Data-Effective Semantic Segmentation on Hand-Object Interaction Involved 4D Point Cloud Videos

Jiye Lee Chaeyun Kim Haemin Jang
Seoul National University
{kay2353, golddohyun, haemin.jang}@snu.ac.kr

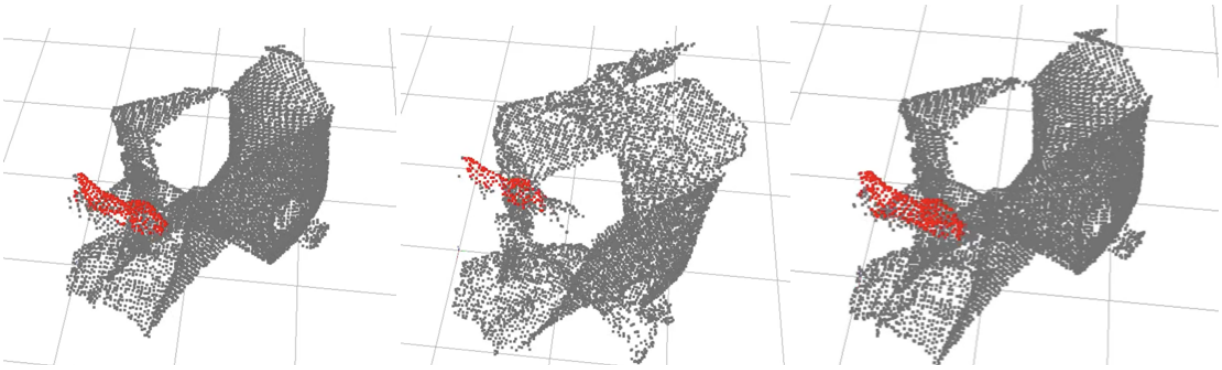

Figure 1. Semantic segmentation of 4D point clouds. Red points indicate points segmented as hands.

Abstract

This paper presents a novel framework for the semantic segmentation of hand-object interactions in real-world 3D scenes using 4D point cloud videos. The framework leverages 4D point clouds, which consist of a 3D point cloud of the environment and temporal streams of point clouds representing dynamic hand and object movements. Our approach addresses the computational challenges associated with processing these complex datasets by introducing a data optimization pipeline. This pipeline aims to effectively reduce the computational costs while maintaining accuracy. Additionally, we introduce active learning and contrastive learning based approaches on top to achieve accurate segmentation with higher data efficiency. We evaluate our framework on a challenging dataset, and further plan to validate the effectiveness of our framework via extensive experiments and comparisons with baselines. We believe this work provides new insights into leveraging temporal information to improve 3D semantic segmentation and facilitates many applications such as robotic manipulation and augmented reality.

1. Introduction

Point clouds are a fundamental and increasingly prevalent type of 3D data, which can be easily obtained using RGB-D sensors. They are valuable source of visual information as the data include details on spatial and temporal information about the environment, making them a critical resource for analyzing complex phenomena such as hand-object interactions. Furthermore, point cloud videos allow for more flexible action recognition in environments with poor visibility and provide more precise geometry dynamics than conventional videos. Therefore, comprehending point cloud videos is crucial for intelligent systems to interact with the world effectively.

Despite the prevalence and importance of 4D point clouds, efficiently leveraging them poses a major challenge due to their computational cost. This trait makes semantic segmentation task particularly challenging, as the complexity and dimension of 4D data can quickly lead to memory and computation explosions.

Furthermore, these challenges are magnified especially for high complexity datasets such as those involving sophisticated hand-object interactions. These datasets contain a large number of labels for segmentation and require de-

tailed 3D information due to the spatial complexity and diversity arising from hand-object interactions. However, modeling the underlying spatio-temporal structure in point cloud videos is extremely challenging due to the irregular and unordered coordinate sets, inconsistent emergence of points across different sets/frames, camera motion, scene changes, occlusion changes, and sampling patterns. These factors cause points between different frames to be unstructured and inconsistent, thus impeding effective integration into the spatio-temporal structure.

To address these challenges, this paper introduces a novel framework for the semantic segmentation of hand-object interaction in 4D point clouds. We suggest a data-efficient pipeline and learning method that focuses on accuracy while drastically reducing the necessary data volume. Our approach builds upon the concept of processing efficiency and employs a data optimization pipeline that substantially diminishes the associated computational costs. Furthermore, our framework aims to learn from point cloud datasets in a more data-efficient manner by additionally integrating active learning and contrastive learning strategies.

