A Unified Framework for Human-centric Point Cloud Video Understanding

Yiteng Xu[1], **Kecheng Ye**[1], **Xiao Han**[1], **Yiming Ren**[1], **Xinge Zhu**[2], **Yuexin Ma**[1,*]
[1] ShanghaiTech University [2] The Chinese University of Hong Kong
{xuyt2023,mayuexin}@shanghaitech.edu.cn

Abstract

Human-centric Point Cloud Video Understanding (PVU) is an emerging field focused on extracting and interpreting human-related features from sequences of human point clouds, further advancing downstream human-centric tasks and applications. Previous works usually focus on tackling one specific task and rely on huge labeled data, which has poor generalization capability. Considering that human has specific characteristics, including the structural semantics of human body and the dynamics of human motions, we propose a unified framework to make full use of the prior knowledge and explore the inherent features in the data itself for generalized human-centric point cloud video understanding. Extensive experiments demonstrate that our method achieves state-of-the-art performance on various human-related tasks, including action recognition and 3D pose estimation. ***All datasets and code will be released soon.***

1. Introduction

Human-centric point cloud video understanding (PVU) is a burgeoning field focused on discerning, interpreting, and quantifying human-related information within sequences of human point clouds. This area has witnessed a surge in attention in recent years, particularly applied in LiDAR captured large-scale unconstrained scenarios [5, 11, 33, 34]. Its significance lies in its critical role in facilitating various downstream tasks, including human action recognition [33], 3D pose estimation [4], motion capture [13, 24], etc. These advancements hold the potential to further drive progress in real-world applications, such as intelligent surveillance, assistive robots, human-robot collaboration, etc.

Current methods [13, 24, 33] usually rely on extensive labeled data for supervision and employ generic point cloud-based feature extraction backbones [10, 17, 21, 22]. Nevertheless, obtaining the necessary data and annotations for 4D human-centric point cloud videos proves to be a challenging and expensive endeavor. Furthermore, fully supervised techniques tend to exhibit overfitting issues when applied to specific datasets or tasks, resulting in limited generalization capabilities. Additionally, the existing feature extraction networks are ill-suited for human-centric data, as they fail to account for human-specific characteristics. Hence, within the domain of human-centric PVU, the significance of self-supervised learning becomes evident in enhancing algorithmic generalization. Simultaneously, the development of a human-specific feature extractor that uses prior human-related knowledge holds great promise in bolstering the effectiveness of methods for downstream tasks.

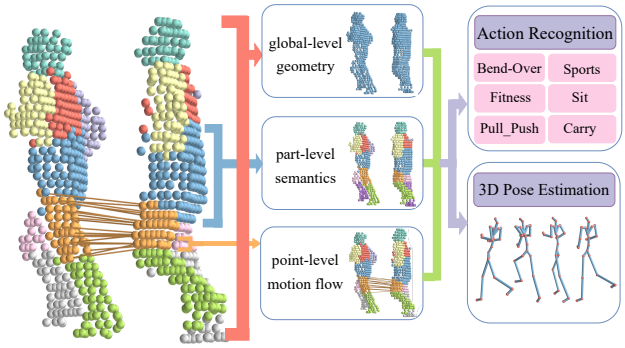

Figure 1. UniPVU-Human extracts human-related prior knowledge at global level, part level, and point level to facilitate subsequent geometric and dynamic representation learning, finally cater to a range of downstream human-centric tasks, such as action recognition, 3D pose estimation, etc.

Actually, self-supervised learning [31, 39] for PVU has made great progress. Some approaches [26–28] leverage contrastive learning techniques to capture essential spatio-temporal features within dynamic point clouds. Nevertheless, due to the inherent challenges posed by irregular point distributions stemming from varying capture distances, occlusions, and noise, the construction of high-

*Corresponding author. This work was supported by NSFC (No.62206173), Shanghai Sailing Program (No.22YF1428700), MoE Key Laboratory of Intelligent Perception and Human-Machine Collaboration (ShanghaiTech University), Shanghai Frontiers Science Center of Human-centered Artificial Intelligence (ShangHAI), Shanghai Engineering Research Center of Intelligent Vision and Imaging.

quality positive and negative samples remains a nontrivial task, thereby making the optimization process difficult. The latest work [25] exploits mask prediction for point cloud video self-learning by dividing sequential point clouds into tubes for masking and recovering. However, this method flattens all tubes for feature learning, inadvertently compromising the semantic and dynamic consistency in 4D videos. Moreover, all these methods are not tailored specifically to address human-centric PVU.

Given the importance of human-centric tasks, the imperative need arises to establish a unified framework for the understanding of human point cloud videos. Notably, no specific solutions to this challenge have been identified to date. In this paper, we approach the problem by addressing two fundamental questions: *first, what human-related prior knowledge can be extracted, and second, how can the knowledge be harnessed to enhance human-centric representation learning*?

Considering the inherent structure of the human body, characterized by fixed components such as torso, head, arms, and legs, as well as the distinctive dynamic traits exhibited during human motion, we exploit both **the structural semantics of human body and human motion dynamics** to facilitate the acquisition of human-specific features from sequences of point clouds. In particular, we create two large-scale point-cloud-based datasets and corresponding pre-trained networks for body segmentation and motion flow estimation, respectively, so that human prior knowledge can be learned in advance and assist subsequent representation learning. Furthermore, within our framework, we introduce two innovative stages tailored to maximize the utility of this prior knowledge. The first one, termed **semantic-guided spatio-temporal representation self-learning**, incorporates a body-part-based mask prediction mechanism designed to facilitate the acquisition of geometric and dynamic representations of humans in the absence of annotations. Building upon this foundation, the following stage, **hierarchical feature enhanced fine-tuning**, integrates and adapts global-level, part-level, and point-level point cloud features to cater to a range of downstream tasks. In this way, our approach, named **UniPVU-Human**, serves as a comprehensive exploration of human prior knowledge, furnishing a unified framework for the effective learning of human-centric representations.

