Complete-to-Partial 4D Distillation for Self-Supervised Point Cloud Sequence Representation Learning

Zhuoyang Zhang[1] * Yuhao Dong[1] * Yunze Liu[1,3] Li Yi[1,2,3] †
[1] IIIS, Tsinghua University [2] Shanghai Artificial Intelligence Laboratory [3] Shanghai Qi Zhi Institute

Abstract

Recent work on 4D point cloud sequences has attracted a lot of attention. However, obtaining exhaustively labeled 4D datasets is often very expensive and laborious, so it is especially important to investigate how to utilize raw unlabeled data. However, most existing self-supervised point cloud representation learning methods only consider geometry from a static snapshot omitting the fact that sequential observations of dynamic scenes could reveal more comprehensive geometric details. To overcome such issues, this paper proposes a new 4D self-supervised pre-training method called Complete-to-Partial 4D Distillation. Our key idea is to formulate 4D self-supervised representation learning as a teacher-student knowledge distillation framework and let the student learn useful 4D representations with the guidance of the teacher. Experiments show that this approach significantly outperforms previous pre-training approaches on a wide range of 4D point cloud sequence understanding tasks. Code is available at: https://github.com/dongyh20/C2P.

1. Introduction

Recently, there is a surge of interest in understanding point cloud sequences in 4D (3D space + 1D time) [7, 8, 11, 21, 30]. As the direct sensor input for a wide range of applications including robotics and augmented reality, point cloud sequences faithfully depict a dynamic environment regarding its geometric content and object movements in the context of the camera ego-motion. Though widely accessible, such 4D data is prohibitively expensive to annotate in large scale with fine details. As a result, there is a strong need for leveraging the colossal amount of unlabeled sequences. Among the possible solutions, self-supervised representation learning has shown its effectiveness in a wide range of fields including images [2, 12, 13], videos [6, 9, 15, 22, 28] and point clouds [16, 24, 31, 33, 34].

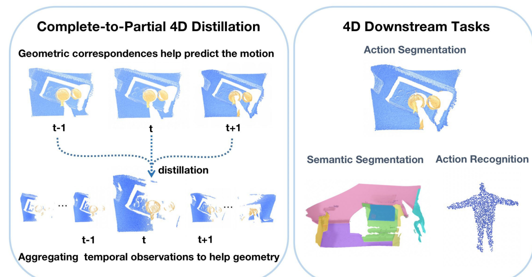

Figure 1. We propose a complete-to-partial 4D distillation (C2P) approach. Our key idea is to formulate 4D self-supervised representation learning as a teacher-student knowledge distillation framework in which students learn useful 4D representations under the guidance of a teacher. The learned features can be transferred to a range of 4D downstream tasks.

We therefore aim to fill in the absence of self-supervised point cloud sequence representation learning in this work.

Learning 4D representations in a self-supervised manner seems to be a straightforward extension of 3D cases. However, a second thought reveals its challenging nature since such representations need to unify the geometry and motion information in a synergetic manner. From the geometry aspect, a 4D representation learner needs to understand 3D geometry in a dynamic context. However, most existing self-supervised point cloud representation learning methods [16, 24, 34] only consider geometry from a static snapshot, omitting the fact that sequential observations of dynamic scenes could reveal more comprehensive geometric details. From the motion aspect, a 4D representation learner needs to understand motion in the 3D space, which requires an accurate cross-time geometric association. Nevertheless, existing video representation learning frameworks [6, 9, 22] mostly model motion as image space flows so geometric-aware motion cues are rarely encoded. Due to such challenges, 4D has been rarely discussed in the self-supervised representation learning literature, with only a few works [4, 23] designing learning objectives in 4D to facilitate static 3D scene understanding.

To address the above challenges, we examine the nature of 4D dynamic point cloud sequences, and draw two

*Equal contribution with the order determined by rolling dice.
†Corresponding author.

main observations. First, most of a point cloud sequence depicts the same underlying 3D content with an optional dynamic motion. Motion understanding could help aggregate temporal observations to form a more comprehensive geometric description of the scene. Second, geometric correspondences across time could help estimate the relative motion between two frames. Therefore better geometric understanding should facilitate a better motion estimation. At the core are the synergetic nature of geometry and motion.

To facilitate the synergy of geometry and motion, we develop a Complete-to-Partial 4D Distillation (C2P) method. Our key idea is to formulate 4D self-supervised representation learning as a teacher-student knowledge distillation framework and let the student learn useful 4D representations with the guidance of the teacher. And we present a unified solution to the following three questions: How to teach the student to aggregate sequential geometry for more complete geometric understanding leveraging motion cues? How to teach the student to predict motion based upon better geometric understanding? How to form a stable and high-quality teacher?

In particular, our C2P method consists of three key designs. First, we design a partial-view 4D sequence generation method to convert an input point cloud sequence which is already captured partially into an even more partial sequence. This is achieved by conducting view projection of input frames following a generated camera trajectory. The generated partial 4D sequence allows bootstrapping multi-frame geometry completion. This is achieved by feeding the input sequence and the generated partial-view sequence to teacher and student networks respectively and distill the teacher knowledge to a 4D student network. Second, the student network not only needs to learn completion by mimicking the corresponding frames of the teacher network, but also needs to predict the teacher features of other frames within a time window, to achieve so-called 4D distillation. Notice the teacher feature corresponds to more complete geometry, which also encourages the student to exploit the benefit of geometry completion in motion prediction. Finally, we design an asymmetric teacher-student distillation framework for stable training and high-quality representation, i.e., the teacher network has weaker expressivity compared with the student but is easier to optimize.