Our framework contributes to not only the accuracy of 4D point clouds, we focus on computational efficiency of point cloud segmentation by employing data optimization pipelines and active learning, self-supervised learning methods. Through comparison with baselines, we demonstrate the efficiency of our method. To summarize, our contributions are as follows:

- We introduce a data optimization pipeline that effectively reduces computational costs while maintaining accuracy. To the best of our knowledge, we are the first to apply data optimization prior to point cloud video modeling.
- Active-Contrastive Hybrid learning architecture is applied to achieve data-efficient learning, while improving the feature representation by contrasting the point features.

2. Related Work

Hand-Object Interaction Understanding the spatial relationship of hand object interactions in 3D has been a widely researched topic in the computer vision community. Previous approaches focus on reconstructing accurate 3D structure of hand and objects from single RGB image [3, 14, 38]. Apart from reconstruction, other methods focus on [27, 35] synthesizing natural hand motions such as grasping with a given 3D object. More recent approaches extends this to full body motion [26] However, most of these methods solely focus on the interaction with a single target object, without putting the 3D background scene under consideration.

Understanding Pointclouds Advances in deep learning allowed to understand 3D point clouds in various ways, including segmentation [16,22,23,34], reconstruction [4], and object detection [2]. As these approaches mainly focus on understanding static point clouds, the temporal information is not considered. More recent approaches target 4D point cloud videos, which include not only static 3D information but also the dynamics of objects. As the temporal information has to be considered, computing 4D point cloud videos is computationally challenging compared to static 3D point clouds. There are two major categories that tackle with point cloud video processing. Some approaches apply voxelization to point clouds [21, 32]. Directly performing convolutions on the entire 4D point cloud space along the temporal dimension can be computationally inefficient due to the sparsity of points. While voxelization is one approach to mitigate this issue, it requires additional computation [32] and may not be suitable for applications that require real-time processing. In this work, we focus on directly modeling the point cloud without voxelization, in order to avoid the computational overhead associated with this method.

Second category directly applies the model to raw points. For instance, PointRNNs proposed by Fan and Yangs [7] use a recurrent neural network architecture to predict the movement of point clouds. MeteorNet [18]extends PointNet++ with a temporal dimension and utilizes point tracking-based chained-flow grouping for merging points. PSTNet [9] constructs a spatio-temporal hierarchy to avoid the need for point tracking. More recently, P4Transformer [8] and PPTr [33] have been proposed to capture spatio-temporal correlations across entire point cloud videos without relying on point tracking.

Self-Supervised Learning on Pointclouds Numerous methodologies have been investigated for conducting self-supervised representation learning on point clouds. Early research focused on generative modeling, employing generative adversarial networks [1, 11] and auto-encoders [5, 12, 15] with diverse architectural designs to reconstruct input point clouds. More recent techniques [24, 25, 29, 37] introduce pretext self-supervision tasks, aiming to acquire rich semantic point attributes that ultimately lead to discriminative knowledge at higher levels of abstraction.

However, in this study, we adopt contrastive learning [10] as a means to learn an invariant mapping in the feature space. The efficacy of contrastive learning has been demonstrated in the domain of representation learning for a broad range of computer vision tasks, spanning from unsupervised to supervised contexts. Significantly, recent research has incorporated contrastive learning into the realm of 3D point cloud processing [13, 20, 36] to facilitate unsupervised representation learning as well as 2D segmentation [30, 31].

Notably, PointContrast [36] implements point-level invariant mapping on two transformed views of a given point

cloud. In a similar vein, Liu et al. [17] examine point-level invariant mapping by introducing a point discrimination loss, which enforces feature consistency for points on the shape surface while maintaining inconsistency with randomly sampled noisy points. P4Contrast [20] presents a more adaptable contrasting strategy that fosters multi-modal fusion between geometric and RGB data. Furthermore, to improve segmentation quality on boundary areas, recent work [28] utilizes the contrastive boundary learning framework to address unsatisfactory performance on boundaries.

