Cross-Modal and Cross-Domain Knowledge Transfer for Label-Free 3D Segmentation

Jingyu Zhang[1], Huitong Yang[2], Dai-Jie Wu[2], Jacky Keung[1], Xuesong Li[4], Xinge Zhu[3(✉)], and Yuexin Ma[2(✉)]

[1] City University of Hong Kong, Hong Kong SAR, China
[2] School of Information Science and Technology, ShanghaiTech University, Shanghai, China
mayuexin@shanghaitech.edu.cn
[3] Chinese University of Hong Kong, Hong Kong SAR, China
zhuxinge123@gmail.com
[4] College of Science, Australian National University, Canberra, Australia

Abstract. Current state-of-the-art point cloud-based perception methods usually rely on large-scale labeled data, which requires expensive manual annotations. A natural option is to explore the unsupervised methodology for 3D perception tasks. However, such methods often face substantial performance-drop difficulties. Fortunately, we found that there exist amounts of image-based datasets and an alternative can be proposed, *i.e.*, transferring the knowledge in the 2D images to 3D point clouds. Specifically, we propose a novel approach for the challenging cross-modal and cross-domain adaptation task by fully exploring the relationship between images and point clouds and designing effective feature alignment strategies. Without any 3D labels, our method achieves state-of-the-art performance for 3D point cloud semantic segmentation on SemanticKITTI by using the knowledge of KITTI360 and GTA5, compared to existing unsupervised and weakly-supervised baselines.

Keywords: Point Cloud Semantic Segmentation · Unsupervised Domain Adaptation · Cross-modal Transfer Learning

1 Introduction

Accurate depth information of large-scale scenes captured by LiDAR is the key to 3D perception. Consequently, LiDAR point cloud-based research has gained popularity. However, existing learning-based methods [10,11,37] demand extensive training data and significant labor power. Especially for the 3D segmentation task, people need to assign class labels to massive amounts of points. Such high cost really hinders the progress of related research. Therefore, relieving the annotation burden has become increasingly important for 3D perception.

J. Zhang and H. Yang – Equal contribution.

Q. Liu et al. (Eds.): PRCV 2023, LNCS 14427, pp. 465–477, 2024.
https://doi.org/10.1007/978-981-99-8435-0_37

Recently, plenty of label-efficient methods have been proposed for semantic segmentation. They try to simplify the labeling operation by using scene-level or sub-cloud-level weak labels [24], or utilizing clicks or line segments as object-level or class-level weak labels [19,20]. However, these methods still could not get rid of the dependence on 3D annotations. Actually, annotating images is much easier due to the regular 2D representation. We aim to transfer the knowledge learned from 2D data with its annotations to solve 3D tasks. However, most related knowledge-transfer works [12,23,31] focus on the domain adaption under the same input modality, which is difficult to adapt to our task. Although recent works [27,35] begin to consider distilling 2D priors to 3D space, they use synchronized and calibrated images and point clouds and leverage the physical projection between two data modalities for knowledge transfer, which limits the generalization capability (Fig. 1).

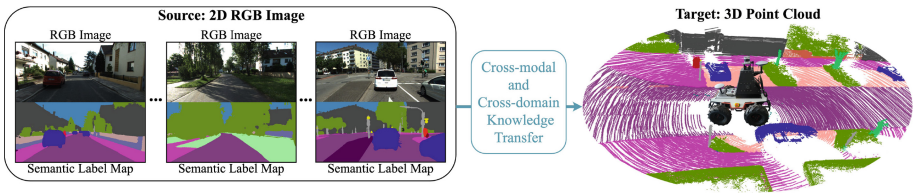

Fig. 1. Our method performs 3D semantic segmentation by transferring knowledge from RGB images with semantic labels from arbitrary traffic datasets.