We evaluate our method on three downstream tasks including indoor and outdoor scenarios: 4D action segmentation on HOI4D [21], 4D semantic segmentation on HOI4D [21], 4D semantic segmentation on Synthia 4D [27] and 3D action recognition on MSR-action3D [17]. We demonstrate significant improvements over the previous method($+2.5\%$ accuracy on HOI4D action segmentation, $+1.0\%$ mIoU on HOI4D semantic segmentation, $+1.0\%$ mIoU on Synthia 4D semantic segmentation and $+2.1\%$ accuracy on MSR-Action3D).

The contributions of this paper are fourfold: First, we propose a new 4D self-supervised representation learning method named Complete-to-Partial 4D Distillation which facilitates the synergy of geometry and motion learning. Second, we propose a natural and effective way to generate partial-view 4D sequences and demonstrate that it can work well as learning material for knowledge distillation. Third, we find that asymmetric design is crucial in the complete-to-partial knowledge distillation process and we propose a new asymmetric distillation architecture. Fourth, extensive experiments on three tasks show that our method outperforms previous state-of-the-art methods by a large margin.

2. Related Work

4D Point Cloud Sequence Understanding. Unlike 3D static scenes, Understanding 4D point cloud sequences require more focus on leveraging spatiotemporal information to perceive complete geometry and sensitive motion. There are many 4D sequence-based tasks [17, 21, 27] that have received extensive attention, such as 4D semantic segmentation [21, 27], 3D action recognition [17], 4D action segmentation [21], etc. These tasks tend to have high computational overhead, as 4D data require a large memory occupation. According to the representation, existing 4D backbone can be categorized into voxel-based [5, 18], and point-based methods [7, 8, 19, 30]. The state-of-the-art approach is based on the transformer architecture [7, 30], which is often difficult to optimize and requires a large amount of labeled data for training. An under-explored problem for 4D sequence understanding is how to learn spatio-temporal features in an unsupervised manner to reduce the difficulty of network optimization and the amount of labeled data required. We proposed a new unsupervised contrastive knowledge distillation approach to learn 4D representation.

3D Representation Learning. Benefiting from the rapid development of 2D representational learning [12, 13, 20], 3D representational learning [23, 24, 32, 34] has also been widely explored. Existing methods can be grouped into generative-based methods [1, 29, 32, 33], and context-based methods [26, 31, 35]. For generative-based [1, 29, 32] methods, PSG-Net [1] propose to learn representation by reconstructing point cloud objects with seed generation. OcCo [29] utilize a U-net to complete occluded point cloud objects. Point-Bert [32] learn the representation for Transformers by recovering masked object parts. For context-based methods, PointContrast [31] propose contrasting different views of scene point clouds at point-level. CSC [14] adopt a spatial partition to improve pointcontrast. RandomRooms [26] construct pseudo scenes with synthetic objects for contrastive learning. SelfCorrection [3] learn the informative representation by distinguishing and restoring destroyed objects. DepthContrast [35] extend to contrast with multiple representations (points and voxels) at scene-

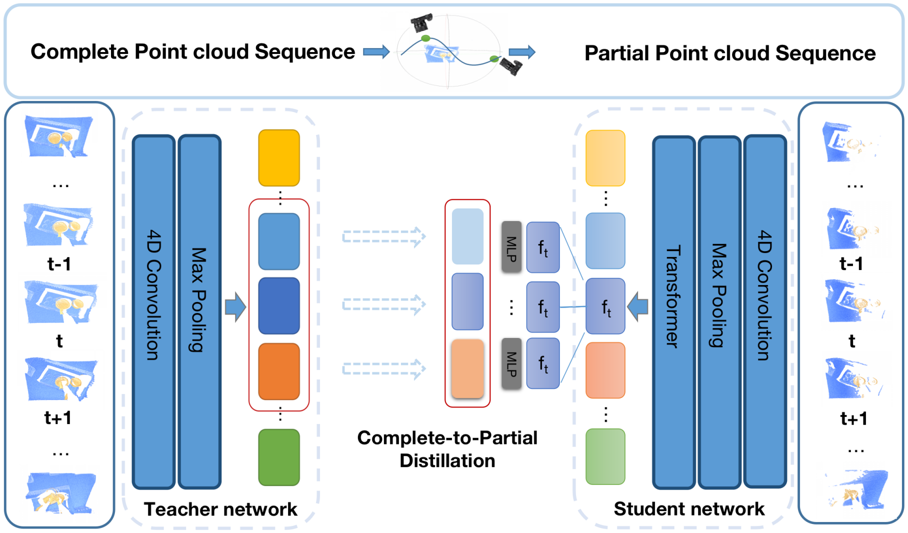

Figure 2. Overview of our method. Our method can be divided into three main parts, partial-view generation, asymmetric twin network feature extraction, and complete to partial 4D knowledge distillation. (a) In the partial-view generation part, our goal is to generate a sequence of partial-views related to the perspective. (b) Asymmetric teacher-student network can perform 4D knowledge distillation in a better and more stable way. (c) Then the knowledge from the teacher network within a time window is distilled into the features of a frame extracted by the student network.

level. Due to the high dimensionality of 4D data and the required computational overhead, most of the existing methods cannot be directly extended to 4D data. DCGLR [10] is most relevant to us, which first constructs the global point cloud set and the local point cloud set by cropping the full point cloud with different cropping ratios and then distilling knowledge from the local branch to the global branch. Our goal is to investigate how to extract high-quality spatio-temporal features at the scene-level to help 4D downstream tasks, so we use teacher network to distill 4D complete information to the student network using contrastive learning.