3. Our Method

3.1. System Overview

The goal of our paper is to determine semantic labels for each point in 4D point clouds, which is a temporal stream of point clouds $\{x_i\}^{t\in 1:T}$ where $x \in R^3$ and t indicates time frame. For each pint x_i, semantic label y_i where $y \in R$ is computed. Therefore, the output would be $\{y_i\}^{t\in 1:T}$ where $y \in R$.

3.2. Data Preprocessing

Dataset Pipeline Optimization The current dataset loader for a set of HDF5 files has been found to have some performance issues, specifically related to the large memory consumption during the initial data load process. HDF5 files themselves are large in size, as they contain metadata in addition to the actual data. Existing data loader loads the entire metadata and data of an HDF5 file into memory, storing each data key in a NumPy array. For example, loading a single training dataset file required allocating 92GB of memory just to store the metadata and NumPy arrays. This can be a bottleneck for systems with limited memory resources. In contrast, the memory usage during the training step itself is much lower, which indicates that the current data pipeline needs to be more optimized for efficient data loading.

In order to solve memory overhead, we tried to build a data pipeline by loading data files converted into a lightweight form instead of directly loading the existing large amount of data. However, if the array data contained in the original HDF5 train file was stored directly in binary form, the original data type and its contents were not preserved intact. Considering this, our data pipeline was built with the following procedure : (a) Load the data in h5 file and convert it to NumPy array, while maintaining its data type (b) Save the dataset to a binary file (either npy or dat format) (c) Load the binary file using np.memmap.

Data Augmentation and Normalization Data augmentation is performed to increase the robustness of our model to variations. Specifically, we augment the point cloud data by adding noise to point positions, simulating real-world conditions where data often contains errors. In addition to augmentation, we also perform data normalization by normalizing the position value of each point with respect to the center coordinate of the point cloud.

3.3. P4Transformer Baseline

In this paper we utiliize P4Transformer [8] as a backbone for our system architecture. P4Transformer [8] introduced a point spatial-temporal 4D convolution, followed by a transformer to capture global appearance and motion information across the entire point cloud video.

Unlike traditional convolutional neural networks (CNNs) that rely on grid-like structures, the P4Transformer operates on point cloud sequences by treating each 3D coordinate set as an unordered set of points. A point 4D convolution enables the model to effectively capture the spatio-temporal local structures present in the point cloud video. The P4Transformer leverages the self-attention mechanism to model the interactions between individual points in a sequence. The model can capture the contextual dependencies and semantic relationships by attending to different points which are necessary for accurate semantic segmentation in 4D point cloud videos.

3.4. Active Learning for Semantic Segmentation

In our proposed method, we aim to enhance data efficiency and performance of semantic segmentation for point cloud data by incorporating an active learning strategy. Active learning is a method where the the most informative data is automatically selected among the data samples from which it learns. By identifying and prioritizing the most informative data points, it is possible for the model to maximize its learning potential per training epoch, thereby achieving a higher degree of data efficiency.

Among various active learning approaches, we utilize margin sampling to obtain the most "informative" data from the training dataset. Margin sampling operates by selecting samples based on the smallest probability difference between the first and second most probable semantic labels. As smaller probability difference indicates the data point where the model is most uncertain of its predictions, they are consequently assumed to be more informative for the learning process.

For each batch within the training dataset, the network generates probabilities for the semantic labels associated with the segmentation task. From these initial output probabilities, margin sampling is employed to identify the most informative instances. In particular, we focus on 2000 points with the highest margin out of the 8192 points per frame. Subsequently, the model is retrained on this selectively sampled data within the training loop. When $P(y|x)$ denotes the probability of a semantic label y with point x, and y_1 and y_2 are the two labels with highest probability,