In this paper, we propose an effective solution for a novel and challenging task, i.e., cross-modal and cross-domain 3D semantic segmentation for LiDAR point cloud based on unpaired 2D RGB images with 2D semantic annotations from other datasets. However, images and point clouds have totally different representations and features, making it not trivial to directly transfer the priors. To deal with the large discrepancy between the two domains, we perform alignment at both the data level and feature level. For data level (i.e., data representation level), we transform LiDAR point clouds to range images by spherical projection to obtain a regular and dense 2D representation as RGB images. For the feature level, considering that the object distributions and relative relationships among objects are similar in traffic scenarios, we extract instance-wise relationships from each modality and align their features from both the global-scene view and local-instance view through GAN. In this way, our method makes 3D point cloud semantic features approach 2D image semantic features and further benefits the 2D-to-3D knowledge transfer. To prove the effectiveness of our method, we conduct extensive experiments on KITTI360-to-SemanticKITTI and GTA5-to-SemanticKITTI. Our method achieves state-of-the-art performance for 3D point cloud semantic segmentation without any 3D labels.

2 Related Work

LiDAR-based Semantic Segmentation: Lidar-based 3D semantic segmentation [5,21] is crucial for autonomous driving. There are several effective representation strategies, such as 2D projection [7,33] and cylindrical voxelization [37]. However, most methods are in the full-supervision manner, which requires accurate manual annotations for every point, leading to high cost and time consumption. Recently, some research has been concentrated on pioneering labeling and training methods associated with weak supervision to mitigate the annotation workload. For instance, LESS [19] developed a heuristic algorithm for point cloud pre-segmentation and proposed a label-efficient annotation policy based on pre-segmentation results. Scribble-annotation [30] and label propagation strategies [28] were also introduced. However, these methods still require expensive human annotation on the point cloud. Several research studies [13,27,35] have explored complementary information to enhance knowledge transfer from images and facilitate 2D label utilization. Our work discards the need for any 3D labels by transferring scene and semantic knowledge from 2D data to 3D tasks, which reduces annotation time and enhances practicality for real-world scenarios.

Unsupervised Domain Adaptation (UDA): UDA techniques usually transfer the knowledge from the source domain with annotations to the unlabeled target domain by aligning target features with source features, which is most related to our task. For many existing UDA methods, the source and target inputs belong to the same modality, such as image-to-image and point cloud-to-point cloud domain adaptation. For image-based UDA, there are several popular methodologies for semantic segmentation, including adversarial feature learning [12,31,34], pseudo-label self-training [9,36] and graph reasoning [17]. For point cloud-based UDA, it is more challenging due to the sparse and unordered point representation. Recent works [2,23] have proposed using geometric transformations, feature space alignment, and consistency regularization to align the source and target point clouds. In addition, xMUDA [13] utilizes synchronized 2D images and 3D point clouds with 3D segmentation labels as the source dataset to assist 3D semantic segmentation. However, xMUDA assumes all modalities to be available during both training and testing, where we relax their assumptions by requiring no 3D labels, no synchronized multi-modal datasets, and during inference, no RGB data. Therefore, existing methods are difficult to extend to solve our cross-domain and cross-modality problem. To the best of our knowledge, we are the first to explore the issue and propose an effective solution.

3 Methodology

3.1 Problem Definition

For the 3D semantic segmentation task, our model associates a class label with every point in a 3D point cloud $X \in R^{N\times 4}$, where N denotes the total number of points. Location information in 3D coordinates, and the reflectance intensity, are

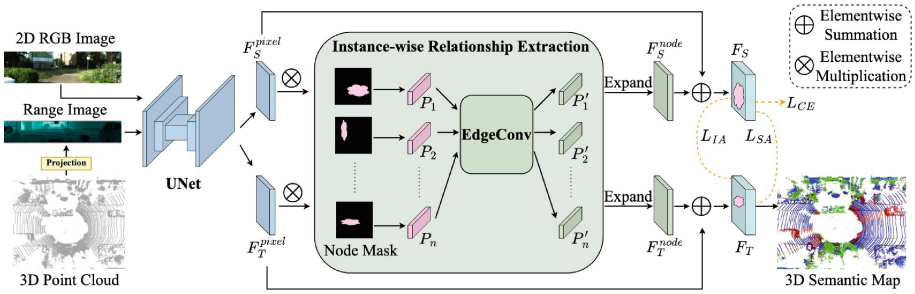

Fig. 2. The pipeline of our method with feature extraction module, instance-wise relationship extraction module, and feature alignment module (L_{IA}, L_{SA}).

