	**42.93**	**48.25**	42.09
Oracle	72.51	84.89	97.63	83.72	55.26	81.47	53.94	44.61	78.55

S6 Experimental Results on Sim 3D → Real RGBD task

S6.1 Datasets.

NYU-Depth V2 [7] is a popular RGBD dataset for semantic segmentation. It contains 1,449 densely annotated RGBD images, *i.e.* 795 training samples and 654 validation samples. Each image has a resolution of 640×480, which can be back-projected to a 3D point cloud containing 3077,200 points. It provides 40 semantic categories.

Table S7. Adaptation results of S3DIS → ScanNet in terms of mIoU (%). We indicate the best adaptation result in **bold**. † denotes the self-training results with TACM based on Noisy Student.

Method	mIoU	wall	floor	chair	sofa	table	door	wind.	bksf.
Source Only	33.43	37.87	84.01	55.26	18.32	36.15	11.43	08.58	15.81
MCD [9]	30.65	39.50	92.76	43.74	00.00	40.57	09.67	06.03	12.88
AdaptSegNet [11]	36.14	58.48	91.61	35.47	21.35	44.23	07.18	09.17	21.62
CBST [18]	43.08	45.43	90.11	67.53	35.48	56.51	**16.94**	09.65	22.97
MinEnt [12]	39.40	58.11	90.31	51.18	24.86	44.20	08.10	10.27	28.19
AdvEnt [12]	38.09	58.83	90.24	41.73	28.96	40.68	10.58	08.11	25.59
Noisy student + [14]	44.81	55.61	92.75	65.72	37.77	57.77	12.54	**15.25**	21.09
APO-DA [15]	38.67	63.85	90.18	49.86	22.34	41.89	06.44	04.64	**30.15**
DODA (only TACM)	**48.47**	**65.03**	**94.25**	**69.23**	**43.13**	**58.79**	03.58	13.86	29.91
Oracle	80.06	86.78	96.02	89.98	84.24	82.15	51.19	64.99	85.16

S6.2 Main Results.

In the main paper, our experiments focus on the sim-to-real adaptation with target scenes reconstructed by RGBD sequences. However, in real-world scenarios, the real scene can be a single RGBD image captured by the depth camera without reconstructions. Therefore, we also investigate the performance of DODA in such a more challenging setting, *i.e.* sim 3D → real RGBD. As demonstrated in Table S3, DODA significantly outperforms source only by around 14.3% and improves CBST by around 8.5%, largely reducing the cross-modal gaps between 3D-FRONT and NYU-Depth V2. Even only equipping source only with VSS, our DODA (only VSS) also shows its superiority, obtaining 6.3% and 0.6% gains compared to source only and CBST separately, which demonstrates the effectiveness of VSS in alleviating the point pattern gaps between simulation 3D and real RGBD. Compared to DODA w/o TACM, TACM further enhances the performance by around 3.4%, largely bridging the context gaps.

Table S8. Adaptation results of 3D-FRONT [2] → NYU-Depth V2 in terms of mIoU (%). We indicate the best adaptation result in **bold**. † denotes our pretrain generalization results only with VSS.

Method	mIoU
Source Only	17.80
CBST [18]	23.58
DODA (only VSS)†	24.14
DODA w/o TACM	28.74
DODA	**32.12**
Oracle	52.88

S7 Analysis of Pseudo label quality

Self-training relies on both pseudo label accuracy and covering ratio (Eq. (3)) for diversity. As shown in Table S9, DODA (only VSS) generates pseudo labels with around 15.6% higher mIoU and 7.7% larger label covering ratio compared to source only, which benefits the follow-up self-training stage. Besides, TACM also improves the pseudo label quality after the first self-training round by about 3.6% mIoU and 0.5% covering ratio, which is supposed to further boost the iterative self-training if applied.

$$\text{covering ratio} = \frac{\#\text{ pseudo-labeled points}}{\#\text{ all points}} \times 100\% \tag{3}$$

Table S9. Results of pseudo label quality with threshold $T = 0.7$.

Method	pseudo label	
	mIoU	covering ratio (%)
Source Only	35.16	59.85
DODA (only VSS)	50.73	67.54
DODA (w/o TACM)	53.24	81.51
DODA	56.85	82.05

S8 Visualization

We provide some qualitative results of DODA on sim-to-real adaptation tasks of 3D-FRONT → ScanNet and 3D-FRONT → S3DIS as illustrated in Fig. S3. Compared to self-training baselines, our DODA can segment instances better and generate more accurate and smooth predictions.