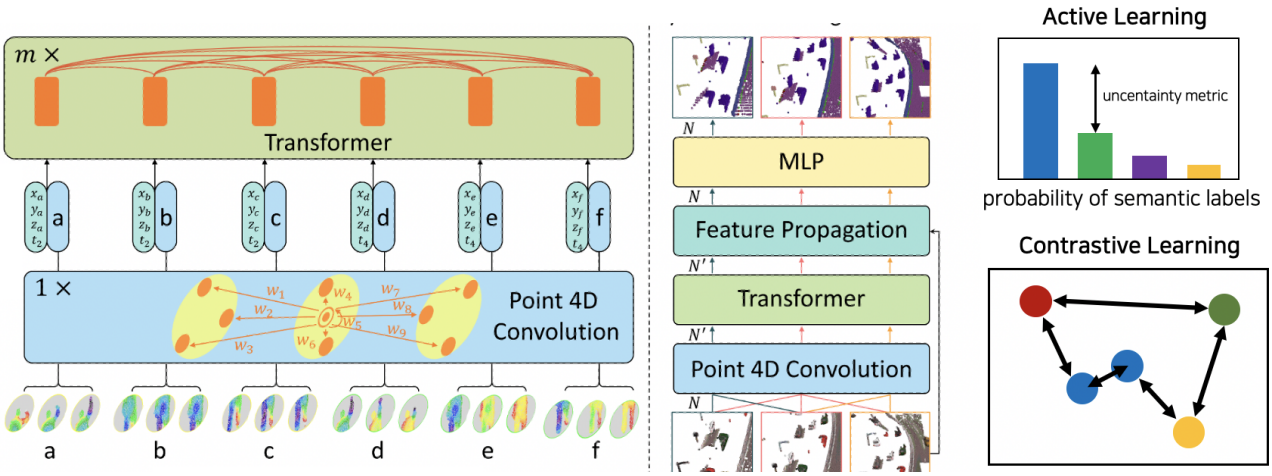

Figure 2. System overview of the P4Transformer baseline with active learning and contrastive learning combined.

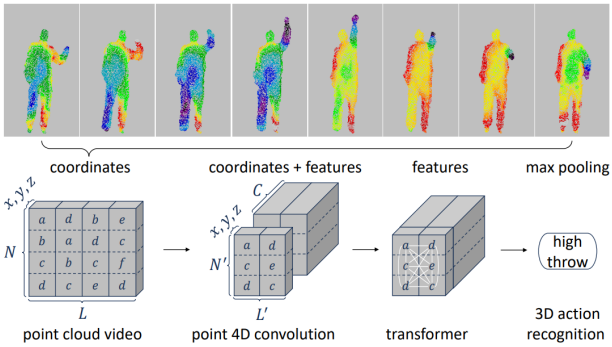

Figure 3. Illustration of point cloud video modeling by our Point 4D Transformer (P4Transformer) network. Color encodes depth.

the margin can be calculated as:

$$M(x) = P(y_1|x) - P(y_2|x) \tag{1}$$

By adopting this active learning approach in our semantic segmentation model, we aim to design a comprehensive and academically rigorous method that offers several key advantages. First and foremost, the method facilitates data-efficient learning by enabling the model to focus on the most informative samples. This, in turn, accelerates the model's learning curve and potentially reduces the overall training time. Additionally, the active learning strategy aids in improving the model's generalization capacity, as it is exposed to a diverse range of informative instances that span the problem space.

3.5. Contrastive Learning

As seen in fig 2, the P4Transformer backbone encodes raw points in point cloud sequences into feature embeddings. To efficiently utilize the given data, we exploit the inherent structure of these feature embeddings to learn useful representations. The primary objective of our method is to enhance the representation of point clouds in the training process by leveraging the power of contrastive learning on feature embeddings. Our approach involves a contrastive loss function. The contrastive loss function is designed to improve the discriminative power of the learned feature embeddings. By pulling together embeddings of points belonging to the same semantic class, the model learns to capture the defining characteristics of each class and differentiate them from other classes. Conversely, by pushing apart embeddings of points from different classes, the model learns to identify and highlight their differences. This approach promotes the formation of semantically meaningful clusters in the embedding space, which can be used to improve downstream tasks, such as object detection or segmentation. Let $\mathbf{z}$ denote the feature embeddings of point x and y denote the labels.

The pairwise distances between the feature vectors are computed as:

$$D_{ij} = ||\mathbf{z}_i - \mathbf{z}_j||_2^2 \tag{2}$$

D_{ij} is the pairwise distance between feature embeddings $\mathbf{z}_i$ and $\mathbf{z}_j$. The distance should be minimized when the feature embeddings share same labels, and should be maximized when the labels are different. Then a label matrix L is formed as:

$$L_{ij} = \begin{cases} 1 & \text{if } y_i = y_j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where L_{ij} is 1 if semantic label y_i and y_j are identical, and 0 otherwise. The contrastive loss $Loss$ for each pair of points is then calculated as:

$$Loss_{ij} = L_{ij} \cdot D_{ij}^2 + (1 - L_{ij}) \cdot \max(0, m - D_{ij})^2 \tag{4}$$

where m indicates the margin of distance. The total contrastive loss is the mean over all the calculated losses:

$$Loss = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} Loss_{ij} \tag{5}$$

Here, N is the total number of data points in the batch.