provided for each point, denoted by $\{x, y, z, i\}$. Without any 3D annotations, we conduct 3D segmentation by using only 2D semantic information on RGB images from arbitrary datasets in similar scenarios. Unlike existing UDA methods that perform adaptation under the same data modality, our task is more challenging because our source and target data are from different domains and different modalities, and can be named as *Unsupervised Domain-Modality Adaptation (UDMA)*. UDMA transfers knowledge from 2D source domain (images $I_S \in R^{H \times W \times 3}$ with semantic labels $Y_S \in R^{H \times W \times C}$, where H and W are height and width of the image, and C is the number of classes) to 3D target domain (point clouds X_T). For UDMA 3D segmentation, we have $\{I_S, Y_S, X_T\}$ as the input data for training. During inference, only 3D point clouds (projected to range images) are input to the model to produce 3D predictions of the shape R^N (i.e., no RGB data are needed).

3.2 Method Overview

As shown in Fig. 2, our model consists of three important modules: Feature Extraction, Instance-wise Relationship Extraction, and Feature Alignment.

Feature Extraction: This module is used to encode various input data to a certain feature representation. We apply UNet [26] as the feature extraction backbone, where the size of the output feature map retains the same as the input image size, which allows us to obtain pixel-level dense semantic features.

Instance-Wise Relationship Extraction (IRE): This module carries the functionality of learning object-level features that encode local neighborhood information, therefore enhancing object-to-object interaction. The learned features are complementary to the mutually-independent pixel-level features learned by Feature Extraction module. Encoding more spatial information rather than domain- or modality-specific information enhances the efficiency of feature alignment.

Feature Alignment: To solve UDMA tasks (defined in Sect. 3.1), we need to narrow down the 2D-3D domain gap to transfer label information. We achieve

this through adversarial learning [8] in both scene level and instance level, which learns to project inputs from different modalities to a common feature space and output domain- and modality-independent features.

3.3 Data Pre-processing

LiDAR Point Cloud Pre-segmentation: Point clouds often contain valuable semantic information that can be used as prior knowledge without the need for

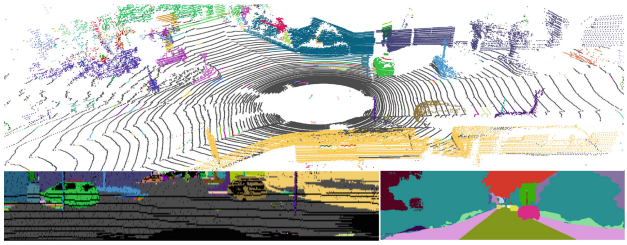

Fig. 3. Visualization of Data-Pre-processing: Top: Point cloud pre-segmentation result. Bottom left: the range image after the spherical projection of the point cloud pre-segmentation. Bottom right: semantic labels in the RGB image. (Color figure online)

any training. We utilize a pre-segmentation method proposed by LESS [19] to group points into components such that each component is of high purity and generally contains only one object. The top image in Fig. 3 displays the pre-segmentation result. The approach involves using the RANSAC [6] for ground points detection. For non-ground points, an adaptive distance threshold is utilized to identify points belonging to the same component. We then utilize component information, such as mean or eigenvalues of point coordinates, to identify component classes. By using this heuristic approach we can successfully identify common classes (e.g., car, building, vegetation) with a high recall rate, and rare classes (e.g., person, bicycle) with a low recall rate. Due to the limited usefulness of low recall rate classes, we finally use three prior categories: car, ground (includes road, sidewalk, and terrain), and wall (building and vegetation) to assist the instance feature alignment in our method. In future work, we will explore the combination of learning-based and heuristic algorithms to improve the quantity and quality of the component class identification. For example, feature matching between unidentified and already identified components.

Range Image Projection: 2D and 3D data have huge a gap in their representations, e.g., structure, dimension, or compactness, which brings difficulties in knowledge transfer. To bridge the gap, we leverage another representation of 3D point cloud, range image [15,21,33]. 2D range images follow similar data representation as RGB (both represented in $height \times width$). As shown in the

bottom two plots in Fig. 3, the road (in gray and olive colors) is always located at the bottom of the picture with cars on top in both range and RGB images. The range image is generated by spherically projecting [33] 3D points onto a 2D plane:

$$\begin{aligned} u &= \frac{1}{2} \cdot (1 - arctan(y \cdot x^{-1})/\pi) \cdot U \\ v &= (1 - (\arcsin(z \cdot r^{-1}) + f_{down}) \cdot f^{-1}) \cdot V, \end{aligned} \tag{1}$$

where $u \in 1, ..., U$ and $v \in 1, ..., V$ indicate the 2D coordinate of the projected 3D point (x, y, z). $r = \sqrt{x^2 + y^2 + z^2}$ is the range of each LiDAR point. We set $U = 2048$ and $V = 64$, respectively, which is ideal for Velodyne HDL-64E laser scan from SemanticKITTI dataset [1]. 2D range images retain the depth information brought by LiDAR and provide rich image-level scene structure information. This helps in scene alignment. During training, we use $[r, r, i]$ for three input channels of range images, where r and i denote the range value defined above and the reflectance intensity, respectively.

3.4 Instance-Wise Relationship Extraction

Based on the consideration that traffic scenarios share similar object location distribution and object-object interaction, we extract relationship features by proposing a novel graph architecture connecting instances as graph nodes. Therefore, it enhances the feature information with object-object interaction details and benefits the following feature alignment. Our Instance-wise Relationship Extraction (IRE) consists of two parts: Instance Node Construction and Instance-level Correlation.

Instance Node Construction: With the motivation of learning object-object relationships in an image, we regard every instance as a node in the input image instead of the whole class as a node. For range images, we use pre-segmentation components as nodes, while for RGB images, we use ground truth to create node masks during training. Initial node descriptors P_i are computed as follows:

$$Agg(F^{pixel} \otimes Mask_i) = P_i, \qquad \forall i = 1, ..., n \tag{2}$$

The operation is the same for source and target data. F^{pixel} represents the pixel-level dense feature map extracted by UNet, with the shape $R^{H \times W \times d}$ for source images (F_S^{pixel}) and $R^{V \times U \times d}$ for target images (F_T^{pixel}). Boolean matrices $Mask_i$ denote the binary node mask for the i^{th} component, where the value 1 denotes the pixel belonging to this component. The mask has the shape $H \times W$ in the source domain and $V \times U$ in the target domain. We denote n by the total number of nodes in the current input image. With element-wise matrix multiplication operation $\otimes$, we obtain m_i nonzero pixels features for each node, with shape $R^{m_i \times d}, i = 1, ..., n$. We apply an aggregation function Agg to obtain a compact node descriptor $P_i \in R^d$. Agg is composed of a pooling layer followed by a linear layer. Then, node descriptors are forwarded to EdgeConv to extract instance-level correlation.

Instance-Level Correlation: This submodule aims to densely connect the instance-level graph nodes and enhance recognizable location information for object interaction. After the Instance Node Construction, we embed local structural information into each node feature by learning a dynamic graph $G = (V, E)$ [32] in EdgeConv, where $V = \{1, ..., n\}$ and $E \subseteq V \times V$ are the vertices and edges. We build G by connecting an edge between each node P_i and its k-nearest neighbors (KNN) in R^d.

$$P_i' = \underset{\{j|(i,j)\in E\}}{\square} f_\Theta(P_i, P_j) \tag{3}$$

f_Θ learns edge features between nodes with learnable parameters Θ. All local edge features of P_i are then aggregated through an aggregation operation $\square$ (e.g., max or sum) to produce $P_i' \in R^d$ which is an enhanced descriptor for node i that contains rich local information from neighbors. Instance-level correlation helps the model extract domain- and modality-independent features by focusing more on spatial-distribution features between instances. We then expand the instance features by replicating each P_i' for m_i times to obtain node-level feature map F^{node} that is of the same size as F^{pixel} (note that we omit the subscript here since the operation is the same for source and target data). We concatenate pixel-wise feature F^{pixel} with instance-wise feature F^{node} to form F, which keeps both dense individual information and local-neighborhood information to assist the feature alignment in the next section.

3.5 Feature Alignment

To transfer the knowledge obtained from RGB images to range images, we make the feature of range images approach that of RGB images from two aspects: the global scene-level alignment and local instance-level alignment.