To evaluate the effectiveness of our method, we conduct extensive experiments on two popular LiDAR-point-cloud-based datasets [24, 33], focusing on human action recognition and human pose estimation, respectively. Our method achieves state-of-the-art performance on both tasks. Detailed ablation studies are also provided to verify each stage and technical design in our framework.

To summarize, our contributions are as follows:

1. We propose UniPVU-Human, the first unified framework for human-centric point cloud video understanding, which is significant for vast downstream applications.
2. Containing two novel stages, including semantic-guided spatio-temporal representation self-learning and hierarchical feature enhanced fine-tuning, our method fully takes advantage of prior knowledge of humans for effective and robust human-centric representation learning.
3. Our method achieves state-of-the-art performance on open datasets for various human-centric tasks.

2. Related Work

2.1. Feature Learning for Point Clouds

Point cloud is an important representation for 3D scenes and objects, and tremendous efforts have been made [17, 23] for extracting valuable features from point clouds. PointNet [21] is to learn a spatial encoding of each point and then aggregate all individual point features to a global point cloud signature[22]. PointNet++[22] further introduces a hierarchical feature learning paradigm to capture the local geometric structures recursively. PointNeXt [23] revisits PointNet++[22] with improved training and scaling strategies. PCT [10] and PointTransformer [40] apply attention-based mechanism [30] to point cloud representations. Subsequently, many methods [2, 2, 9, 15] extend them to process dynamic point cloud videos for more extensive applications. P4Transformer [7] and PST-Transformer [8] use transformers among all local 4D tubes' features to capture long relationships. With the development of autonomous driving, some works [35, 41–43] propose voxel-based or voxel-point-based feature extractors to process LiDAR point clouds for high efficiency. However, all these methods are not specifically designed for human dynamic point clouds, lacking the consideration of human-specific characteristics.

2.2. LiDAR-based Human-centric Understanding

Recently, the understanding of human-centric point cloud videos, which are captured by LiDARs in large-scale scenes, has become an emerging field with a lot of new datasets and benchmarks for various human-centric tasks. LiDARCap [13] contributes an in-the-wild human motion dataset and proposes a LiDAR point cloud video-based motion capture framework. Subsequent works, LIP [24], explores the feature fusion of different visual sensors to address 3D pose estimation task based on point clouds and images. Recently, HuCenLife [33] proposes a huge human-centric dataset with diverse daily-life scenarios and rich human activities, and provides baselines for human perception, action recognition, motion prediction, etc. However, all these approaches follow previous generic backbones to extract features from sequence human point clouds without

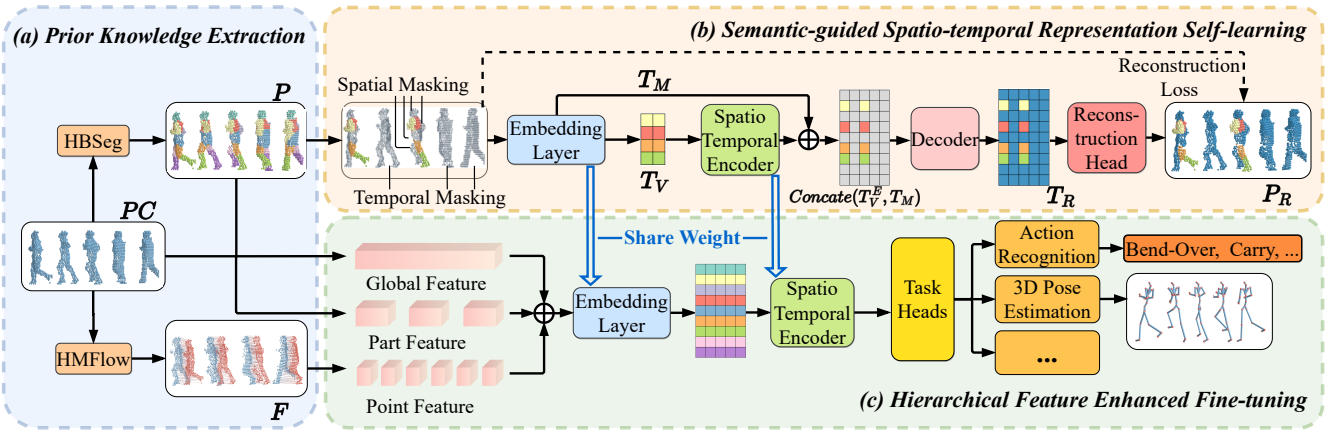

Figure 2. The main pipeline of UniPVU-Human, which can be divided into three stages, including (a) Prior Knowledge Extraction, (b) Semantic-Guided Spatio-temporal Representation Self-learning, and (c) Hierarchical Feature Enhanced Fine-tuning. First, the pre-trained HBSeg and HMFlow are used to provide geometric and dynamic information, including body part segmentation results and point-wise motion flow. Then, our self-learning stage incorporates a body-part-based mask prediction mechanism designed to facilitate the acquisition of geometric and dynamic representations of humans in the absence of annotations. Finally, we integrate global-level, part-level, and point-level features to boost the knowledge transfer to downstream tasks in the fine-tuning stage.

making use of human prior knowledge. Moreover, they are all supervised methods, causing unsatisfactory results when generalizing to other datasets or tasks.