4D Representation Learning. 4D Representation Learning is still a relatively new field, and thanks to the experience in 3D, researchers have gradually started to focus on how to exploit information from 4D sequences. 4Dcontrast [4] proposes to exploit 4D motion information to improve the effectiveness of 3D tasks. However, due to the memory overhead, it is difficult to use this point-level comparison learning method on long 4D sequences. STRL [16] uses temporal-spatial contrastive learning to learn good representations of 3D objects. Although pre-trained on 4D data, both of these methods are designed to learn static 3D representations. Our approach broadens this scope to use 4D data for pre-training to improve the understanding of dynamic scenes, and to our knowledge, our approach is also the first exploration in this direction.

3. Method

In this paper, we propose a new 4D self-supervised representation learning method named Complete-to-Partial 4D Distillation. To avoid ambiguity, we first emphasize the definitions of complete and partial. The "complete" represents an original input 4D point cloud sequence, which was obtained through the natural collection. A "complete point cloud" may still be a geometrically incomplete point cloud, but it can be fully described as complete compared to the new 4D point cloud sequence we generate. The "partial" represents a synthetic point cloud generated from the captured data, which is significantly less complete than the original point cloud, so we call it a "partial 4D point cloud sequence". In this paper, the definitions of partial and partial-view are not distinguished since we are generating new data through the camera view.

Our main idea is to distill the spatial-temporal information of a complete point cloud sequence into a partial point cloud sequence so that the neural network can extract strong features in a self-supervised manner.

The overview of our method is shown in Figure 2. Our method is divided into three main parts, partial-view generation, asymmetric twin network feature extraction, and complete to partial 4D knowledge distillation. In the partial-view generation part, our goal is to generate a sequence of partial-views related to the perspective. Since this approach can generate a wide variety of 4D sequences, various motion patterns during the point cloud change can be simulated. After this, the partial-view sequences are fed to the student network and the complete point cloud sequences are fed to the teacher network. We design an asymmetric teacher-student network to perform 4D knowledge distillation in a better and more stable way. The teacher network has a weaker representational capacity, which facilitates the student network to distill stable information in an easy-optimized manner. Finally, the knowledge from the teacher network within a time window is distilled into the features of a frame extracted by the student network, which makes the student network not only learn the geometric completeness but also capture the motion information from the spatial-temporal context and the geometric change information of the foreground-background. Since the student network is distilling spatio-temporal features from a time window of the teacher network, we call the method a 4D knowledge distillation framework.

In the rest of this section, we will introduce our method in detail. We first show how to generate the partial point cloud sequence in section 3.1. Then we present the asymmetric teacher-student network design in section 3.2. In section 3.3, We illustrate in detail how to distill 4D spatio-temporal information from complete point cloud sequences to partial point cloud sequences.

3.1. Generating partial-view 4D sequences

A static 3D point cloud can generate new point clouds with different views. These new data can reflect the geometric properties of the same 3D object/scene in a complementary way. Inspired by this observation, different 4D sequences can be generated according to different trajectories. We think it is a natural idea to use different motion trajectories in 4D data to generate a large number of partial-view 4D sequences. These new trajectories are all descriptions of the same "complete 4D sequence", but the differences in their poses and occlusion patterns can reflect the real physical world comprehensively.

We define the "complete point cloud sequence" with sequence length L as $S = \{s_1, s_2, \ldots, s_L\}$ where each s_i denotes one frame. For each frame, it consists of N points, $P_i = \{p_1^i, p_2^i, \ldots, p_N^i\}$, where each point p^i is a vector of both coordinates and other features such as color and normal. We generate the "partial point cloud sequence" by sampling a natural camera trajectory and occluding invisible points from the camera view. Figure 3 show the process of partial-view 4D sequences generation. And we will describe how to sample natural camera trajectory and how to do occlusion sampling in detail.

Sampling Natural Camera Trajectory. Based on the camera trajectory we can generate a new 4D sequence and sample the visible points according to the camera pose of each frame, which naturally fits the real physical world better than random sampling. Moreover, random sampling is not consistent, which hinders the network from learning spatio-temporal contextual information from the data. We believe that a natural camera trajectory can simulate the motion of the 3D world well, so the new 4D data generated is more conducive to learning spatio-temporal features of the network. We also verified this in our experiments.

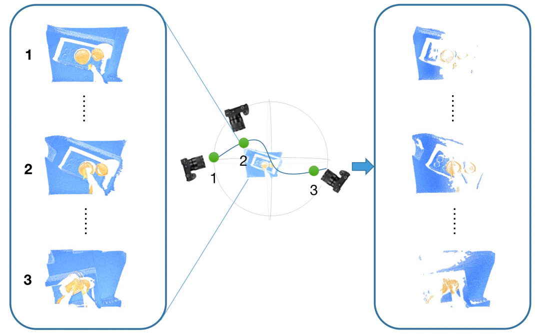

Figure 3. Illustration of partial-view sequences generation.

