Learnable Query Contrast and Spatio-temporal Prediction on Point Cloud Video Pre-training

Xiaoxiao Sheng , Zhiqiang Shen , Longguang Wang , and Gang Xiao

***Abstract*—Point cloud videos capture the time-varying environment and are widely used for dynamic scene understanding. Existing methods develop effective networks for point cloud videos but do not fully utilize the prior information uncovered during pre-training. Furthermore, relying on a single supervised task with a large amount of manually labeled data may be insufficient to capture the foundational structures in point cloud videos. In this paper, we propose a pre-training framework Query-CP to learn the representations of point cloud videos through multiple self-supervised pretext tasks. First, token-level contrast is developed to predict future features under the guidance of historical information. Using a position-guided autoregressor with learnable queries, the predictions are directly contrasted with corresponding targets in the high-level feature space to capture fine-grained semantics. Second, performing only contrastive learning fails to fully explore the complementary structures and dynamics information. To alleviate this, a decoupled spatio-temporal prediction task is designed, where we use a spatial branch to predict low-level features and a temporal branch to predict timestamps of the target sequence explicitly. By combining the above self-supervised tasks, multi-level information is captured during the pre-training stage. Finally, the encoder is fine-tuned and evaluated for action recognition and dynamic semantic segmentation on three datasets. The results demonstrate the effectiveness of our Query-CP. Especially, compared with the state-of-the-art methods, the fine-tuning accuracy on action recognition improves by 3.23% for 24-frame point cloud videos, and the mean accuracy increases by 4.21%. Our code will be available at https://github.com/JohnsonSign/Query-CDP.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/9033

***Index Terms*—3D deep learning, point clouds, self-supervised pre-training, contrastive learning.**

I. Introduction

Point cloud video is a sequence composed of multiple frames of static point clouds. The understanding of point cloud videos is a novel task in recent years for human-object interaction and autonomous driving. Although recent supervised works have achieved great performance, they heavily rely on limited tasks based on manual labeling. To remedy this, it is essential to introduce self-supervised pre-training to learn spatio-temporal features from raw point cloud videos.

The associate editor coordinating the review of this manuscript and approving it for publication was Luis Camarinha-Matos (Corresponding author: Xiaoxiao Sheng).

X. Sheng, Z. Shen, and G. Xiao are with Shanghai Jiao Tong University, Shanghai, China (e-mails: shengxiaoxiao@sjtu.edu.cn, shenzhiqiang@sjtu.edu.cn, and xiaogang@sjtu.edu.cn).

L. Wang is with the Aviation University of Air Force, Changchun, China (e-mail: wanglongguang15@nudt.edu.cn).

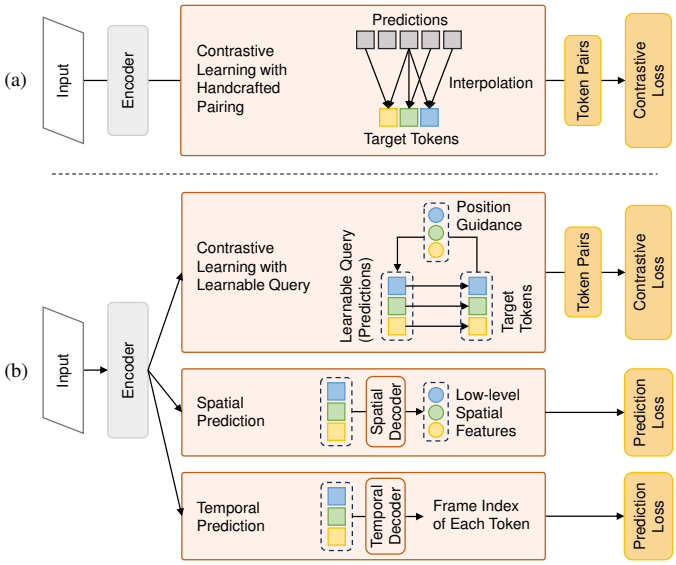
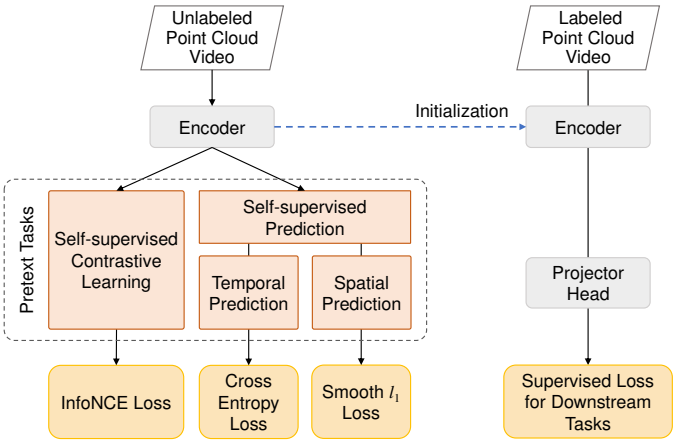