2.3. Self-supervised Learning for Point Clouds

To understand the point cloud representation from data itself instead of supervision by manual annotations, some methods improve the generalization capability via self-learning in the pre-training stage. There exist many methods [1, 19, 32, 36, 38] learning the geometric representation from static point cloud in a self-supervised manner. Recently, more and more self-learning methods [31, 39] are proposed to learn the spatio-temporal representations of point cloud videos. Some approaches [26–28] adopt contrastive learning spatially and among frames to learn inherent geometric and dynamic features. However, LiDAR point clouds are sparsity-varying across different capture distances, are usually incomplete due to occlusions, and contain undesired noises, making these methods unstable due to low-quality positive and negative pairs. In the recent study [25], mask prediction [12] is employed for self-learning in point cloud videos by segmenting sequential point clouds into tubes. However, it flattens all tubes during feature learning, inadvertently affecting the semantic and dynamic consistency in 4D videos. Additionally, all these methods lack customization for the specific requirements of human-centric point cloud video understanding.

3. Method

The whole architecture of our method is presented in Figure. 2. A point cloud video sequence for human instances is denoted as $PC \in \mathcal{R}^{L\times N\times D}$, where L is the sequence length, N is number of points in each frame, and D is the dimension of each point. To exploit both the structural semantics of human body and human motion to facilitate the acquisition of human-specific features from sequences of point clouds, we create two large-scale point-cloud-based datasets and corresponding pre-trained networks for body segmentation and motion flow estimation in **Prior Knowledge Extraction**. Besides, to learn the essential geometric and dynamic representations of humans from data itself, we explore the spatial and temporal relationships of structural semantics in human body by applying spatio-temporal modeling upon the embedding of body parts in the stage of **Semantic-guided Spatio-temporal Representation Self-learning**. Building upon this foundation, we use extracted multiple levels of human-related prior knowledge to benefit various downstream tasks by **Hierarchical Feature Enhanced Fine-tuning**, which integrates global-level, part-level, and point-level point cloud features.

3.1. Prior Knowledge Extraction

Different from the movements of rigid objects such as vehicles in traffic scenarios, the motion of non-rigid humans is more complicated, for it contains not only global rotations and translations but also local rotations and translations, such as relative motions among joints, making capturing human motion features more challenging. The key to addressing this problem is to model the human motion in more fine-grained levels, including body part level and point-wise level. Therefore, we build the Human Body Segmentation (HBSeg) and Human Motion Flow (HMFlow) networks to provide more fine-grained geometric and motion information about human body, which can serve as prior knowledge

to facilitate following human-centric representation learning.

3.1.1 Human Body Segmentation (HBSeg)

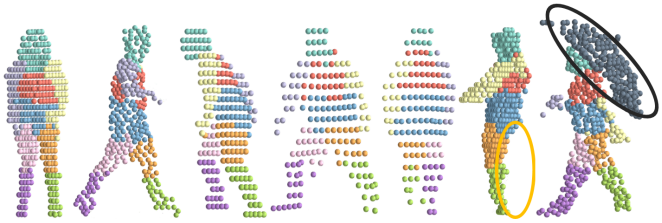

Figure 3. Visualization results of HBSeg on HuCenLife [33]. We show cases with different densities of LiDAR point cloud, occlusion (yellow circle), and noise (black circle). HBSeg has robust performance even merely trained on our synthesized dataset.

To fully utilize the structural semantics of human body, we deconstruct human into fine-grained body parts by HBSeg, pre-trained on our synthetic dataset, which is constructed by employing a simulated LiDAR model [24] to scan the surfaces of 3D human meshes from the AMASS dataset [18] at various perspectives and distances, meanwhile introducing random occlusions and noise to minimize the distribution gap between synthetic data and real data. We define 9 parts of human body and generate annotations by attaching the body part label of the nearest SMPL [16] mesh vertex to the synthetic LiDAR Point. More details are in Section. 4.1 and supplementary material.

We adopt the PointNeXt-L [23] as the main body network of HBSeg. After training on synthetic data, we apply the pre-trained HBSeg on real data to get the part segmentation labels $S \in \mathcal{R}^{L\times N\times 1}$. As Figure. 3 shows, HBSeg performs stable on real-life data with changing sparsity and works well even for occlusion and noise cases, mainly due to our efforts on generating realistic synthetic data.

3.1.2 Human Motion Flow Estimation (HMFlow)

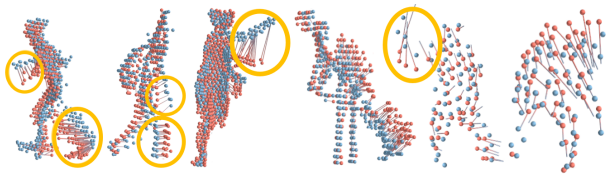

Figure 4. Visualization results of HMFlow on HuCenLife [33]. We present several cases from near to far relative to the LiDAR sensor. HMFlow has good capability of estimating point flow even for the parts with significant movements (yellow circle), which can provide explicit features of human dynamics.

We pre-train the HMFlow on our synthetic dataset to provide point-wise motion information, which can benefit the feature enhancement in the fine-tuning stage. Similar to Section. 3.1.1, we associate each synthetic LiDAR point to its nearest SMPL vertex. Therefore, we are able to establish the correspondence between synthetic LiDAR points across different frames by using SMPL vertices indices, so that we can obtain motion flow ground truth for training our HMFlow. More details are in Section. 4.1 and supplementary material.