Specifically, we first determine a sphere on which our trajectory can be approximated as a curve. The center of the sphere is the same as the center of the point cloud and the radius is the distance between the center of the point cloud and the original camera position. For each position on the sphere, it can actually be determined by two angles θ and ϕ. The θ determines the angle on the horizontal plane and ϕ determines the angle on the vertical plane. We define the sphere coordinates of the original camera position as $(\frac{\pi}{2}, 0)$. For the horizontal movement, we have two movement modes: one is from $-\frac{5}{12}\pi$ to $\frac{5}{12}\pi$ (left to right from the original camera view), and the other is from $\frac{5}{12}\pi$ to $-\frac{5}{12}\pi$ (right to left from the original camera view). Each move pattern has a horizontal angle change of 150 degrees, and we set the angle change evenly for adjacent frames. For the vertical movement, we assign only $\pm$ 5 degrees of continuous interference to the trajectory. In addition to horizontal and vertical movement, randomly zooming in and out is also included in our trajectory.

Occlusion Sampling. Based on each camera track, we can sample visible points and drop invisible points based on occlusion relationships. This generates a more realistic partial-view point cloud. Occlusion sampling is done per frame with the same camera intrinsic and different camera view-point. Given a camera with its view-point and intrinsic matrix, we need to first transform the point cloud in the world coordinate system into the camera coordinate system. suppose the intrinsic matrix is K, and the view-point is described by [R|t] where R is the rotation matrix and t is the translation vector. Then for a point $P = (x, y, z)^T$ in the world coordinate system, the position $\tilde{P} = (\tilde{x}, \tilde{y}, \tilde{z})^T$ in the camera coordinate system will be $\tilde{P}$ = K[R|t][P] where [P] is the screw representation $(x, y, z, 1)^T$. After the projection, we obtain a new 4D sequence that is realistic and natural. In the camera coordinate system, we will project the point clouds back into a depth image with 2D pixels. Invisible points will have the same 2D pixel coordinate as visible points but have a larger depth value. In this way we can determine the points to be occluded and get the partial-view depth image. Then we just map the depth image back into point clouds via the inverse view change transformation and get the partial point cloud. Repeating the above operation, we can easily obtain a large number of camera traces and the corresponding 4D partial-view sequences.

3.2. Asymmetric Distillation Architecture

Most of the current 4D backbone of state of the art is based on Transformer's architecture. Such expressive transformer architecture is quite challenging to optimize. We experimentally find it will negatively influence the teacher's features if we directly mirror the teacher network as such expressive architecture due to the optimization difficulty. With low-quality teacher features, the distillation performance also decreases. Based on this, we remove the transformer in the teacher network to weaken the teacher's expression ability. Although the teacher's expression ability is restricted, but it gets easier to optimize.

So unlike previous 2D/3D distillation work, we designed an asymmetric knowledge distillation framework to perform easy optimization. Specifically, we design the student network as a point 4D Conv layer followed by a transformer layer, while the teacher network contains only a point 4D Conv layer. The transformer in the student network is necessary to leverage motion information to find the geometric correlation, which can help construct complete geometry. While for the teacher branch, a teacher network with weaker expression ability facilitates a better distillation of knowledge for the student network. We observe experimentally that such asymmetric architecture introduces a positive influence on the distillation process. The ablation study also verifies our design usage.

3.3. Complete to Partial 4D Distillation

This section focuses on how to teach student networks to use motion cues to aggregate continuous geometric information for a more complete geometric understanding, and how to teach student networks to predict motion based on a better geometric understanding.

Based on this, we design a 4D-to-4D distillation framework as shown in Figure 4. On the one hand, our feature extraction backbone is a 4D backbone, and our goal is to teach it to efficiently and stably utilize spatial and temporal information in order to recover the underlying 3D real world. On the other hand, 4D means that the distillation source is not only a single-frame feature, but a window of features across time. From the single frame, we can only distill static geometric information, even if the backbone network is 4D. Whereas from a temporal window, the network can better reconstruct the geometry of the current frame from temporal cues, and can predict motion based on the geometric consistency of adjacent frames.

Frame-level Feature Extraction. The overall above process is executed at a frame-level feature. For nowadays state-of-the-art transformer-based 4D backbones, a point cloud will be divided into several tokens and feature communicates at the token-level.

We do the complete to partial distillation on the frame-level features instead of at the token or point level. This is due to the irregular representation and high down-sample ratio of the point cloud sequence. For the token level, it is very difficult to align tokens in different frames precisely under nowadays popular token-generating methods(use furthest point sampling to first generate anchor points and then ball-query to search near-neighbor points to extract token features). The error of alignment may seriously destroy the learning process. For the point level, a high down-sample ratio will make the point cloud even more sparse and irregular. The same region may get a totally different point cloud pattern after high ratio downsampling so it is very hard for the network to predict raw points directly. The frame-level information exchange allows for better learning of global geometric integrity and motion, which is critical for 4D downstream tasks.