Fig. 1. A comparison between (a) existing contrastive learning paradigm and (b) our Query-CP pre-training. We propose query-based contrastive learning with a learnable pairing method. Moreover, we introduce spatio-temporal prediction tasks. These complementary self-supervised pretext tasks promote the capturing of multi-level features during pre-training.

The development of static point cloud modeling has achieved significant success [1], [2]. Motivated by this, efforts have been made to model point cloud videos employing spatio-temporal convolutions [3], [4] or Transformer networks [5]–[7] in a supervised manner. Wang *et al*. [8] explicitly transformed the points into voxel sequences for motion modeling. Liu *et al*. [9] constructed a spatio-temporal encoder based on PointNet++ [10] to process raw point cloud videos. PSTNet [3] employs multiple decoupled spatio-temporal convolutions to extract effective features. P4Transformer [5] constructs an attention-based network to capture spatio-temporal relations between global tokens. PST-Transformer [7] introduces an advanced position-based self-attention mechanism.

Recently, self-supervised pre-training has been studied to enhance the performance of encoders [11]–[16], with contrastive learning emerging as an effective mainstream method on point cloud video modeling. Dong *et al*. [12] proposed frame-level semantics distillation with contrastive loss by feeding complete-to-partial sequences into teacher-student networks. CPR [15] develops token-level contrastive learning to capture spatio-temporal features, and PointCPSC [13] further designs a sampling strategy to improve the token-level contrast. However, as shown in Fig. 1(a), they rely on hand-crafted



(a) Self-supervised Pre-training (b) Fine-tuning

Fig. 2. **The pipeline of our Query-CP.** Three collaborative pretext tasks are designed for self-supervised pre-training of point cloud videos. Utilizing contrastive learning and decoupled spatio-temporal prediction, the encoder learns multi-level information. After pre-training, the encoder is fine-tuned for downstream tasks. Pre-training enhances the encoder without adding any complexity during deployment.

sample matching and utilize nearest-neighbor interpolation to align the predictions and target tokens for contrastive pairing. The interpolation-based method can introduce misalignment when constructing sample pairs, failing to capture fine-grained features accurately. Moreover, they neglect the essential low-level structures and temporal dynamics necessary for downstream tasks.

To alleviate the above issues, we develop a self-supervised pre-training framework for point cloud videos, termed Query-CP. Our Query-CP predicts multi-grained spatio-temporal representations of the later sequence using early information. As shown in Fig.2, three collaborative pretext tasks are designed for pre-training. Following the setup of previous works, the encoder is fine-tuned for downstream tasks after pre-training. Specifically, inspired by related work in images [17]–[19], the learnable queries are updated through an autoregressor under the positional guidance of target tokens. Using positional clues, the accurate positive pairs are directly formed between the updated queries and targets, alleviating hand-crafted matching and interpolation-based alignment. Then, the decoupled spatial and temporal branches are established to promote the learning of geometry and dynamics, respectively. For the spatial branch, the prediction of low-level features aggregated from local regions enables the encoder to capture fine-grained structures. Meanwhile, the temporal branch explicitly predicts the timestamps of tokens in future sequences to force the encoder to learn motion information. Finally, we conduct evaluations on action recognition and dynamic semantic segmentation for point cloud videos. Compared with supervised counterparts and current pre-training works, our method achieves state-of-the-art results on the MSRAction 3D [20], NTU-RGBD [21], and Synthia 4D [22] datasets. Extensive ablation studies are also conducted to demonstrate the effectiveness of the proposed Query-CP. The main contributions of this paper are as follows:

- We propose a self-supervised pre-training framework with three collaborative pretext tasks for point cloud videos to learn effective spatio-temporal representations.
- The high-level features are predicted using a position-guided autoregressor with learnable queries, and self-supervised contrastive learning is conducted between the predictions and targets.
- Decoupled spatial low-level feature prediction and temporal index prediction branches are designed to ensure that the learned representations are rich in geometries and dynamics simultaneously.
- Extensive experiments and ablation studies are conducted to demonstrate the effectiveness of the proposed method.

The contents of the rest are organized as follows. The proposed method Query-CP is presented in Section II. The experimental results and ablation studies are presented in Section III. The conclusion and future works are illustrated in Section IV.

II. Method

In this section, we first briefly present the spatio-temporal encoding. Then, we introduce query-based contrastive learning and decoupled spatio-temporal prediction. The framework of Query-CP is presented in Fig. 3.

A. Spatio-Temporal Encoding

Each point cloud video is divided into the early sequence $\boldsymbol{S_e} \in \mathbb{R}^{M\times N\times 3}$ and the later sequence $\boldsymbol{S_l} \in \mathbb{R}^{L\times N\times 3}$, where M and L are the length of each sequence, and N is the number of points in each frame sampled by the farthest point sampling. Following the setup of previous works [12], [14], we focus on designing self-supervised pretext tasks to empower the existing encoder with strong representation capabilities through pre-training. Without loss of generality, the pioneering work PSTNet [3] is chosen as the encoder to verify the effectiveness of our pre-training framework.

Following [3], centered at n-th point in the t-th frame, a spatial neighborhood with radius r is constructed. Then, the center point is duplicated into neighboring frames to establish corresponding spatial neighborhoods. These neighborhoods within a short temporal window d form a spatio-temporal region, denoted as $\mathbf{\Phi} = \{\phi^{t+i}_{p_{(t,n)}} | i \in [\lfloor -d/2 \rfloor, \lfloor d/2 \rfloor]\}$. PSTNet [3] consists of four stages. In each stage, spatial convolution is firstly performed within the i-th neighborhood as follows:

$$\tilde{F}^i_{p_{(t,n)}} = \sum_{p_j \in \phi^{t+i}_{p_{(t,n)}}} f_s(p_j - p_{(t,n)}) \cdot F_{p_j}, \tag{1}$$

where $f_s(\cdot)$ is an Multilayer Perceptron (MLP) based module and F_{p_j} represents the feature of point p_j. MLP contains the linear layer, the BatchNorm operation, and the ReLu activation function. Then, the 1D temporal convolution $f_t(\cdot)$ is performed, and the spatio-temporal feature of point $p_{(t,n)}$ is calculated as follows:

$$F_{p_{(t,n)}} = \sum_{i=\lfloor -d/2 \rfloor}^{\lfloor d/2 \rfloor} f_t(\tilde{F}^i_{p_{(t,n)}}). \tag{2}$$