We employ FLOT [20] as the human motion flow estimator, which casts the task of scene flow estimation as finding soft correspondences on a pair of point clouds via solving an optimal transport problem [14]. When testing on the real data, we input adjacent LiDAR point clouds PC_t and $PC_{t+1} \in \mathcal{R}^{L\times N\times D}$ into the pre-trained HMFlow to obtain human motion flow vectors $F \in \mathcal{R}^{L\times N\times D'}$ for each LiDAR point in the t-th frame. As Figure. 4 demonstrates, HMFlow can generate reasonable prediction for point-wise motion flow on real data even without annotations, which is valuable to provide explicit priors for human dynamics.

3.2. Semantic-guided Spatio-temporal Representation Self-learning

Due to spatial irregularities and temporal redundancies, annotating dynamic point cloud videos is labor-intensive and error-prone. Moreover, fully-supervised methods usually overfit to manual annotations in specific domains, struggling to capture the underlying patterns of new data [37]. This limitation leads to a restricted ability to generalize to other tasks or datasets. Additionally, existing feature extraction networks employ generic point-cloud-based backbones that are ill-suited for human-centric data, as they do not account for human-specific characteristics. Based on structure semantics of human bodies obtained in previous stage, we propose a module, named Semantic-guided Spatio-temporal Representation Self-learning, which mines essential geometric and motion features from human point cloud video data itself by masking and predicting body part patches to enhance the generalization ability of the model to benefit a variety of downstream tasks.

As Figure. 2 shows, We first mask some of the body-part tokens in temporal and spatial dimensions. After extracting the token embedding of part patches, only visible tokens will be input to the STEncoder to extract latent representation, which will be decoded with masked tokens together to reconstruct the 3D Cartesian coordinates of masked part patches. Details are as follows.

3.2.1 Spatio-temporal Masking Strategy

In temporal dimension, we mask all part patches in some random frames and reconstruct the masked tokens, encour-

aging STEncoder networks to estimate the part motion over a long period. In spatial dimension, we randomly mask some part patches in the remaining frames after temporal masking, making the STEncoder network estimate the spatial geometric features of the entire human based on the visible tokens.

We apply temporal masking first, and then spatial masking. Given part patches $P \in \mathcal{R}^{L\times M\times N'\times D}$, we adopt a temporal mask ratio r_t and spatial mask ratio r_s, respectively. Firstly, all part patches in $r_t L$ frames are masked, named as temporal masked patches $P_M^t \in \mathcal{R}^{r_t L\times M\times N'\times D}$, which will be used as the reconstruction ground truth of temporal masking. For every frame of the remaining $(1-r_t)L$ visible frames, $r_s M$ part patches are masked randomly, hence spatial masked patches $P_M^s \in \mathcal{R}^{(1-r_t)L\times r_s M\times N'\times D}$ will be used as the reconstruction ground truth of the spatial masking.

3.2.2 Embedding Layer

For visible part patches, we use Mini-PointNet [21] as a tokenizer to obtain visible part tokens embedding T_V from visible part patches P_V.

$$T_V = Tokenizer(P_V), \tag{1}$$

where $T_V \in \mathcal{R}^{(1-r_t)L\times(1-r_s)M\times C}$, $P_V \in \mathcal{R}^{(1-r_t)L\times(1-r_s)M\times N'\times D}$. C is the channel dimension of part tokens. For every masked part patch, we replace it with a share-weighted learnable masked part token T_M, which will be concatenated with the output of STEncoder and processed by the decoder together.

Learnable spatial positional encoding and temporal positional encoding will also be added to the input of every transformer layer in the STEncoder and decoder.

3.2.3 Spatio-temporal Encoder (STEncoder)

To fully utilize the inherent structure of the human body, characterized by fixed components such as torso, head, arms, and legs, as well as the distinctive dynamic traits exhibited during human motion, our STEncoder applies spatial modeling and temporal modeling on body part tokens, respectively. The STEncoder learns to extract the high-level latent geometric and motion features of humans from only T_V, which are input to STEncoder to get enhanced visible part tokens T_V^E.

For the network design of STEncoder, we interlace multiple spatial transformer [30] layers and temporal transformer layers to extract the spatial geometry feature and temporal motion feature, respectively. For each spatial transformer layer, we apply self-attention [30] among all visible parts tokens in every frame:

$$V_T^{s'} = SpatialTransformer(V_T^s), \tag{2}$$

where $V_T^s, V_T^{s'} \in \mathcal{R}^{(1-r_s)M\times C}$ are the visible part tokens in a frame. For each temporal transformer layer, we apply self-attention for every visible part token among all L frames:

$$V_T^{t'} = TemporalTransformer(V_T^t), \tag{3}$$

where $V_T^t, V_T^{t'} \in \mathcal{R}^{(1-r_t)L\times C}$ are the visible part tokens of a body part in L frames.

3.2.4 Mask Reconstruction

Our decoder is similar to the STEncoder with fewer layers. We take the enhanced visible part tokens T_V^E as well as masked part token T_M as the input of the decoder.

$$T_R = Decoder(Concate(T_V^E, T_M)), \tag{4}$$

where $T_R \in \mathcal{R}^{L\times M\times C}$ is reconstructed part tokens.