Scene Feature Alignment (SA): Traffic scenarios in range and RGB images follow similar spatial distribution and object-object interaction (e.g., cars and buildings are located above the road). Therefore, we align the global scene features for two modalities. Specifically, we implement the unsupervised alignment through Generative Adversarial Networks (GAN) [8]. Adversarial training of the segmentation network can decrease the feature distribution gap between the source and target domain. Given the ground truth source labels, we train our model with the objective of producing source-like target features. In this way, we can maximize the utilization of low-cost RGB labels by transferring them to the target domain. This can be realized by two sub-process [22]: train the generator (i.e., the segmentation network) with target data to fool the discriminator; train the discriminator with both target and source data to discriminate features from two domains. In Scene Feature Alignment, we use one main discriminator to discriminate global features of the entire scene.

Instance Feature Alignment (IA): Accurate classification for each point is critical for 3D semantic segmentation. We further design fine-grained instance-

level feature alignment for two domains to improve the class recognition performance. To perform the instance-level alignment, we build extra category-wise discriminators to align source and target instances features. For target range images, as no labels are available, we use prior knowledge described in Sect. 3.3 to filter prior categories, while for RGB we use ground truth. Then, we compute target features and align them with the corresponding source features of each prior category.

3.6 Loss

Our model is trained with three loss functions, with one supervised Cross-Entropy loss (CE loss) on source domain and two unsupervised feature alignment losses. Formally, we compute the CE loss for each pixel (h, w) in the image and sum them over.

$$L_{CE}(I_S, Y_S) = -\sum_h \sum_w \sum_c Y_S^{(h,w,c)} \log P_S^{(h,w,c)}, \tag{4}$$

where $h \in \{1, ..., H\}$, $w \in \{1, ...W\}$, and class $c \in \{1, ...C\}$. P_S is the prediction probability map of the source image I_S. For Scene Feature Alignment (SA), the adversarial loss is imposed on global target and source features F_T, F_S:

$$L_{SA}(F_S, F_T; G, D) = -\log D(F_T) - \log D(F_S) - \log(1 - D(F_T)). \tag{5}$$

The output of discriminator $D(\cdot)$ is the domain classification, which is 0 for target and 1 for source. In SA, we use one main discriminator to discriminate range and RGB scene global features. For Instance Feature Alignment (IA),

$$L_{IA}(F_S^E, F_T^E; G, D^E) = \sum_{e=1}^{E} -y_T^e \log D^e(F_T^e) - y_S^e \log D^e(F_S^e) - y_T^e \log(1 - D^e(F_T^e)), \tag{6}$$

where F_T^e and F_S^e are the target and source features for each prior category e. $y_S^e = 1$ if category e occurs in the current source image, otherwise $y_S^e = 0$, similarly for y_T^e. Therefore, we only compute losses for the categories that are present in the current image.

We use fine-tuning to further improve the performance. We apply weak label loss [19] (Eq. 7) for the prior categories that contain more than one class and CE loss $L_{CE}(X_T^e, Y_T^e)$ for the prior category that has fine-grained class identification, where $X_T^e \in R^{1 \times n_e \times 3}$ and $Y_T^e \in R^{1 \times n_e \times C}$. In our experiments, the weak label loss is applied to ground and wall, and the CE loss is applied to car,

$$L_{weak} = -\frac{1}{n_e} \sum_{i=1}^{n_e} \log(1 - \sum_{k_{ic}=0} p_{ic}), \tag{7}$$

where n_e indicates the number of pixels in the range image belonging to this prior category. $k_{ic} = 0$ if the class c is not in this prior-category. p_{ic} is the predicted probability of range pixel i for class c. By using weak labels, we only punish negative predictions.

4 Experiments

4.1 Datasets

We evaluate our UDMA framework on two set-ups: *KITTI360* → *SemanticKITTI* and *GTA5* → *SemanticKITTI*. KITTI360 [18] contains more than 60k real-world RGB images with pixel-level semantic annotations of 37 classes. GTA5 [25] is a synthetic dataset generated from modern commercial video games, which consists of around 25k images and pixel-wise annotations of 33 classes. SemanticKITTI [1] provides outdoor 3D point cloud data with 360° horizontal field-of-view captured by 64-beam LiDAR as well as point-wise semantic annotation of 28 classes.

Table 1. The first row presents the results of a modern segmentation backbone [4]. The medium five rows are 2D UDA methods [12,16,29,31,34]. Ours is the result of our trained model with fine-tuning.