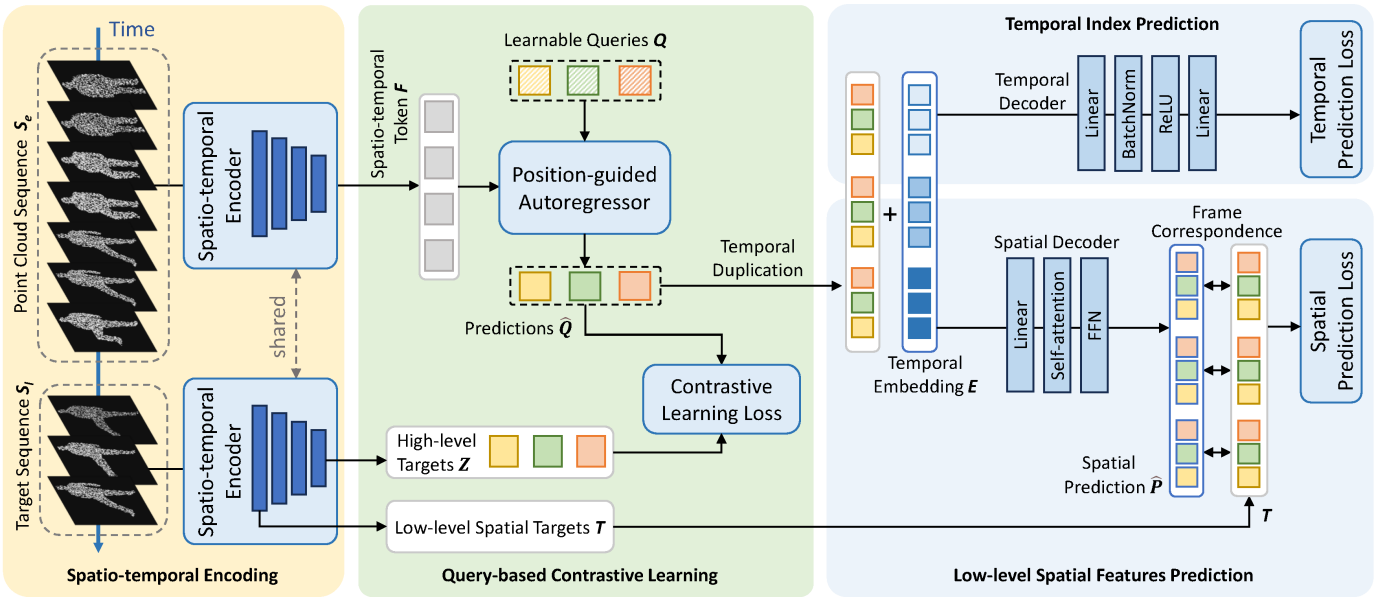


Fig. 3. **Framework of Query-CP.** A point cloud video is divided into an early sequence $\boldsymbol{S_e}$ and a later target sequence $\boldsymbol{S_l}$. The spatio-temporal tokens $\boldsymbol{F}$ obtained through the encoder from $\boldsymbol{S_e}$ are used to predict the multi-level features of $\boldsymbol{S_l}$. The pre-training pretext tasks include query-based contrastive learning, low-level spatial feature prediction, and temporal index prediction. After pre-training, only the spatio-temporal encoder is kept and evaluated for multiple downstream tasks.

Using stacked spatial and temporal convolutions, the encoder embeds these spatio-temporal regions into tokens. The early sequence $\boldsymbol{S_e}$ is embedded into $\boldsymbol{F} \in \mathbb{R}^{M' \times N' \times C}$, where $M' = M/4$ is the length of aggregated sequence, $N' = N/16$ is the aggregated token number, and C is feature dimension. The tokens encoded from the later sequence $\boldsymbol{S_l}$ by the shared spatio-temporal encoder are regarded as high-level target features, denoted as $\boldsymbol{Z} \in \mathbb{R}^{N' \times C}$. It should be noted that, in addition to the extracted features mentioned above, the position of each token is defined by the xyz-coordinates and temporal index of its center point.

B. Query-based Contrastive Learning

This pre-training pretext task performs token-wise contrastive learning between predictions modeled from the early sequence $\boldsymbol{S_e}$ and target features $\boldsymbol{Z}$. In previous work CPR [15], the last few tokens in $\boldsymbol{F}$ are directly considered as predictions, paired with targets through a hand-crafted interpolation. Despite being temporally adjacent, it still introduces misalignment in positive pair construction. To alleviate this, we propose query-based contrastive learning. A group of learnable queries $\boldsymbol{Q} \in \mathbb{R}^{N' \times C}$ are introduced after spatio-temporal encoding. The positional embeddings of $\boldsymbol{Q}$ are learned using a linear layer, where the xyz-coordinates and temporal index t of $\boldsymbol{Z}$ are taken as inputs after concatenation $[.]$:

$$Pos_Q = Linear([xyz, t]_Z). \tag{3}$$

For each token in $\boldsymbol{F}$, its xyz-coordinates and temporal index t are also concatenated as inputs to model the positional embedding Pos_F by a linear layer as follows:

$$Pos_F = Linear([xyz, t]_F), \tag{4}$$

where two linear layers in Eq. 3 and Eq. 4 are different to avoid information leakage. Pos_Q and Pos_F are added to $\boldsymbol{Q}$ and $\boldsymbol{F}$, respectively. Then, a three-layer position-guided autoregressor is used to model token relations and predict the later sequence. In each layer, self-attention is firstly applied to $\boldsymbol{F}$, and cross-attention is employed to facilitate interaction between $\boldsymbol{F}$ and $\boldsymbol{Q}$ with positional cues:

$$\hat{\boldsymbol{F}} = Self\text{-}attention(\boldsymbol{F} + Pos_F), \tag{5}$$

$$[\hat{\boldsymbol{Q}}, \hat{\boldsymbol{F}}] = Cross\text{-}attention([\boldsymbol{Q} + Pos_Q,\ \hat{\boldsymbol{F}}]). \tag{6}$$

Specifically, as shown in Fig. 4, $\boldsymbol{Q}$ is projected by the linear layer W_Q, and $\hat{\boldsymbol{F}}$ is projected by the linear layers W_V and W_K. Through the attention mechanism and the feedforward network (FFN) [23], the updated queries are directly taken as predictions, indicated as $\hat{\boldsymbol{Q}} \in \mathbb{R}^{N' \times C}$. These predictions can be aligned directly with targets using positional cues, thus avoiding manual interpolation alignment. Then, the InfoNCE loss [24] is utilized for contrastive learning as follows:

$$\mathcal{L}_c = -\frac{1}{N'} \sum_{\boldsymbol{z} \in \boldsymbol{Z}} \log \frac{\exp(\boldsymbol{z}^T \hat{\boldsymbol{q}}_+ / \tau)}{\exp(\boldsymbol{z}^T \hat{\boldsymbol{q}}_+ / \tau) + \sum_{\boldsymbol{q}_j \in \Psi} \exp(\boldsymbol{z}^T \boldsymbol{q}_j / \tau)}, \tag{7}$$

where $\hat{\boldsymbol{q}}_+ \in \hat{\boldsymbol{Q}}$, $(\boldsymbol{z}, \hat{\boldsymbol{q}}_+)$ is a positive pair constructed using positional clues, Ψ is the negative set containing all mismatched tokens, and τ is a temperature hyperparameter.

C. Decoupled Spatio-temporal Prediction

To learn multi-grained representations, two other pretext tasks are designed. The spatial branch captures the structures and geometry. Meanwhile, the temporal branch captures dynamics, which is beneficial for downstream tasks requiring motion modeling.

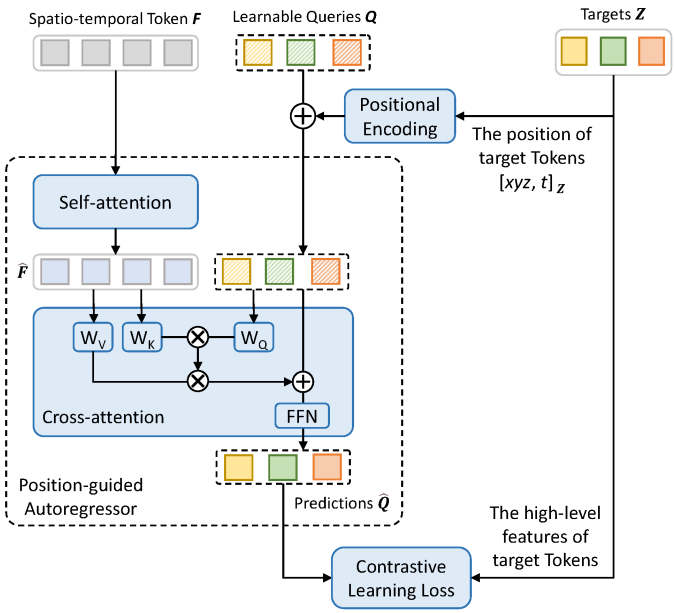


Fig. 4. **The position-guided autoregressor.** Self-attention is first performed on $\boldsymbol{F}$. Then, a cross-attention layer is conducted between $\hat{\boldsymbol{F}}$ and the queries $\boldsymbol{Q}$. By utilizing positional cues from $\boldsymbol{Z}$, predictions can be directly matched with corresponding targets, avoiding manual interpolation alignment.