Among the T_R, only the tokens masked before will be fed to the reconstruction head to predict the original masked part patches. The structure of the reconstruction head is similar to that in Point-MAE [19], which is a fully connected (FC) layer with a reshape operation.

$$\begin{aligned} P_R^s &= Reshape(FC(T_R^s)), \\ P_R^t &= Reshape(FC(T_R^t)), \end{aligned} \tag{5}$$

where $P_R^s \in \mathcal{R}^{(1-r_t)L\times r_s M\times N'\times D}$, $T_R^s \in \mathcal{R}^{(1-r_t)L\times r_s M\times C}$, $P_R^t \in \mathcal{R}^{r_t L\times M\times N'\times D}$, $T_R^t \in \mathcal{R}^{r_t L\times M\times C}$ are spatial reconstructed part patches, spatial reconstructed part tokens, temporal reconstructed part patches, temporal reconstructed part tokens, respectively. We first reconstruct the spatial masked tokens, and then the temporal masked tokens. We adopt Chamfer Distance [6] Loss L_{CD} as the reconstruction loss function:

$$\begin{aligned} L_{CD}(P_M, P_R) = &\frac{1}{\|P_M\|}\sum_{x\in P_M}\min_{y\in P_R}\|x-y\|_2^2 \\ &+\frac{1}{\|P_R\|}\sum_{y\in P_R}\min_{x\in P_M}\|y-x\|_2^2. \end{aligned} \tag{6}$$

3.3. Hierarchical Feature Enhanced Fine-tuning

After the above self-learning process, STEncoder is endowed with the ability to extract the representations of humans in part-level structural semantics. When fine-tuning on multiple downstream tasks, hierarchical features can enhance the STEncoder to capture more complicated and challenging fine-grained geometric and dynamic representations. Specifically, we integrate global-level, part-level, and point-level point cloud features to pre-trained STEncoder (See Figure. 2), therefore fully leveraging prior knowledge for effective and robust human-centric representation learning.

During this stage, all part patches are visible to the STEncoder, and information of point-wise motion flow vector will be fused to that of body parts in Tokenizer. (See details in supplementary materials)

$$T = Tokenizer(P, F), \tag{7}$$

where $T \in \mathcal{R}^{L\times M\times C}, P \in \mathcal{R}^{L\times M\times N'\times D}, F \in \mathcal{R}^{L\times M\times N'\times D'}$ are part tokens, part patches, and motion flow vectors, respectively. We will also append a global token extracted from the entire human instance to enable the interaction of features between global and part. For classification tasks like action recognition, a class token will be appended as well. we discard the decoder and add corresponding task heads after the pre-trained tokenizer, learnable position encoding, and STEncoder for different tasks.

4. Experiments

To evaluate the effectiveness of our method, we conduct experiments on open datasets on tasks of human action recognition and human pose estimation. Extensive ablation studies are also conducted for the comprehensive evaluation of modules and technical designs of our method.

4.1. Datasets

In this section, we first introduce our two synthetic datasets for body segmentation and motion flow estimation. Then we give details for two open datasets, which are used for evaluating our unified framework on downstream real-life human-centric tasks.

Human Body Segmentation Synthetic Dataset. To address the absence of 3D human body part segmentation datasets based on LiDAR point clouds, we leverage SMPL mesh from AMASS [18] to simulate 1 million LiDAR human point cloud instances following LIP [24]. We automatically label the data by utilizing the SMPL mesh properties. Specifically, since ordered and regular SMPL mesh vertices provide 24 human body part labels, we can automatically assign each simulated LiDAR point the label of its nearest vertex. Due to the sparsity of point clouds, there tend to be fewer points for some body parts such as hands or feet, so we simplify the original 24 body part labels in the mesh vertices to 9, including head, left arm, right arm, upper body, lower body, left upper leg, left lower leg, right upper leg, and right lower leg. In practical applications, occlusions and noise are inevitable. To address these challenges, we synthesize point clouds in various shapes and attach them to the appropriate positions of human point clouds to simulate common noises, such as carrying objects or using umbrellas. Subsequently, these points are labeled as "noise". Additionally, we randomly crop the human point clouds to mimic occlusions. These operations enable our human body segmentation network to distinguish noise and adapt to occlusions, thus improving its robustness and performance.

Human Motion Flow Synthetic Dataset. Based on the LiDAR point cloud generation model [3, 24], we create a motion flow estimation synthetic dataset. We derive human motion flow by matching irregular and unordered LiDAR point clouds with regular and ordered mesh vertices, thereby establishing a correspondence between synthetic points in consecutive frames. We follow LIP [24] to create 2,378,871 frames of synthetic point clouds from SMPL mesh in AMASS [18] and SURREAL [29], which provide diverse human motions. We match each point to the nearest mesh vertex by utilizing the k-Nearest Neighborhood (kNN) algorithm. Since the vertices of each frame have one-to-one correspondences, we can find the corresponding points in each frame based on the matched vertices. Moreover, we also use bidirectional filtering and set distance thresholds to improve the accuracy of finding corresponding points.

Action Recognition Dataset. **HuCenLife** [33] is a human-centric action recognition dataset. It comprises 65,265 human instances and 12 kinds of human actions. We divide the dataset into 27142 partially overlapping clips containing 30 consecutive frames. 19594 clips are used as the train set while 7548 clips are used as the test set. It adopts the class mean accuracy (mAcc) as the evaluation metric.

3D Pose Estimation Dataset. **LIPD** [24] is a long-range LiDAR-IMU hybrid human mocap dataset with diverse challenge motions. It comprises 15 performers with 30 types of motions, totaling 62,341 LiDAR point cloud frames, each paired with corresponding IMU measurements. Following the LIP [24] protocol, we divide the dataset into 39,593 frames for training and 22,748 frames for testing. We use mean per root-relative joint position error (MPJPE) in millimeters as the evaluation metric.

4.2. Implementation Details

For HuCenLife, we use point clouds with consecutive frames of $L = 30$ frames as the input ($L = 32$ when dealing with LIP). The dimension D of each point is 3. After normalizing and sampling to $N = 384$ points by Farthest Point Sampling (FPS), we apply pre-trained HBSeg and HMFlow on real data to obtain the point-wise segmentation labels S and motion flow vectors F with dimension D' set to 3.