4. Experiments

In this section, we first introduce implementation details and datasets used in this paper.

4.1. Evaluation Metrics and Datasets

Dataset We use HOI4D [19] dataset for training. HOI4D dataset is a large-scale 4D egocentric dataset with rich annotations for category-level human-object interaction, which includes 2.4M RGB-D egocentric video frames over 4000 sequences collected by 9 participants interacting with 800 different object instances from 16 categories over 610 different indoor rooms.

Each 4D visual sequence is densely annotated with frame-wise panoptic segmentation, motion segmentation, 3D hand pose, rigid and articulated object pose, and action segmentation. These annotations provide ground truth information for evaluating the performance of our semantic segmentation model. The dataset delivers high levels of detail for human-object interaction at the category level.

Using 4D point cloud visualizer, we acknowledged that the quality of ground truth annotations was limited. However, we still decided to proceed with this dataset as it is one of the only datasets that provides both 4D point clouds and hand-object interaction.

Metrics For quantitative performance evaluation, we use mean intersection-over-union(mIoU) [6], which has become the de facto standard for measuring the quality of instance-level segmentation results. The metric is leveraged over all classes, given by :

$$\frac{1}{C} \sum_{c=1}^{C} \frac{TP_c}{TP_c + FP_c + FN_c} \tag{6}$$

, where TP_c, FP_c, and FN_c represent true positive, false positive, and false negative predictions for class c, and C implies the number of classes.

4.2. Implementation Details

Setup Utilizing temporal data can enhance the comprehension of dynamic objects within a scene, leading to improved segmentation accuracy and resilience against noise. Due to constraints in memory, current approaches can only handle point cloud videos up to three frames. Note that, while it is possible to achieve 4D semantic segmentation from a single frame, incorporating temporal correlation can provide a more nuanced understanding of scene structure, leading to improved segmentation accuracy and noise resilience. By considering the temporal aspects of a scene, a more comprehensive understanding of its underlying organization can be obtained, resulting in superior segmentation outcomes. Initially, our experiments adhered to the methodologies employed by previous studies.

Baseline Comparison In order to establish a solid baseline for our proposed approach, we conducted a thorough comparison with several prior works on 4D segmentation, including P4Transformer, PSTNet, and PPTr. These works have demonstrated impressive results on the HOI4D dataset, which is widely used in 4D segmentation research. However, due to variations in training and testing setups, the reported performance numbers for each of these works may differ. To address this issue, we trained both P4Transformer and PPTr models on the HOI4D dataset using the standard settings described in their respective papers for 50 epochs. Our comparison revealed that P4Transformer outperformed PSTNet, achieving an mIOU of 61.97, which was nearly 10 points higher than that of PSTNet as shown in Table 1. Although PPTr exhibited better performance than P4Transformer, its GPU memory consumption during training was significantly higher. Specifically, with a batch size of 8, P4Transformer requires 15,000 GPU memory, whereas PPTr requires 40,000 GPU memory. In light of our goal to achieve data-efficient training, we selected P4Transformer as our baseline and further developed three unique modules to address its current limitations. Our results indicate that the proposed approach, which builds upon the P4Transformer model, outperforms the baseline model while maintaining a reasonable GPU memory consumption. Overall, our comparison with prior works on 4D segmentation highlights the importance of carefully evaluating and selecting a suitable baseline model. While PPTr exhibited better performance than P4Transformer, its high GPU memory consumption may not be practical in certain scenarios. Therefore, we believe that our choice of P4Transformer as the baseline model was a reasonable decision, given our focus on data-efficient training.