Formally, given a complete point cloud sequence $S = \{s_1, s_2, ..., s_L\}$ and a partial point cloud sequence $\hat{S} = \{\hat{s}_1, \hat{s}_2, ..., \hat{s}_L\}$, the teacher network and student network will take these two sequences as input separately and get two sequences of per-frame features $F = \{f_1, f_2, ..., f_L\}$, $\hat{F} = \{\hat{f}_1, \hat{f}_2, ..., \hat{f}_L\}$. Then for the i-th frame of the partial sequence where the student feature is $\hat{f}_i$, its distillation source is $D = \{\dots, f_{i-1}, f_i, f_{i+1}, \dots\}$.

For geometry, the i-th frame features extracted by the student network need to directly learn from the i-th frame information of the teacher network. Since the features are extracted by a 4D backbone, this process actually encourages the network to reconstruct the complete geometry based on temporal cues. For time, the i-th frame extracted by the student network needs to have the ability to predict information to non-corresponding frames within a time window of the teacher network. Taking a time window of 3 as an example, we use two prediction heads to predict the i-1 frame and the i+1 frame of the teacher network from the i-th frame, respectively. This process encourages the network to learn the geometric correlation between adjacent frames through knowledge distillation of motional information.

4D Contrastive Learning Supervision. Since contrast learning shows excellent performance on the organized feature space, we use contrastive learning to supervise the feature learning process. For geometry, the i-th frame feature drawn by the student network is used as the anchor sample, and its positive sample is the i-th frame feature drawn by the teacher network. The other frames outside the time window form the negative sample pool. We assume our time window is $D = \{\dots, f_{i-1}, f_i, f_{i+1}, \dots\}$. We denote the set of

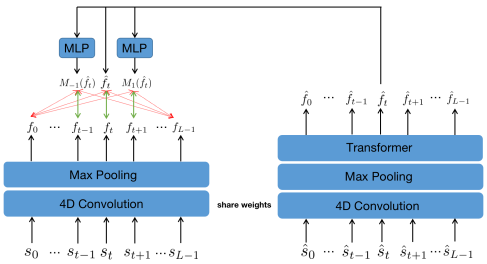

Figure 4. Objective of our proposed contrastive distillation.

frame indexes within the time window by W_i.

$$L_{geo} = -\sum_{i} log \frac{\exp(\hat{f}_i \cdot f_i / \tau)}{\sum_{j \notin W_i} \exp(\hat{f}_i \cdot f_j / \tau)} \tag{1}$$

For time, the i-th frame of the student network is used as the anchor, and the frame features within the time window are first obtained after the prediction head. As an example, the positive samples of frame i-1 and i+1 obtained from the predictor are the frame i-1 and i+1 features from the teacher's network, respectively, and the same for the other frames. The negative samples are the pool of features from all frames outside the time window.

$$L_{time} = -\sum_{i} \sum_{k=-1,1} log \frac{\exp(M_k(\hat{f}_i) \cdot f_{i+k} / \tau)}{\sum_{j \notin W_i} \exp(M_k(\hat{f}_i) \cdot f_j / \tau)} \tag{2}$$

where M_k indicates the predictor for the k-th frame within the time window and τ is the temperature coefficient.

So the final loss function is:

$$L_{total} = \alpha_1 L_{geo} + \alpha_2 L_{time} \tag{3}$$

where α_1, α_2 are the coefficient that $\alpha_1 + \alpha_2 = 1$. Note that we set both α_1 and α_2 to 0.5 in our experiments for simplicity.

By constructing the above objective, We encourage the student network to learn geometric completeness and temporal motility from the teacher network simultaneously.

4. Experiments

In this section, following other representation learning methods, we use the pre-trained network weights from each task as initialization and fine-tune them on 4D downstream tasks. The performance gain will be a good indicator of measuring the quality of the learned feature. In this section, we cover three 4D point cloud sequence understanding tasks: action segmentation on HOI4D [21], semantic segmentation on HOI4D [21], and 3D action recognition on MSR-action3D [17] in Section 4.2, 4.3 and 4.4 respectively. Result on Synthia 4D [27] semantic segmentation is introduced in the supplementary materials. For these three tasks, some basic settings will first be introduced in Section 4.1. In addition, we provide extensive ablation studies to validate our design choices in Section 4.5.

4.1. Basic setting

Partial-view Sequence Generating. To guarantee sufficient partial-view observations, we always make the camera go around the point cloud with a 150 degrees horizontal angle(θ) change. The horizontal angle change between two adjacent frames is the same. We set randomly the moving direction so it has the equal possibility for the camera to move from $\theta = -\frac{5}{12}\pi$ to $\frac{5}{12}\pi$ or $\theta = \frac{5}{12}\pi$ to $-\frac{5}{12}\pi$. Vertical angle(ϕ) change and zoom-in/zoom-out disturbance setting follow the strategy we introduce in Section 3.1.

Distillation Network. We set the time window size as 3 consisting of frames i-1, i, i+1 through all the tasks. P4Transformer [7] and PPTr [30] are used as the backbone by default. Little differences in the backbone across different tasks will be further clarified in each section. For the dynamic predictor, we use two fully connected layers together with one norm layer and a rectification layer. There are two dynamic predictors for predicting the i+1 frame and i-1 frame separately.