For each instance, we group points with the same part segmentation labels into 9 point patches, denoted as P, with M representing the total number of patches. Subsequently, $N' = 48$ points are sampled using FPS in each point patch P. After being masked with temporal mask ratio $r_t = 0.8$ and spatial mask ratio $r_s = 0.6$, each visible point patch has its part token embedding derived using Mini-PointNet [21], with a channel dimension $C = 384$. In the self-learning stage, the features of F are not used, to prevent the premature leakage of location information of masked tokens to the

Table 1. Action Recognition in HuCenLife [33]. † means adding global token and motion flow to these methods fair comparisons.

	lift	carry	move	pull_push	sco-bal	hum-inter	fitness	entertain	sports	bend-over	sit	walk-stand	mAcc
PointNet [21]	45.5	48.8	33.3	84	59.4	2.6	65.3	49.3	34.8	29.2	54.3	61	47.3
PointNet++ [22]	49.5	45.7	35.6	52.7	59	6	28.6	43.8	41.2	31.9	38.8	55	40.7
PointMLP [17]	48.5	47.7	57.7	80.1	80.3	36.1	75.7	60.8	39.5	54.9	55.8	59.7	58.1
PointNeXt [23]	48.1	56.6	34.1	80	85.6	22.6	50	38	25.7	25.5	63.1	70.9	50
PCT [10]	39.7	54.9	52.3	80.2	89.8	9.8	63.3	73.6	37.7	62.5	51	75.8	57.6
HuCenLife [33]	45	44.4	52.7	81.2	86.7	23.1	81.2	54.8	41.7	54.8	53.2	70	57.4
PointMAE† [19]	53.4	53.1	47.2	84.9	88.8	7.8	71.4	76.8	39.2	57.9	41.8	74.2	58.0
MaST-Pre† [25]	32.8	39.9	48.4	84.5	87.4	31.4	70.7	59.1	43.3	51.7	66.9	32.5	54.1
UniPVU-Human	27.1	37.3	57.1	82.6	84	24.7	85.4	52.1	53.9	93.8	67.3	76.1	**61.8**

Table 2. 3D Pose Estimation in LIP [24].

	MPJPE(mm)↓
LiDARCap(PC) [13]	69.4
LIP(PC) [24]	60.1
UniPVU-Human(PC)	**58.8**
LIP(PC+IMU) [24]	48.9
UniPVU-Human(PC+IMU)	**47.2**

STEncoder. The spatial and temporal positional encoding are obtained by applying MLP to the average coordinate of P and to the time index ranging from 0 to L-1, respectively. For STEncoder, we set the number of heads to 6. It contains 4 spatial, 4 temporal, and 4 spatial transformer layers sequentially. The decoder consists of 4 spatial transformer layers, and ChamferDistanceL2 is used as the loss function for mask reconstruction. AdamW optimizer is used with an initial learning rate of 0.001 and a weight decay of 0.05. The model is trained for 300 epochs with a batch size of 512.

During fine-tuning, P and corresponding motion flow vector F are extracted by two tokenizers respectively and element-wise added in latent space (See details in supplementary materials), with a global token extracted from the entire human and a class token appended. Totally 11 tokens are sent to the pre-trained STEncoder. The pre-trained model is fine-tuned for 100 epochs using a batch size of 256 on 4 GPUs. We use the AdamW optimizer, and the initial learning rate is set to 0.0005 with a cosine decay strategy.

4.3. Results

4.3.1 Action Recognition in HuCenLife

The results of all methods on HuCenLife are shown in Table.1, and the evaluation metric is class mean accuracy (mAcc). For the first seven methods in the table, which are designed for processing static point clouds, we apply them on each frame of the point cloud sequence and then fuse these frame features after the encoder network by element-wise adding. For methods that also use self-learning like PointMAE [19] and MaST-Pre [25], we add a global token and motion flow for fair comparisons. The experimental results demonstrate that our method significantly outperforms the methods that do not utilize self-learning. Even against general static point cloud methods with self-learning, like PointMAE, our approach shows a marked improvement due to our comprehensive temporal dynamic modeling. Compared to methods like MaST-Pre, which also models both temporal and spatial dimensions of point cloud videos along with mask prediction, our method still maintains a substantial advantage. This advantage is particularly evident in some action categories where accurate recognition hinges on extracting not just geometric features but also motion characteristics, including Fitness, Sports (encompassing activities like basketball and badminton), Bend-Over, and Walk-Stand. Experimental data shows that UniPVU-Human demonstrates superior recognition performance in these action categories. This highlights the effectiveness of utilizing human body prior knowledge and underscores the powerful capability of our model to fully leverage this prior knowledge for enhanced performance.

4.3.2 3D Pose Estimation in LIP

The Mean Per Root-Relative Joint Position Error (MPJPE) in millimeters is used as the evaluation metric, where a smaller value indicates better performance. Unlike action recognition, pose estimation in long point cloud videos focuses on long-term joint movement consistency. Our UniPVU-Human leverages a self-learning module for crucial human motion representation and enhances motion detail with motion flow during fine-tuning. To ensure a fair comparison, we establish two settings as shown in Table. 2. 1) **Pure PC:** involves only pure point clouds as input, where our method's MPJPE (58.8) is 1.3 lower than that of LIP (60.1). 2) **PC+IMU:** involves using both point cloud and IMU data as inputs. For UniPVU-Human, we replace the PointNet used for point cloud feature extraction in LIP with our model. The experimental results indicate that our method's MPJPE (47.2) is 1.7 lower than that of LIP (48.9) of full configuration. Our method achieves SOTA performance under both settings, proving its superiority.

4.4. Ablation Studies

All ablation studies are conducted on HuCenLife [33], for it is collected in real-life scenarios, making it more relevant to real-world applications. The results are shown in Table. 3.

