See More and Know More: Zero-shot Point Cloud Segmentation via Multi-modal Visual Data

Supplementary Materials

Yuhang Lu[1,*], **Qi Jiang**[1,*], **Runnan Chen**[2], **Yuenan Hou**[3], **Xinge Zhu**[4], **Yuexin Ma**[1,†]
[1] ShanghaiTech University [2] The University of Hong Kong
[3] Shanghai AI Laboratory [4] The Chinese University of Hong Kong
{luyh2,jiangqi,mayuexin}@shanghaitech.edu.cn

Appendix A. More Details of SVFE

Why SVFE improves the performance? The main function of the SVFE module is to narrow the semantic-visual gap and facilitate early knowledge transfer between semantic and visual spaces, rather than simply scaling up the model. To demonstrate the importance of the semantic-visual interaction, we conduct an experiment where we replace it with self-attention operation with the same parameter scale for each single modality. The results in Table.1 show the performance drops sharply without the SVFE module.

Table 1. Ablation experiments of the design of SVFE module on SemanticKITTI dataset with the 4-unseen-class setting,

Model	Seen	Unseen	Overall	
	mIoU	mIoU	mIoU	hIoU
baseline	54.6	17.3	46.7	26.3
baseline + self attention	57.3	19.4	49.3	29.0
baseline + SVFE	**58.8**	**23.4**	**51.3**	**33.5**
image features first(F'_{es})	58.3	16.1	49.4	25.2
point cloud features first(F_{es})	**58.8**	**26.8**	**52.1**	**36.8**

Does fusion order in SVFE matter? As mentioned in Sec 3.4, semantic feature enhancement is implemented as: $F_{es} = \mathrm{TD}(\mathrm{TD}(F_s, F_l, F_l), F_i, F_i)$. We provide the result of fusing image visual features first and then point cloud visual features: $F'_{es} = \mathrm{TD}(\mathrm{TD}(F_s, F_i, F_i), F_l, F_l)$. As shown in Table.1, The ordering of feature fusion presented in the paper is superior because visual features extracted from point clouds are more central to 3D point cloud segmentation. By fusing these visual features with semantic features first, we are able to provide better guidance for the segmentation process.

Appendix B. More Details of SGVF

Are there any better fusion methods than SGVF module? As SGVF adopts an attention-based design, to further validate the effectiveness of the SGVF module, we design experiments to compare our method with two variants of transformer-based multimodal fusion methods, as shown in Table.2. We find that the performance of “w/o SGVF, w/ cross attention“, which uses LiDAR to query image features for fusion without considering semantic features, is not as good as our SGVF module. This is consistent with our intuition that simply fusing the visual features without considering the semantic information is not sufficient for zero-shot tasks. However, the result of “w/ SGVF, w/ self attention“ is unexpected. The performance of the method with the added self-attention mechanism for the fused features is lower than that of SGVF, even though the parameter quantity is increased. This suggests that simply increasing the model complexity does not necessarily lead to better performance. In fact, the additional self-attention mechanism may have introduced noise and decreased the discriminative power of the fused features.

Table 2. Ablation experiments of the design of SGVF module on SemanticKITTI dataset with the 4-unseen-class setting,

Model	Seen	Unseen	Overall	
	mIoU	mIoU	mIoU	hIoU
w/o SGVF, w/ cross attention	56.6	21.9	49.3	31.6
w/ SGVF, w/ self attention	50.4	21.2	44.3	29.8
Ours	**58.8**	**26.8**	**52.1**	**36.8**

Appendix C. Model inference time

With the addition of an extra image modality, our model's inference time is **0.097 seconds per frame**, slightly larger than **0.087s/f** of the SOTA single-modal method TGP[13]. But our model outperforms it with more than

[*] Equal contribution. † Corresponding author.

50% improvement of unseen category mIOU. Furthermore, it yields real-time performance (All of the results are tested on 1 NVIDIA GTX3090 GPU).

Appendix D. The impact of various image encoders on performance

We employed ResUnet-34 as our image encoder (L591). To show the impact of various image encoders, we replace the encoder with ResUnet-18 and ResUnet-50 and get comparable performance, as shown in the below table.

Model	Seen	Unseen	Overall	
	mIoU	mIoU	mIoU	hIoU
ResUnet-18	57.3	24.7	50.4	34.5
ResUnet-50	**58.9**	**27.1**	**52.2**	**37.1**
Ours(ResUnet-34)	58.8	26.8	52.1	36.8

Appendix E. Discussion on the CLIP Model

Table 3. CLIP model experiment on SemanticKITTI dataset with the 4-unseen-class setting.

Model	Seen	Unseen	Overall	
	mIoU	mIoU	mIoU	hIoU
Ours ← CLIP model	56.6	14.1	47.7	22.6
Ours	**58.8**	**26.8**	**52.1**	**36.8**

Given the success of the CLIP [4] model in 2D zero-shot segmentation [5, 3, 1, 2, 6], we aim to investigate its potential for 3D point cloud semantic segmentation by incorporating the CLIP model into our method. We follow the approach used in MaskCLIP [6], where the class name is inserted into 85 hand-crafted prompts and they are fed into CLIP's text encoder to generate multiple text features. Additionally, we replace the 2D ResUNet backbone with MaskCLIP+. As shown in Table 3, even though unseen objects may already occur in the CLIP training data, causing data leakage, the incorporation of the CLIP model still performs worse than our **pure zero-shot method**. It is mainly because CLIP is based on the contrastive learning between image and text pairs and the significant disparity between point cloud features and image features makes point cloud visual features difficult to align with semantic features extracted by CLIP. However, it is interesting to explore the projection between point cloud and images and transfer the knowledge learnt by CLIP to solve 3D zero-shot problems in large scenarios.

References

[1] Jian Ding, Nan Xue, Guisong Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. *CVPR*, pages 11573–11582, 2022. 2

[2] Boyi Li, Kilian Q. Weinberger, Serge J. Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. *ArXiv*, abs/2201.03546, 2022. 2

[3] Timo Lüddecke and etc. Image segmentation using text and image prompts. *CVPR*, pages 7076–7086, 2022. 2

[4] Alec Radford, Jong Wook Kim, and etc. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2

[5] Mengde Xu, Zheng Zhang, and etc. A simple baseline for zero-shot semantic segmentation with pre-trained vision-language model. *ArXiv*, abs/2112.14757, 2021. 2

[6] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *ECCV*, 2022. 2