4.2. Fine-tuning on HOI4D action segmentation

Setup. To demonstrate the effect of our approach, we first conduct experiments on the HOI4D action segmentation task. For each point cloud sequence, we need to predict the action label for each frame. We follow the official data split of HOI4D with 2971 training scenes and 892 test scenes. Each sequence has 150 frames, and each frame has 2048 points. We use the version of PPTr without primitive fitting as the backbone, which is released by its author. However, due to the new emergence of the action segmentation task in 4D, there is no specific model for action segmentation of P4Transformer and PPTr before, we do a little change to the model of action recognition to adapt to the action segmentation task. The significant change is that per frame feature is formed before the temporal transformer, so attention is performed on frame features instead of token features as designed in the former backbone. And the output of the model will be a sequence of labels instead of one label. We have verified this design is more effective from the backbone design view. All experiments in action segmentation will use this model design. We also conduct experiments with other 4D pre-training strategies including STRL(a 4D spatial-temporal contrastive learning method) and VideoMAE(an MAE-based method for video representation learning which can be easily extended to 4D data). More introduction about STRL and VideoMAE can be found in the supplementary materials. The following metrics are reported: framewise accuracy (Acc), segmental edit distance, as well as segmental F1 scores at the over-

Table 1. Action segmentation on HOI4D dataset [21]

Method	Frames	Acc	Edit	F1@10	F1@25	F1@50
P4Transformer [7]	150	71.2	73.1	73.8	69.2	58.2
P4Transformer+C2P [7]	150	73.5	76.8	77.2	72.9	62.4
PPTr [30]	150	77.4	80.1	81.7	78.5	69.5
PPTr+STRL [16]	150	78.4	79.1	81.8	78.6	69.7
PPTr+VideoMAE [9]	150	78.6	80.2	81.9	78.7	69.9
PPTr+C2P [30]	150	**81.1**	**84.0**	**85.4**	**82.5**	**74.1**

Table 2. Semantic segmentation on HOI4D dataset [21]

Method	Frames	mIoU
P4Transformer [7]	3	40.1
P4Transformer+C2P [7]	3	41.4
PPTr [30]	3	41.0
PPTr+STRL [16]	3	41.2
PPTr+VideoMAE [9]	3	41.3
PPTr+C2P [30]	3	**42.3**

lapping thresholds of 10%, 25%, and 50%. Overlapping thresholds are determined by the IoU ratio.

Result. As reported in Table 1, our method has big improvement on both two backbones. For the state-of-the-art backbone PPTr, it can be seen that our method consistently outperforms STRL and VideoMAE by a big margin for all metrics. Considering STRL, its short sequence augmentation can not well guide the model to notice motion cues so it is very hard to leverage temporal information which is actually very important in action segmentation tasks. Simple extension of VideoMAE to point cloud sequence also shows very little improvement. Unlike video pixel tokens which are regular and compact, high down-sampled point cloud tokens have very irregular and sparse patterns. It is hard for the model to learn to predict raw points. Also notice that VideoMAE do self-supervised learning on a point level, which is very hard to learn motion features. Our method emphasizes motion information by cross-time distillation and the learning process is done on a stable frame feature level which finally comes in a satisfying result. A more straight forward visualization result is shown in the supplementary materials.

4.3. Fine-tuning on HOI4D semantic segmentation

Setup. To verify that our approach can also be effective on fine-grained tasks, we conduct further experiments on HOI4D for 4D semantic segmentation. Since pre-training methods generally benefit from large data, unlike previous papers, we use the full set of HOI4D for our experiments. The dataset consists of 3863 4D sequences, each including 300 frames of point clouds, for a total of 1.158M frames of point clouds. For one frame, there are 8192 points. We follow the official data split of HOI4D with 2971 training scenes and 892 test scenes. We use the version of PPTr without primitive fitting as the backbone, which is released by its author. During representation learning and training/fine-tuning, we randomly select 1/5 of the whole data to form one epoch for efficient training. We use mean IoU(mIoU) % as the evaluation metric and 39 category labels are used to calculate it.

Considering our representation learning method prefers long sequence which has relatively abundant temporal information while the limitation of GPU memory, we set the sequence length as 10 and num points per frame as 4096. Fine-tuning and testing are performed on sequence length of 3 to be consistent with the baseline.

Result. As reported in Table 2, there is still a performance improvement on the 4D semantic segmentation task, which also shows the effectiveness of our approach for fine-grained feature understanding. The very small improvement that STRL can provide suggests that simple data augmentation at the scene-level does not help much in learning fine-grained features. VideoMAE performs one convolution layer on pixels to form token features so it is very easy to match the token with the raw points. However, there are several convolution layers in the semantic segmentation model so it is very hard to figure out the corresponding token and the raw points which may cause serious information leakage that hurt the pretraining. Compared with previous methods, our method can effectively extract features with high representational and generalization capabilities by introducing 4D distillation.

4.4. Fine-tuning on 3D Action Recognition

Setup. Following P4Transformer and PPTr, we use the MAR-Action3D dataset, which consists of 567 human point cloud videos, including 20 action categories. Each frame is sampled with 2,048 points. The point cloud videos are segmented into multiple segments. During training, video-level labels are used as segment-level labels. To estimate the video-level probabilities, we take the mean of all segment-level probability predictions. To be able to compare with the best performance, We fit the human point cloud to 4 primitives and then use PPTr as our 4D backbone. Our pretraining is done on the sequence of length 24, and the pre-trained model weights are used for all the other sequence lengths.