(a) KITTI360 → SemanticKITTI.

Method	Road	Sidewalk	Building	Vegetation	Terrain	Car	mIoU
DeepLabV2	17.96	6.79	5.34	33.24	1.07	0.71	10.85
AdvEnt	20.34	10.87	14.48	34.13	0.06	0.21	13.35
CaCo	27.43	8.73	18.05	34.65	6.15	0.43	15.91
CCM	0.01	5.63	**24.15**	20.83	0	7.11	9.62
DACS	19.35	7.44	0	0.13	0	6.64	5.59
DANNet	15.30	0	13.75	31.46	0	0	10.09
Ours	**45.05**	**21.50**	2.18	**45.99**	**4.18**	**28.49**	**24.57**

(b) GTA5 → SemanticKITTI.

Method	Road	Sidewalk	Building	Vegetation	Terrain	Car	mIoU
DeepLabV2	18.87	0	24.19	17.46	0	0.97	10.24
AdvEnt	0.65	0	11.96	12.32	0	1.45	4.40
CaCo	12.06	13.85	17.21	21.69	3.14	2.61	11.76
CCM	0	2.18	20.47	20.14	0.27	0.83	7.32
DACS	23.73	0	17.02	0	0	0	6.79
DANNet	1.67	**18.28**	**28.19**	0	**12.17**	0	10.05
Ours	**37.63**	1.23	27.79	**31.40**	0.03	**30.00**	**21.35**

4.2 Metrics and Implementation Details

To evaluate the semantic segmentation of point clouds, we apply the commonly adopted metric, mean intersection-over-union (mIoU) over C classes of interests:

$$\frac{1}{C}\sum_{c=1}^{C}\frac{TP_c}{TP_c + FP_c + FN_c}, \tag{8}$$

where TP_c, FP_c, and FN_c represent the number of true positive, false positive, and false negative predictions for class c. We report the labeling performance over six classes that frequently occur in driving scenarios on SemanticKITTI validation set.

Implementation Details: We use the original size of KITTI360 and resize GTA5 images to [720,1280] for training. Our segmentation network is trained using SGD [3] optimizer with a learning rate 2.5×10^{-4}. We use Adam [14] optimizer with learning rate 10^{-4} to train the discriminators which are fully connected networks with a hidden layer of size 256.

4.3 Comparison

Comparison with UDA Methods: We evaluate the performance of adapting RGB images to range images under five 2D UDA state-of-the-art frameworks and one modern 2D semantic segmentation backbone (DeepLabV2). As shown in Table 1, our method outperforms all existing methods in terms of mIoU under both set-ups. Compared to AdvEnt, CaCo, and DANNet which perform scene-level alignment through either adversarial or contrastive learning, our method achieves 8.66 $\sim$ 14.48 mIoU performance gain in KITTI360 and 9.59 $\sim$ 16.95 mIoU gain in GTA5 in terms of mIoU. This is attributed to our instance-level alignment and instance-wise relationship extraction modules. As for CCM and DACS, they show inferior results on both source datasets (even compared to other 2D UDA methods). DACS mixes images across domains to reduce the domain gap, which is more compatible when both domains follow the same color system. However, in our case, mixing $[R, G, B]$ pixels with $[r, r, i]$ range images is unreasonable and may distort the feature space. Similarly, the horizontal layout matrix used in CCM may fail in our setting as we have different horizontal field-of-view in range (360°) and RGB images (90°). They focus more on data-level modification. Our method accomplishes both data-level and feature-level alignment. In our case, KITTI360 achieves higher mIoU, this may attribute to the fact that KITTI360 is using real-world images, whereas synthetic data in GTA5 may cause a large domain gap for learning.

Table 2. Comparison with two 3D baselines.

Method	Road	Side-walk	Building	Vegetation	Terrain	Car	mIoU
Cylinder3D	7.63	5.18	**9.30**	19.01	**26.39**	31.81	16.55
LESS	6.63	6.88	9.15	41.15	23.18	**39.42**	21.07
Ours	**45.05**	**21.50**	2.18	**45.99**	4.18	28.49	**24.57**

Table 3. Ablation Study: effectiveness of IRE, SA, IA.