Training Details Our model aims to learn from point cloud datasets in a more data-efficient manner by incorporating three proposed modules, namely margin sampling, centering, and contrastive learning, into the backbone architecture of P4Transformer. In the preceding section, we contrasted the efficacy of the baseline P4Transformer and comparative group models by training them over a span of 50 epochs. However, given the constraints of time and resources, coupled with the voluminous nature of the dataset, we opted to revise our original training plan by reducing the number of epochs to 20. In order to achieve a judicious balance between realizing meaningful results and optimally

Method	Frames	Table	Ground	Metope	Locker	Pliers	Laptop	Safe Deposit	Pillow	Hand/Arm	mIoU
PSTNet	3	57.45	63.38	83.80	44.69	13.71	35.03	51.55	76.30	40.39	51.81
P4Transformer	3	63.58	66.60	87.17	58.39	32.29	72.03	65.87	57.41	54.36	61.97
PPTr	3	66.78	72.76	88.21	60.83	41.22	72.04	73.10	80.64	61.27	68.54

Table 1. Baseline Evaluation Comparison for semantic segmentation on HOI4D Dataset

Method	Input	Frames	mIoU
PSTNet	point	3	32
P4Baseline	point	3	31.2
P4MNC	point	3	26.1

(a) Quantitative comparision with SOTA methods.

Method	Input	Frames	mIoU
P4M	point	3	23.5
P4N	point	3	21.2
P4C	point	3	25.9
P4MNC	point	3	26.1

(b) Ablation on various training options.

Table 2. **Comparison of mIoU(%) and ablation for semantic segmentation on HOI4D Dataset.** We report performance on mIoU of 20 epoch-trained Baseline models and proposed model options. We use "M" to denote margin sampling, "N" to denote representation learning by centering, "C" to denote Contrastive Learning method, and "MNC" to denote method where all three options mentioned above are used.

managing available resources, we arrived at the decision to utilize train 1, 2, and 4 from the HOI4D dataset for training purposes, while reserving train 3 for testing. To ensure fairness and a unified evaluation test-bed, we retrained state-of-the-art (SOTA) models using the prescribed settings as reported in their respective papers, but tailored to our target dataset. The retraining process was carried out under identical conditions, utilizing the same GPU and a batch size of 8, ensuring a rigorous and equitable comparison between the SOTA models and our proposed approach.

4.3. Comparison with the State-of-the-Art Methods

To inspect the generalizability of our proposed methods, we evaluate our model on a large dataset and compare the results with recent approaches for semantic segmentation of 4D point clouds, which includes: P4Transformer and PPTr. From train3 of the HOI4D dataset, we randomly selected 200 data points for evaluation.

mIOU We observed that our model (P4MNC) achieved an mIOU of 26.1, which is 5.1 lower than the mIOU of our baseline model. It is important to note that the comparison with PSTNet and P4Baseline in Table 2 is not aligned with the results in Table 1. The discrepancy arises from the fact that our model was trained for 20 epochs, while the comparison models were trained for 50 epochs. Although our model was trained with much less epoches, our model showed on-par performance compared to baseline models. This demonstrates that our model is more data-efficient compared to baselines. For instance, the P4Transformer achieved an mIOU of 61.97 when trained for 50 epochs, whereas it only attained 31.2 under 20 epochs. Consequently, it is challenging to draw meaningful conclusions and make a fair comparison under such disparate training settings. We believe that if we had sufficient time and memory resources, the results would likely differ significantly.

4.4. Ablation study

In this section, we conduct ablation studies using the P4Transformer as the baseline to further investigate individual contributions of our proposed modules. The training settings remain consistent with Section 4.2, and we present the corresponding test results. Since we discussed that it is hard to make comparison with baseline model under 20 epochs, we primarily focus on the results of each modules individually applied, P4M, P4N, P4C and our model, P4MNC presented in Table 2.

Effect of Margin sampling As the core operation of our proposed model, implementation of active learning for semantic segmentation with margin sampling can enhance the sampling of the most informative data from training dataset. To verify its effectiveness, we introduce the margin sampling module to the baseline model, P4Transformer, and evaluted its impact on performance. The inclusion of margin sampling resulted in noticeable improvement, mIOu reaching 23.5. This demonstrates that the margin sampling module effectively enhances the model's ability to gain more informative data on where the models is most uncertain of its prediction.

Effect of Centering Inclusion of clustering-based approach enabled our model to capture fine-grained spatial details. mIOU reached 21.2 followed by the introduction of centering to baseline highlights the effectiveness of this module, indicating that the centering plays a vital role in our architecture.