Result. The results are shown in Table 3. We can observe that we improve the performance by a large margin for different sequence lengths which demonstrates the effectiveness of our method. We observe that STRL and VideoMAE also don not have a significant improvement as the same in action segmentation. This indicates that our method significantly outperforms the existing methods in terms of the ability to extract global features of sequences.

4.5. Analysis Experiments and Discussions.

In this section, we first conduct an ablation study to verify the design of complete to partial. We then discuss the necessity of asymmetric distillation frameworks. We also compare the effects of random sampling and sampling

Table 3. Action recognition on MSR-Action3D dataset [17]

Method	Input	Frames	Video Acc@1
PointNet++ [25]	point	1	61.61
MeteorNet [19]	point	8	81.14
	point	16	88.21
	point	24	88.50
PSTNet [8]	point	8	83.50
	point	16	89.90
	point	24	91.20
P4Transformer [7]	point	8	83.17
	point	16	89.56
	point	24	90.94
PPTr [30]	point	8	84.02
	point	16	90.31
	point	24	92.33
PPTr+STRL [16]	point	24	92.66
PPTr+VideoMAE [9]	point	24	92.66
PPTr+C2P	point	8	87.16
	point	16	91.89
	point	24	**94.76**

based on natural camera trajectories. We use HOI4D Action Segmentation as the downstream task and PPTr as the default backbone.

Necessity of complete to partial distillation. We propose a complete-to-partial distillation framework, where the teacher network uses complete point cloud sequences to teach the student network that has only seen partial-view sequences. This knowledge distillation not only encourages networks to use temporal cues to achieve an understanding of the complete geometry, but also to use geometric consistency to capture motion information. This is a capability that neither the complete-to-complete nor the partial-to-partial distillation frameworks possess. On the one hand, geometric complementarity between multiple frames from the same data is difficult to help the understanding of the current frame due to the lack of teacher material that can be used as input for teacher network. On the other hand, the teacher network and the student network have seen almost the same data with little point cloud variation, which greatly reduces the difficulty of the self-supervised task. All these problems hinder the integration of spatio-temporal information. To demonstrate this, we experiment with three different distillation strategies. Frame-wise accuracy achieved by the three strategies are 79.48, 79.66, 81.10 respectively as shown in Table 4, and our Complete-to-Partial strategy achieves the best performance.

Table 4. Comparison of different distillation strategies.

Methods	Frame-wise accuracy
Complete-to-Complete	79.48
Partial-to-Partial	79.66
Complete-to-Partial	**81.10**

Asymmetric versus symmetric framework. One of the more popular frameworks for knowledge distillation is the use of symmetric twin networks, where the teacher network and the student network are the same and share weights. In our setup, we remove the transformer layer in the teacher network compared with the student network. This is because transformer is difficult to optimize especially with limited 4D data, and the existence of two transformers will introduce extreme optimization difficulty. We believe that a teacher network with weaker representational abilities is able to facilitate the student network to distill knowledge. And we expect that the student should be better than the teacher network, i.e., have stronger learning and representational capabilities than the teacher network. To verify this, we re-trained the teacher network using a symmetric twin network, i.e., adding a Transformer layer to the teacher network. The results show that only an accuracy of 78.95 can be obtained, which is 2.15 lower than the asymmetric structure. the performance degradation is probably due to the optimization difficulty of the teacher network during the training process.

Partial-view generation strategy. We design a partial-view sequence generation method that simulates camera motion in the natural world, i.e., first generating camera trajectories and then sampling visible points according to occlusion relations. We believe that this approach can help the student network generate effective learning input. With this natural sequence, the student network can make full use of the temporal information to understand the complete geometric, and can use the learned geometry to understand the motion of the object/scene. The learning inputs we generate are significantly more meaningful and effective than those obtained by random sampling. The random sampling approach sacrifices the geometric consistency of the point cloud and the reality of the data, both of which have a negative impact on knowledge distillation. We conducted experiments to verify this. Using random sampling to generate 4D point cloud sequences, the same framework can only achieve an accuracy of 80.22. This verifies that our partial-view generation approach is reasonable and effective.

5. Conclusions

In this paper, we propose Complete-to-Partial 4D Distillation, a new pre-training method for point cloud sequence representation learning. Our main idea is to distill the spatial-temporal information of a complete point cloud sequence into a partial point cloud sequence. Experiments show our proposed method significantly outperforms previous pre-training approaches on a wide range of point cloud sequence understanding tasks. Although the pre-training of point cloud sequences is still at an early stage, this problem is undoubtedly very important and challenging. Our work is encouraging and suggests future work to explore more possible design for 4D representation learning.