Label	# of WeChat accounts
Identifiable	11 (13.4 %)
Partially anonymous	44 (53.7 %)
Anonymous	23 (28.0 %)
Unclassifiable	4 (4.9 %)

Comparison with 3D Segmentation Methods: We evaluate Cylinder3D [37] and LESS [19] with prior labels obtained from pre-segmentation in Table 2. Cylinder3D is trained using CE loss for car and weak label loss for ground and wall. For LESS, we employ their proposed losses except L_{sparse}. Comparing with them, our method delivers a minimum of 3.50 mIoU increase in performance. 3D baselines may produce suboptimal outcomes as they depend on the number of accurate 3D label annotations, while our approach utilizes rich RGB labels. We note that 3D-based methods provide better results in some classes (e.g., cars), which may be due to that they leverage 3D geometric features and representations more effectively, while our range image projection causes spatial informa-

tion loss. Future work will explore the possibility of modifying our framework to function directly on the 3D point cloud without the need for projection.

4.4 Ablation Studies

Thorough experiments on KITTI360 → SemanticKITTI are conducted in Table 3 for the effectiveness of network modules and fine-tuning. w/o denotes deleting the corresponding module in our network and * denotes adding fine-tuning. We validate the effectiveness of *IRE*, *SA*, and *IA* based on their 6.95, 3.66, and 2.17 boost in mIoU, respectively (compared to 22.53). Through fine-tuning, we obtain another +2.04 in mIoU.

We also analyze how the number of training data in the source domain affects the result in the target domain in Fig. 4 under the setting KITTI360 → SemanticKITTI.

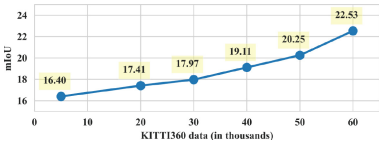

Fig. 4. Analysis for different numbers of training data.

The different number of training data is randomly sampled from KITTI360 with the same training protocol. The mIoU is lifted by 6.13 with 55k more source data. More data allow the model to learn more comprehensive scene features capturing object interaction in driving scenarios. This demonstrates the potential efficacy of our proposed method, where further improvement in 3D mIoU is expected to be achieved with more annotated 2D training data.

5 Conclusion

In this work, we propose and address the cross-modal and cross-domain knowledge transfer task for 3D point cloud semantic segmentation from 2D labels to alleviate the burden of 3D annotations. We tackle the challenges via two complementary modules, Instance-wise Relationship Extraction and Feature Alignment. Extensive experiments illustrate the superiority and potential of our method.

Acknowledgments. This work was supported by NSFC (No.62206173), Natural Science Foundation of Shanghai (No.22dz1201900).

References

1. Behley, J., et al.: Semantickitti: a dataset for semantic scene understanding of lidar sequences. In: ICCV, pp. 9297–9307 (2019)
2. Bian, Y., et al.: Unsupervised domain adaptation for point cloud semantic segmentation via graph matching. In: IROS, pp. 9899–9904. IEEE (2022)
3. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Lechevallier, Y., Saporta, G. (eds.) Proceedings of COMPSTAT'2010. Physica-Verlag HD, pp. 177–186. Springer, Cham (2010). https://doi.org/10.1007/978-3-7908-2604-3_16