1) Spatial Features Prediction: Low-level spatial features aggregated from local regions are regarded as prediction targets. Specifically, we utilize the predictions $\hat{\boldsymbol{Q}} \in \mathbb{R}^{N' \times C}$ to predict the low-level features of the target sequence $\boldsymbol{T} \in \mathbb{R}^{L \times N' \times C'}$, where C' is feature dimension. $\boldsymbol{T}$ is extracted from the first stage of the encoder by aggregating regional features centered at the positions of target tokens $\boldsymbol{Z}$. The predictions are temporally duplicated by L times, denoted as $\hat{\boldsymbol{Q}}' \in \mathbb{R}^{L \times N' \times C}$. The learnable temporal positional embedding $\boldsymbol{E} \in \mathbb{R}^{L \times C}$ is added to $\hat{\boldsymbol{Q}}'$ to predict low-level spatial features in each frame by a lightweight transformer decoder as follows:

$$\hat{\boldsymbol{P}} = Decoder_s(Linear(\hat{\boldsymbol{Q}}' + \boldsymbol{E})), \tag{8}$$

where a linear layer is performed to transform feature dimension, $Decoder_s$ contains one layer of self-attention with feed-forward network [5], and $\hat{\boldsymbol{P}} \in \mathbb{R}^{L \times N' \times C'}$ is the predictions of low-level features. The smooth l_1 loss is introduced for spatial prediction. $\mathcal{L}^l_{i,c}$ is the loss of c-th dimension of the i-th token in the l-th frame:

$$\mathcal{L}^l_{i,c} = \begin{cases} 0.5 \times (\boldsymbol{T}^l_{i,c} - \hat{\boldsymbol{P}}^l_{i,c})^2, & \text{if } |\boldsymbol{T}^l_{i,c} - \hat{\boldsymbol{P}}^l_{i,c}| < 1 \\ |\boldsymbol{T}^l_{i,c} - \hat{\boldsymbol{P}}^l_{i,c}| - 0.5, & \text{otherwise} \end{cases}, \tag{9}$$

$$\mathcal{L}_s = \frac{1}{L \times N'} \sum_{l=1}^{L} \sum_{i=1}^{N'} \frac{1}{C'} \sum_{c=1}^{C'} \mathcal{L}^l_{i,c}, \tag{10}$$

where $\mathcal{L}_s$ is the spatial feature prediction loss.

2) Temporal Index Prediction: The motion information is learned through temporal index prediction. Specifically, $\hat{\boldsymbol{Q}}'$ added with the temporal embedding $\boldsymbol{E}$ is input into a temporal decoder as follows:

$$\boldsymbol{I_{pt}} = Decoder_t(\hat{\boldsymbol{Q}}' + \boldsymbol{E}), \tag{11}$$

where $\boldsymbol{I_{pt}}$ is the prediction of timestamps, and $Decoder_t$ is an MLP head. The cross entropy loss CE is used to optimize this pretext task as follows:

$$\mathcal{L}_t = CE(\boldsymbol{I_{pt}}, \boldsymbol{I_{gt}}), \tag{12}$$

where the timestamps of target tokens in future sequence $\boldsymbol{S_l}$ are ground truth, indicated as $\boldsymbol{I_{gt}}$, and $\mathcal{L}_t$ is the temporal index prediction loss. In this way, the temporal features are explicitly captured. Furthermore, the learning of temporal positional encoding $\boldsymbol{E}$ is facilitated. Since $\boldsymbol{E}$ is shared between two branches, it also promotes spatial prediction. Finally, the total loss is calculated as follows:

$$\mathcal{L}_{total} = \mathcal{L}_c + \lambda_1 \mathcal{L}_s + \lambda_2 \mathcal{L}_t, \tag{13}$$

where λ_1 and λ_2 represents the loss balance coefficient. Overall, the self-supervised pretext tasks of query-based contrastive learning and decoupled spatio-temporal prediction make the representations rich in multi-grained information.