References

[1] Daniel M. Bear, Chaofei Fan, Damian Mrowca, Yunzhu Li, Seth Alter, Aran Nayebi, Jeremy Schwartz, Li Fei-Fei, Jiajun Wu, Joshua B. Tenenbaum, and Daniel L. K. Yamins. Learning physical graph representations from visual scenes. *CoRR*, abs/2006.12373, 2020. 2

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709, 2020. 1

[3] Ye Chen, Jinxian Liu, Bingbing Ni, Hang Wang, Jiancheng Yang, Ning Liu, Teng Li, and Qi Tian. Shape self-correction for unsupervised point cloud understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8382–8391, 2021. 2

[4] Yujin Chen, Matthias Nießner, and Angela Dai. 4dcontrast: Contrastive learning with dynamic correspondences for 3d scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1, 3

[5] Christopher B. Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. *CoRR*, abs/1904.08755, 2019. 2

[6] Ishan R. Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah. TCLR: temporal contrastive learning for video representation. *CoRR*, abs/2101.07974, 2021. 1

[7] Hehe Fan, Yi Yang, and Mohan S. Kankanhalli. Point 4d transformer networks for spatio-temporal modeling in point cloud videos. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 14204–14213, 2021. 1, 2, 6, 7, 8

[8] Hehe Fan, Xin Yu, Yuhang Ding, Yi Yang, and Mohan Kankanhalli. Pstnet: Point spatio-temporal convolution on point cloud sequences. In *International conference on learning representations*, 2020. 1, 2, 8

[9] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners, 2022. 1, 7, 8

[10] Kexue Fu, Peng Gao, Renrui Zhang, Hongsheng Li, Yu Qiao, and Manning Wang. Distillation with contrast is all you need for self-supervised point cloud representation learning. *arXiv preprint arXiv:2202.04241*, 2022. 3

[11] Shi Hanyu, Wei Jiacheng, Wang Hao, Liu Fayao, and Lin Guosheng. Learning spatial and temporal variations for 4d point cloud segmentation, 2022. 1

[12] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. *CoRR*, abs/2111.06377, 2021. 1, 2

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. *CoRR*, abs/1911.05722, 2019. 1, 2

[14] 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

[15] Deng Huang, Wenhao Wu, Weiwen Hu, Xu Liu, Dongliang He, Zhihua Wu, Xiangmiao Wu, Mingkui Tan, and Errui Ding. Ascnet: Self-supervised video representation learning with appearance-speed consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8096–8105, 2021. 1

[16] Siyuan Huang, Yichen Xie, Song-Chun Zhu, and Yixin Zhu. Spatio-temporal self-supervised representation learning for 3d point clouds. *arXiv preprint arXiv:2109.00179*, 2021. 1, 3, 7, 8

[17] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Action recognition based on a bag of 3d points. In *2010 IEEE computer society conference on computer vision and pattern recognition-workshops*, pages 9–14. IEEE, 2010. 2, 6, 8

[18] Yangbin Lin, Cheng Wang, Dawei Zhai, Wei Li, and Jonathan Li. Toward better boundary preserved supervoxel segmentation for 3d point clouds. *ISPRS journal of photogrammetry and remote sensing*, 143:39–47, 2018. 2

[19] 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, 8

[20] Yunze Liu, Qingnan Fan, Shanghang Zhang, Hao Dong, Thomas Funkhouser, and Li Yi. Contrastive multimodal fusion with tupleinfonce. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 754–763, 2021. 2

[21] Yunze Liu, Yun Liu, Che Jiang, Zhoujie Fu, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, He Wang, and Li Yi. HOI4D: A 4D Egocentric Dataset for Category-Level Human-Object Interaction. *arXiv e-prints*, Mar. 2022. 1, 2, 6, 7

[22] Yang Liu, Keze Wang, Lingbo Liu, Haoyuan Lan, and Liang Lin. TCGL: temporal contrastive graph for self-supervised video representation learning. *CoRR*, abs/2112.03587, 2021. 1

[23] 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. 1, 2

[24] Yatian Pang, Wenxiao Wang, Francis E. H. Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning, 2022. 1, 2

[25] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 8

[26] Yongming Rao, Benlin Liu, Yi Wei, Jiwen Lu, Cho-Jui Hsieh, and Jie Zhou. Randomrooms: Unsupervised pre-training from synthetic shapes and randomized layouts for 3d object detection. *CoRR*, abs/2108.07794, 2021. 2

[27] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3234–3243, 2016. 2, 6

[28] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain. Time-contrastive networks: Self-supervised learning from video. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 1134–1141. IEEE, 2018. 1

[29] Hanchen Wang, Qi Liu, Xiangyu Yue, Joan Lasenby, and Matthew J. Kusner. Unsupervised point cloud pre-training via occlusion completion. In *International Conference on Computer Vision, ICCV*, 2021. 2

[30] Hao Wen, Yunze Liu, Jingwei Huang, Bo Duan, and Li Yi. Point primitive transformer for long-term 4d point cloud video understanding. In *European Conference on Computer Vision*, pages 19–35. Springer, 2022. 1, 2, 6, 7, 8

[31] Saining Xie, Jiatao Gu, Demi Guo, Charles R. Qi, Leonidas J. Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. *CoRR*, abs/2007.10985, 2020. 1, 2

[32] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[33] Renrui Zhang, Ziyu Guo, Rongyao Fang, Bin Zhao, Dong Wang, Yu Qiao, Hongsheng Li, and Peng Gao. Point-m2ae: Multi-scale masked autoencoders for hierarchical point cloud pre-training, 2022. 1, 2

[34] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. *arXiv preprint arXiv:2112.02413*, 2021. 1, 2

[35] Zaiwei Zhang, Rohit Girdhar, Armand Joulin, and Ishan Misra. Self-supervised pretraining of 3d features on any point-cloud. 2021. 2