Our UniPVU-human exploits both the structural semantics of human body and human motion to facilitate the acquisition of human-specific features by adopting human body parts as local patches for following spatio-temporal modeling with Transformers, unlike other methods which cluster the neighbor point clouds around the kernels sampled by FPS algorithm. As shown in the first and second lines of 3, our setting is identical to PointMAE [19] both with and without self-learning. The mean accuracies (mAcc) in these settings are 53.4% and 56.1%, respectively. By enhancing PointMAE with hierarchical features in the fine-tuning stage, as shown in the third line of the table, the mAcc reaches 58%, yielding a performance gain of 1.9%. This indicates that hierarchical human-related features can also enhance performance in other models. However, this setting still trails our best model by 3.8%, underscoring the effectiveness of our model's approach of using body parts as semantic tokens.

Lines 5 to 7 of the table demonstrate that our model benefits from predicting masked tokens in both spatial and temporal dimensions within the self-learning module. Adding our self-learning module resulted in a substantial 6.2% improvement in total.

During the aforementioned self-learning module, our model has already achieved the capability to model the semantic structure of the human body at the part level. In the Hierarchical Feature Enhanced Fine-tuning module, we incorporate a global token and point-wise motion flow. As indicated in lines 8 to 10, the inclusion of these two designs leads to a notable performance improvement, highlighting the critical role of hierarchical human-related prior knowledge in enhancing the extraction of human geometric and motion representations. In conclusion, by seamlessly integrating various elements of our design, UniPVU-Human achieves exceptional performance with a final accuracy of 61.8%. This demonstrates the effectiveness of our harmonious incorporation of components in facilitating human-centric representation learning.

4.5. Effectiveness of Our Self-learning Mechanism in Semi-supervised Settings

For supervised learning methods, the optimization targets of neural networks mainly come from human annotations. To endow models with strong robustness and generalizability for diverse applications, extensive data annotation is usually required. Therefore, our method introduces a self-learning mechanism, which diminishes the dependency on manual annotations, allowing our model to undergo self-learning on a vast quantity of unannotated data. Additionally, our method learns intrinsic representations directly from the data, uninhibited by the constraints and biases of task-specific, scenario-specific, or dataset-specific manual annotations. As a result, the representations acquired are not task-specific and exhibit strong generalization capabilities, which improves performance on downstream tasks.

To validate this, we randomly sample the training set of HuCenLife by categories and assess our model's performance in fine-tuning with reduced data volumes, thereby verifying the effectiveness of self-learning. The experimental results in Table. 4 illustrate that when downsampling the downstream task dataset HuCenLife to 20%, 30%, and 50%, our method shows the least decline in performance compared to UniPVU-Human without self-learning and MaST-Pre.

Table 3. Ablation Studies of Network Design. To assess the effectiveness of designs in our UniPVU-Human, we perform ablation experiments by adding (✓) or removing (✗) them, and then present the corresponding resulting changes in performance on HuCenLife [33].

part division	Self-learning Mask		Hierarchical Feature		mAcc
	spatial	temporal	global token	motion flow	
✗	✗	✗	✗	✗	53.4
✗	✓	✗	✗	✗	56.1
✗	✓	✗	✓	✓	58
✓	✗	✗	✗	✗	54.1
✓	✗	✗	✓	✓	55.6
✓	✓	✗	✓	✓	59.9
✓	✗	✓	✓	✓	59.2
✓	✓	✓	✗	✗	58.9
✓	✓	✓	✗	✓	59.3
✓	✓	✓	✓	✗	61.3
✓	✓	✓	✓	✓	**61.8**

Table 4. Effectiveness of Our Self-learning Mechanism in Semi-supervised Settings on HuCenLife. * means training on HuCenLife directly without the self-learning stage. The experimental results indicate that our method demonstrates the smallest decline in performance.

	proportion of fine-tuning dataset			
	20%	30%	50%	100%
MaST-Pre [25]	39.8(-14.3)	42(-12.1)	48.8(-5.3)	54.1
UniPVU-Human*	44.9(-10.9)	46.4(-9.4)	49.5(-6.3)	55.8
UniPVU-Human	51(**-10.8**)	53.8(**-8**)	57.3(**-4.5**)	61.8

5. Conclusion

Given the distinctive characteristics inherent to humans, including the structural semantics of the human body and the dynamics of human motions, we introduce a novel method in this paper to delve into the intrinsic features present within the data itself to facilitate a more comprehensive understanding of human-centric point cloud videos. To our knowledge, our method is the first work to provide a unified framework designed specifically for tackling human-centric tasks. Extensive experiments on various tasks have demonstrated the state-of-the-art performance of our method.

References

[1] Guangyan Chen, Meiling Wang, Yi Yang, Kai Yu, Li Yuan, and Yufeng Yue. Pointgpt: Auto-regressively generative pre-training from point clouds. *arXiv preprint arXiv:2305.11487*, 2023. 3

[2] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019. 2

[3] Peishan Cong, Xinge Zhu, and Yuexin Ma. Input-output balanced framework for long-tailed lidar semantic segmentation. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2021. 6

[4] Peishan Cong, Yiteng Xu, Yiming Ren, Juze Zhang, Lan Xu, Jingya Wang, Jingyi Yu, and Yuexin Ma. Weakly supervised 3d multi-person pose estimation for large-scale scenes based on monocular camera and single lidar. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 461–469, 2023. 1

[5] Yudi Dai, YiTai Lin, XiPing Lin, Chenglu Wen, Lan Xu, Hongwei Yi, Siqi Shen, Yuexin Ma, and Cheng Wang. Sloper4d: A scene-aware dataset for global 4d human pose estimation in urban environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 682–692, 2023. 1