Effect of Contrastive learning As the raw points are encoded into feature embeddings in the point cloud sequence, defining a contrastive loss function on them can improve the overall performance of the model. To confirm this, we incorporated contrastive loss function into the baseline,

and this led to huge improvement in mIOU each reaching 25.9. This shows that the module enable the model to learn discriminative representations and better differentiates between classes.

Overall, these ablation studies highlight the importance of the margin sampling, centering, and contrastive learning modules in our proposed approach, as they significantly contribute to the model's performance. While our model with all three modules achieves the highest performance with an mIOU of 26.1, the inclusion of a single module alone does not reach this level of performance. This indicates that the combinations of all these modules are required to fully exploit their synergistic effects and achieve the best segmentation results.

5. Discussions

5.1. Limitations

As mentioned in Section 4.2, it was highlighted that due to time and memory constraints, the model was only trained for 20 epochs instead of the initially planned 50 epochs. This abbreviated training duration may have limited the model's ability to converge and reach its optimal performance. As such, the results obtained may not fully reflect the model's true potential. Additionally, the reduced training time may have affected the model's capacity to capture more intricate patterns and nuances in the 4D point cloud data. Future studies could consider training the model for longer durations to evaluate its performance more comprehensively.

Furthermore, the ablation study conducted in this paper primarily focused on evaluating the individual impact of each proposed module when added individually to the baseline model. However, different combinations of these modules were not extensively explored, and it is possible that certain combinations of modules could have resulted in even greater performance improvements. Therefore, the overall impact and synergy of different module combinations remain unexplored in this paper. Future studies could focus on evaluating the effects of various module combinations to identify the optimal configuration for improving the model's performance.

5.2. Future Ideas

Applying Contrastive Boundary Learning One suggested idea for further improvement is to explore the use of multi-scale contrastive boundary learning framework for spectral clustering that aims to improve the alignment of model predictions with ground truth data's boundaries. The contrastive boundary learning approach mainly focuses on boundary points only and aims to learn representations that are more similar to their neighbor points from the same category and more distinguished from neighbor points from different categories. Furthermore, it is anticipated that the current subscene boundary mining approach can be adapted to apply the boundary loss to the reduced feature space obtained from multiple transformer blocks.

By solely focusing on the boundary points, the proposed contrastive learning framework offers a more efficient and effective way to improve the model's ability to capture the contours of the data. By exploiting the inherent constraints of the boundary points, the model can learn more informative representations that are better aligned with the underlying structure of the data.

The application of contrastive boundary loss and subscene boundary mining to 4D point cloud data represents a novel and unexplored research direction. However, the integration of sub-sampled boundary mining in 4D point clouds poses a significant challenge due to the abundance of information to process in the time dimension. Addressing this challenge requires developing innovative techniques that can efficiently handle the increased complexity and information volume of 4D point cloud data, thus paving the way for more effective and comprehensive learning of the data's contours.

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. In *International conference on machine learning*, pages 40–49. PMLR, 2018. 2

[2] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *CVPR*, 2017. 2

[3] Zerui Chen, Yana Hasson, Cordelia Schmid, and Ivan Laptev. Alignsdf: Pose-aligned signed distance fields for hand-object reconstruction. In *ECCV*, 2022. 2

[4] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 2

[5] Benjamin Eckart, Wentao Yuan, Chao Liu, and Jan Kautz. Self-supervised learning on 3d point clouds by learning discrete generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8257, 2021. 2

[6] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111, 2015. 5

[7] Hehe Fan and Yi Yang. Pointrnn: Point recurrent neural network for moving point cloud processing. *arXiv preprint arXiv:1910.08287*, 2019. 2

[8] Hehe Fan, Yi Yang, and Mohan Kankanhalli. Point 4d transformer networks for spatio-temporal modeling in point cloud videos. In *CVPR*, 2021. 2, 3

[9] Hehe Fan, Xin Yu, Yuhang Ding, Yi Yang, and Mohan Kankanhalli. Pstnet: Point spatio-temporal convolution on

point cloud sequences. *arXiv preprint arXiv:2205.13713*, 2022. 2

[10] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006. 2

[11] Zhizhong Han, Mingyang Shang, Yu-Shen Liu, and Matthias Zwicker. View inter-prediction gan: Unsupervised representation learning for 3d shapes by learning global shape memories to support local view predictions. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8376–8384, 2019. 2