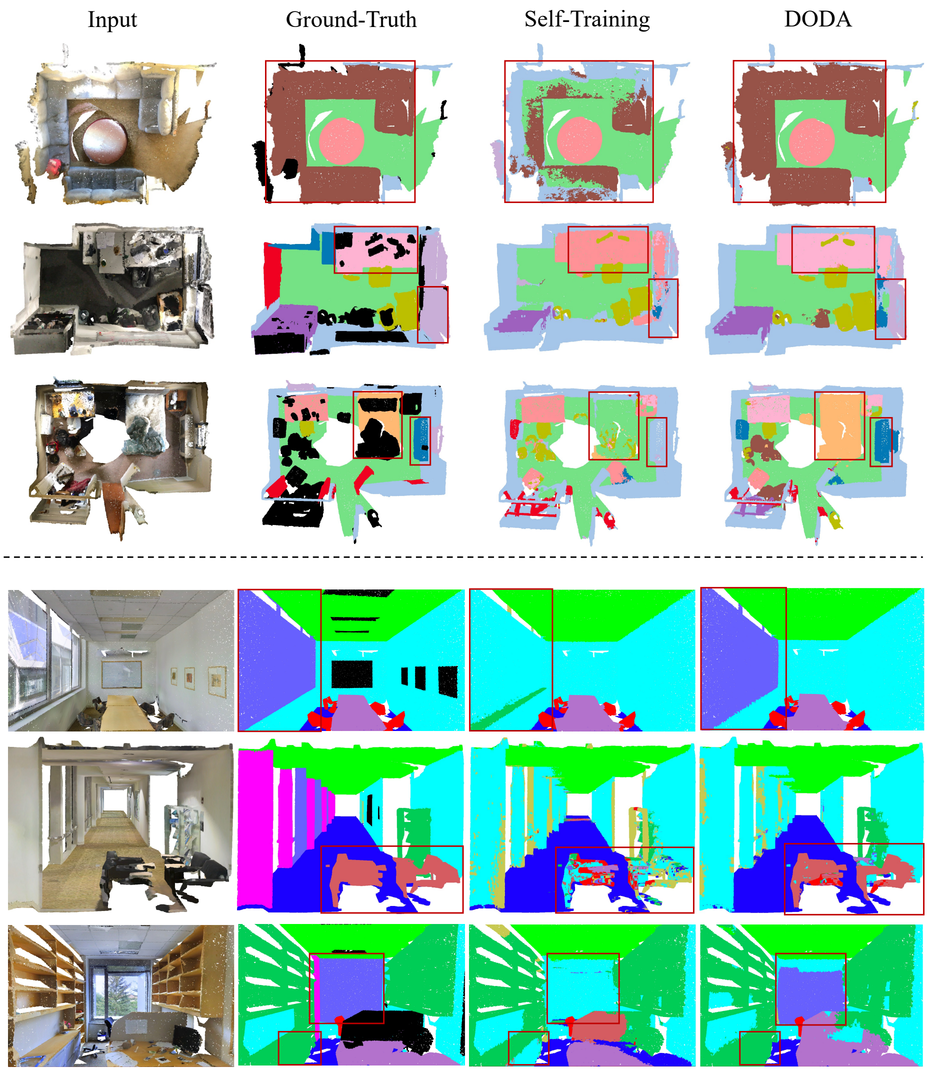

Fig. S3. Qualitative results of 3D-FRONT → ScanNet (top) and 3D-FRONT → S3DIS (bottom). Note that the third column is the prediction of self-training baselines, *i.e.* Noisy Student for ScanNet and CBST for S3DIS. The red bounding boxes indicate the specific areas where our DODA significantly outperforms self-training baselines.

References

1. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3075–3084 (2019)
2. Fu, H., Cai, B., Gao, L., Zhang, L.X., Wang, J., Li, C., Zeng, Q., Sun, C., Jia, R., Zhao, B., et al.: 3d-front: 3d furnished rooms with layouts and semantics. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10933–10942 (2021)
3. Fu, H., Jia, R., Gao, L., Gong, M., Zhao, B., Maybank, S., Tao, D.: 3d-future: 3d furniture shape with texture. International Journal of Computer Vision pp. 1–25 (2021)
4. Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.Y., Cubuk, E.D., Le, Q.V., Zoph, B.: Simple copy-paste is a strong data augmentation method for instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2918–2928 (2021)
5. Graham, B., Engelcke, M., Van Der Maaten, L.: 3d semantic segmentation with submanifold sparse convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9224–9232 (2018)
6. Li, Z., Yu, T.W., Sang, S., Wang, S., Song, M., Liu, Y., Yeh, Y.Y., Zhu, R., Gundavarapu, N., Shi, J., et al.: Openrooms: An open framework for photorealistic indoor scene datasets. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7190–7199 (2021)
7. Nathan Silberman, Derek Hoiem, P.K., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV (2012)
8. Nekrasov, A., Schult, J., Litany, O., Leibe, B., Engelmann, F.: Mix3d: Out-of-context data augmentation for 3d scenes. In: 2021 International Conference on 3D Vision (3DV). pp. 116–125. IEEE (2021)
9. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3723–3732 (2018)
10. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.: Semantic scene completion from a single depth image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1746–1754 (2017)
11. Tsai, Y.H., Hung, W.C., Schulter, S., Sohn, K., Yang, M.H., Chandraker, M.: Learning to adapt structured output space for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7472–7481 (2018)
12. Vu, T.H., Jain, H., Bucher, M., Cord, M., Pérez, P.: Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2517–2526 (2019)
13. Wu, B., Zhou, X., Zhao, S., Yue, X., Keutzer, K.: Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 4376–4382. IEEE (2019)
14. Xie, Q., Luong, M.T., Hovy, E., Le, Q.V.: Self-training with noisy student improves imagenet classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10687–10698 (2020)

15. Yang, J., Xu, R., Li, R., Qi, X., Shen, X., Li, G., Lin, L.: An adversarial perturbation oriented domain adaptation approach for semantic segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 12613–12620 (2020)
16. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6023–6032 (2019)
17. Zheng, J., Zhang, J., Li, J., Tang, R., Gao, S., Zhou, Z.: Structured3d: A large photo-realistic dataset for structured 3d modeling. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16. pp. 519–535. Springer (2020)
18. Zou, Y., Yu, Z., Vijaya Kumar, B., Wang, J.: Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In: European Conference on Computer Vision. pp. 289–305 (2018)