[6] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017. 5

[7] Hehe Fan, Yi Yang, and Mohan Kankanhalli. Point 4d transformer networks for spatio-temporal modeling in point cloud videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14204–14213, 2021. 2

[8] Hehe Fan, Yi Yang, and Mohan Kankanhalli. Point spatio-temporal transformer networks for point cloud video modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):2181–2192, 2022. 2

[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] Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7:187–199, 2021. 1, 2, 7

[11] Xiao Han, Peishan Cong, Lan Xu, Jingya Wang, Jingyi Yu, and Yuexin Ma. Licamgait: Gait recognition in the wild by using lidar and camera multi-modal visual sensors. *arXiv preprint arXiv:2211.12371*, 2022. 1

[12] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 3

[13] Jialian Li, Jingyi Zhang, Zhiyong Wang, Siqi Shen, Chenglu Wen, Yuexin Ma, Lan Xu, Jingyi Yu, and Cheng Wang. Lidarcap: Long-range marker-less 3d human motion capture with lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20502–20512, 2022. 1, 2, 7

[14] Zhiqi Li, Nan Xiang, Honghua Chen, Jianjun Zhang, and Xiaosong Yang. Deep learning for scene flow estimation on point clouds: A survey and prospective trends. In *Computer Graphics Forum*. Wiley Online Library, 2023. 4

[15] 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

[16] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 4

[17] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *arXiv preprint arXiv:2202.07123*, 2022. 1, 2, 7

[18] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5442–5451, 2019. 4, 6

[19] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *European conference on computer vision*, pages 604–621. Springer, 2022. 3, 5, 7, 8

[20] Gilles Puy, Alexandre Boulch, and Renaud Marlet. Flot: Scene flow on point clouds guided by optimal transport. In *European conference on computer vision*, pages 527–544. Springer, 2020. 4

[21] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 1, 2, 5, 6, 7

[22] 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. 1, 2, 7

[23] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *Advances in Neural Information Processing Systems*, 35:23192–23204, 2022. 2, 4, 7

[24] Yiming Ren, Chengfeng Zhao, Yannan He, Peishan Cong, Han Liang, Jingyi Yu, Lan Xu, and Yuexin Ma. Lidar-aid inertial poser: Large-scale human motion capture by sparse inertial and lidar sensors. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2337–2347, 2023. 1, 2, 4, 6, 7

[25] Zhiqiang Shen, Xiaoxiao Sheng, Hehe Fan, Longguang Wang, Yulan Guo, Qiong Liu, Hao Wen, and Xi

Zhou. Masked spatio-temporal structure prediction for self-supervised learning on point cloud videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16580–16589, 2023. 2, 3, 7, 8

[26] Zhiqiang Shen, Xiaoxiao Sheng, Longguang Wang, Yulan Guo, Qiong Liu, and Xi Zhou. Pointcmp: Contrastive mask prediction for self-supervised learning on point cloud videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1212–1222, 2023. 1, 3

[27] Xiaoxiao Sheng, Zhiqiang Shen, and Gang Xiao. Contrastive predictive autoencoders for dynamic point cloud self-supervised learning. *arXiv preprint arXiv:2305.12959*, 2023.

[28] Xiaoxiao Sheng, Zhiqiang Shen, Gang Xiao, Longguang Wang, Yulan Guo, and Hehe Fan. Point contrastive prediction with semantic clustering for self-supervised learning on point cloud videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16515–16524, 2023. 1, 3

[29] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 109–117, 2017. 6

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2, 5

[31] Haiyan Wang, Liang Yang, Xuejian Rong, Jinglun Feng, and Yingli Tian. Self-supervised 4d spatio-temporal feature learning via order prediction of sequential point cloud clips. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3762–3771, 2021. 1, 3

[32] 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. 3

[33] Yiteng Xu, Peishan Cong, Yichen Yao, Runnan Chen, Yuenan Hou, Xinge Zhu, Xuming He, Jingyi Yu, and Yuexin Ma. Human-centric scene understanding for 3d large-scale scenarios. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20349–20359, 2023. 1, 2, 4, 6, 7, 8

[34] Ming Yan, Xin Wang, Yudi Dai, Siqi Shen, Chenglu Wen, Lan Xu, Yuexin Ma, and Cheng Wang. Cimi4d: A large multimodal climbing motion dataset under human-scene interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12977–12988, 2023. 1

[35] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *CVPR*, pages 11784–11793, 2021. 2

[36] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 19313–19322, 2022. 3

[37] Changyu Zeng, Wei Wang, Anh Nguyen, and Yutao Yue. Self-supervised learning for point cloud data: A survey. *Expert Systems with Applications*, page 121354, 2023. 4

[38] Renrui Zhang, Ziyu Guo, Peng Gao, Rongyao Fang, Bin Zhao, Dong Wang, Yu Qiao, and Hongsheng Li. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. *Advances in neural information processing systems*, 35:27061–27074, 2022. 3

[39] Zhuoyang Zhang, Yuhao Dong, Yunze Liu, and Li Yi. Complete-to-partial 4d distillation for self-supervised point cloud sequence representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17661–17670, 2023. 1, 3

[40] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021. 2

[41] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *CVPR*, pages 4490–4499. IEEE Computer Society, 2018. 2

[42] Zixiang Zhou, Xiangchen Zhao, Yu Wang, Panqu Wang, and Hassan Foroosh. Centerformer: Center-based transformer for 3d object detection. In *ECCV*, pages 496–513. Springer, 2022.

[43] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Wei Li, Yuexin Ma, Hongsheng Li, Ruigang Yang, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar-based perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6807–6822, 2021. 2