[12] Zhizhong Han, Xiyang Wang, Yu-Shen Liu, and Matthias Zwicker. Multi-angle point cloud-vae: Unsupervised feature learning for 3d point clouds from multiple angles by joint self-reconstruction and half-to-half prediction. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10441–10450. IEEE, 2019. 2

[13] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15587–15597, 2021. 2

[14] Di Huang, Xiaopeng Ji, Xingyi He, Jiaming Sun, Tong He, Qing Shuai, Wanli Ouyang, and Xiaowei Zhou. Reconstructing hand-held objects from monocular video. In *SIGGRAPH Asia 2022 Conference Papers*, 2022. 2

[15] Jiaxin Li, Ben M Chen, and Gim Hee Lee. So-net: Self-organizing network for point cloud analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9397–9406, 2018. 2

[16] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. In *NeurIPS*, 2018. 2

[17] Fayao Liu, Guosheng Lin, Chuan-Sheng Foo, Chaitanya K Joshi, and Jie Lin. Point discriminative learning for unsupervised representation learning on 3d point clouds. *arXiv preprint arXiv:2108.02104*, 2021. 3

[18] Xingyu Liu, Mengyuan Yan, and Jeannette Bohg. Meteornet: Deep learning on dynamic 3d point cloud sequences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9246–9255, 2019. 2

[19] Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *CVPR*, 2022. 5

[20] Yunze Liu, Li Yi, Shanghang Zhang, Qingnan Fan, Thomas Funkhouser, and Hao Dong. P4contrast: Contrastive learning with pairs of point-pixel pairs for rgb-d scene understanding. *arXiv preprint arXiv:2012.13089*, 2020. 2, 3

[21] Wenjie Luo, Bin Yang, and Raquel Urtasun. Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In *CVPR*, 2018. 2

[22] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 2

[23] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017. 2

[24] Yongming Rao, Jiwen Lu, and Jie Zhou. Global-local bidirectional reasoning for unsupervised representation learning of 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5376–5385, 2020. 2

[25] Chao Sun, Zhedong Zheng, Xiaohan Wang, Mingliang Xu, and Yi Yang. Point cloud pre-training by mixing and disentangling. *arXiv e-prints*, pages arXiv–2109, 2021. 2

[26] Omid Taheri, Vasileios Choutas, Michael J Black, and Dimitrios Tzionas. Goal: Generating 4d whole-body motion for hand-object grasping. In *CVPR*, 2022. 2

[27] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasping of objects. In *ECCV*, 2020. 2

[28] Liyao Tang, Yibing Zhan, Zhe Chen, Baosheng Yu, and Dacheng Tao. Contrastive boundary learning for point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8489–8499, 2022. 3

[29] Hanchen Wang, Qi Liu, Xiangyu Yue, Joan Lasenby, and Matt J Kusner. Unsupervised point cloud pre-training via occlusion completion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9782–9792, 2021. 2

[30] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7303–7313, 2021. 2

[31] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3024–3033, 2021. 2

[32] Yancheng Wang, Yang Xiao, Fu Xiong, Wenxiang Jiang, Zhiguo Cao, Joey Tianyi Zhou, and Junsong Yuan. 3dv: 3d dynamic voxel for action recognition in depth video. In *CVPR*, 2020. 2

[33] Hao Wen, Yunze Liu, Jingwei Huang, Bo Duan, and Li Yi. Point primitive transformer for long-term 4d point cloud video understanding. In *ECCV*, 2022. 2

[34] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *CVPR*, 2019. 2

[35] Yan Wu, Jiahao Wang, Yan Zhang, Siwei Zhang, Otmar Hilliges, Fisher Yu, and Siyu Tang. Saga: Stochastic whole-body grasping with contact. In *ECCV*, 2022. 2

[36] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 574–591. Springer, 2020. 2

[37] Juyoung Yang, Pyunghwan Ahn, Doyeon Kim, Haeil Lee, and Junmo Kim. Progressive seed generation auto-encoder

for unsupervised point cloud learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6413–6422, 2021. 2

[38] Yufei Ye, Abhinav Gupta, and Shubham Tulsiani. What's in your hands? 3d reconstruction of generic objects in hands. In *CVPR*, pages 3895–3905, 2022. 2