4. Chen, L.C., et al.: DeepLab: semantic image segmentation with deep convolutional nets, Atrous convolution, and fully connected CRFs. TPAMI **40**(4), 834–848 (2017)
5. Cortinhal, T., et al.: SalsaNext: fast, uncertainty-aware semantic segmentation of lidar point clouds for autonomous driving (2020). arXiv:2003.03653
6. Fischler, M.A., et al.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. CACM **24**(6), 381–395 (1981)
7. Gerdzhev, M., et al.: Tornado-net: multiview total variation semantic segmentation with diamond inception module. In: ICRA, pp. 9543–9549. IEEE (2021)
8. Goodfellow, I., et al.: Generative adversarial networks. CACM **63**(11), 139–144 (2020)
9. Guo, X., et al.: SimT: handling open-set noise for domain adaptive semantic segmentation. In: CVPR, pp. 7032–7041 (2022)
10. Guo, Y., et al.: Deep learning for 3D point clouds: a survey. TPAMI **43**(12), 4338–4364 (2020)
11. Hou, Y., et al.: Point-to-voxel knowledge distillation for lidar semantic segmentation. In: CVPR, pp. 8479–8488 (2022)
12. Huang, J., et al.: Category contrast for unsupervised domain adaptation in visual tasks. In: CVPR, pp. 1203–1214 (2022)
13. Jaritz, M., et al.: xMUDA: cross-modal unsupervised domain adaptation for 3D semantic segmentation. In: CVPR, pp. 12605–12614 (2020)
14. Kingma, D.P., et al.: Adam: a method for stochastic optimization. arXiv preprint: arXiv:1412.6980 (2014)
15. Langer, F., et al.: Domain transfer for semantic segmentation of LiDAR data using deep neural networks. In: IROS, pp. 8263–8270. IEEE (2020)
16. Li, G., Kang, G., Liu, W., Wei, Y., Yang, Y.: Content-consistent matching for domain adaptive semantic segmentation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12359, pp. 440–456. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58568-6_26
17. Li, W., et al.: SIGMA: semantic-complete graph matching for domain adaptive object detection. In: CVPR, pp. 5291–5300 (2022)
18. Liao, Y., et al.: KITTI-360: a novel dataset and benchmarks for urban scene understanding in 2D and 3D. TPAMI **45**, 3292–3310 (2022)
19. Liu, M., et al.: Less: Label-efficient semantic segmentation for lidar point clouds. In: Avidan, S., Brostow, G., Cisse, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. Lecture Notes in Computer Science, vol. 13699, pp. 70–89. Springer, Cham (2022)
20. Liu, Z., et al.: One thing one click: a self-training approach for weakly supervised 3D semantic segmentation. In: CVPR, pp. 1726–1736 (2021)
21. Milioto, A., et al.: RangeNet++: fast and accurate lidar semantic segmentation. In: IROS, pp. 4213–4220. IEEE (2019)
22. Paul, S., Tsai, Y.-H., Schulter, S., Roy-Chowdhury, A.K., Chandraker, M.: Domain adaptive semantic segmentation using weak labels. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12354, pp. 571–587. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58545-7_33
23. Peng, X., et al.: CL3D: unsupervised domain adaptation for cross-LiDAR 3D detection (2022). arXiv:2212.00244
24. Ren, Z., et al.: 3D spatial recognition without spatially labeled 3D. In: CVPR, pp. 13204–13213 (2021)

25. Richter, S.R., Vineet, V., Roth, S., Koltun, V.: Playing for data: ground truth from computer games. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 102–118. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_7
26. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
27. Sautier, C., et al.: Image-to-lidar self-supervised distillation for autonomous driving data. In: CVPR. pp. 9891–9901 (2022)
28. Shi, H., et al.: Weakly supervised segmentation on outdoor 4D point clouds with temporal matching and spatial graph propagation. In: CVPR, pp. 11840–11849 (2022)
29. Tranheden, W., et al.: DACS: domain adaptation via cross-domain mixed sampling. In: WACV, pp. 1379–1389 (2021)
30. Unal, O., et al.: Scribble-supervised lidar semantic segmentation. In: CVPR, pp. 2697–2707 (2022)
31. Vu, T.H., et al.: ADVENT: adversarial entropy minimization for domain adaptation in semantic segmentation. In: CVPR, pp. 2517–2526 (2019)
32. Wang, Y., et al.: Dynamic graph CNN for learning on point clouds. TOG **38**(5), 1–12 (2019)
33. Wu, B., et al.: SqueezeSeg: convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud. In: ICRA, pp. 1887–1893. IEEE (2018)
34. Wu, X., et al.: DanNet: a one-stage domain adaptation network for unsupervised nighttime semantic segmentation. In: CVPR, pp. 15769–15778 (2021)
35. Yan, X., et al.: 2DPASS: 2D priors assisted semantic segmentation on LiDAR point clouds. In: Avidan, S., Brostow, G., Cisse, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision - ECCV 2022. Lecture Notes in Computer Science, vol. 13688, pp. 677–695. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19815-1_39
36. Zhang, P., et al.: Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In: CVPR, pp. 12414–12424 (2021)
37. Zhu, X., et al.: Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation. In: CVPR, pp. 9939–9948 (2021)