III. Experiments

In this section, we first introduce the experimental details. Then, the performance comparisons are conducted on two downstream tasks of point cloud video modeling. Next, extensive ablation studies show the effectiveness of our Query-CP.

A. Datasets

MSRAction 3D [20] dataset consists of 567 videos for action recognition, including 20 categories collected by 10 subjects in real-world scenarios. We utilize the same training and test splits as previous works [3], [9]. The action categories in the MSRAction 3D dataset contain high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, and so on.

NTU-RGBD [21] dataset contains 56,880 real-world videos, recorded by multi-view cameras. Following cross-subject settings in [21], this dataset is split into 40,320 training videos and 16,560 test videos. There are 60 action categories in the NTU-RGBD dataset, including 40 daily actions (*e.g.*, reading, drinking, eating), 11 interaction actions (*e.g.*, hugging), and 9 health-related actions (*e.g.*, falling).

Synthia 4D [22] dataset is used for semantic segmentation on autonomous driving. It includes 6 long-term videos, containing 12 fine-grained semantic categories. Following [3], [9], 19,888 frames are allocated for training data, 815 frames for validation data, and 1,886 frames for test data.

B. Implementation Details

The proposed method is implemented on a Linux system using 8 NVIDIA 2080Ti GPUs, each with 11GB of RAM. The versions of PyTorch and Python are 1.7.1 and 3.8, respectively. For action recognition, PSTNet [3] is adopted as the encoder. To fairly compare with the recent advanced methods, we follow PSTNet [3] and P4Transformer [5] to use the same dataset splits. We also use the optimal frame selection as presented in these classic methods. The frame intervals on the MSRAction 3D and the NTU-RGBD dataset

TABLE I
ACTION RECOGNITION ACCURACY (%) ON MSRACTION 3D. THE VALUES WITHIN THE PARENTHESES INDICATE THE IMPROVEMENT BROUGHT BY OUR SELF-SUPERVISED PRE-TRAINING COMPARED TO THE BASELINE

Methods	**#Input Frames**				**Mean**
	8	12	16	24	
MeteorNet [9]	81.14	86.53	88.21	88.50	86.10
Kinet [25]	83.84	88.53	91.92	93.27	89.39
PST^2 [6]	86.53	88.55	89.22	-	-
PPTr [26]	84.02	89.89	90.31	92.33	89.14
P4Transformer [5]	83.17	87.54	89.56	90.94	87.80
PST-Transformer [7]	83.97	88.15	91.98	93.73	89.46
PSTNet++ [4]	83.50	88.15	90.24	92.68	88.64
PSTNet [3] (*Baseline**)	83.50	87.88	89.90	91.20	88.12
PSTNet + PointCPSC [13]	88.89	90.24	92.26	92.68	91.02
PSTNet + Query-CP (Ours)	**89.90** (6.40↑)	**92.04** (4.16↑)	**92.93** (3.03↑)	**94.43** (3.23↑)	**92.33** (4.21↑)

are set to 1 and 2, respectively. For each point cloud video, we densely sample 24 frames as input for pre-training. Each frame contains 1024 points sampled by the farthest point sampling. The lengths of early sequence $\boldsymbol{S_e}$ and target sequence $\boldsymbol{S_l}$ are set to 20 and 4, respectively. Random scaling augmentation, a Stochastic Gradient Descent (SGD) optimizer, and a cosine decay scheduler are used during pre-training. We pre-train our model for 200 epochs. The batchsize and the initial learning rate are set to 80 and 0.01, respectively. The temperature hyper-parameter is set to 0.01.

For comprehensive evaluations, P4Transformer [5] is also adopted as the encoder in dynamic semantic segmentation. In this task, each sample contains 4 frames, and each frame contains 4096 points during pre-training. The frame interval on the Synthia 4D dataset is set to 1. The length of $\boldsymbol{S_e}$ and $\boldsymbol{S_l}$ is 3 and 1, respectively. An AdamW optimizer and a cosine decay scheduler are used for optimization. We pre-train for 75 epochs with an initial learning rate of 0.0008, and the batchsize is set to 16. In addition, λ_1 and λ_2 are set to 0.5. After pre-training, the encoder is kept for fine-tuning in the following experiments.

To avoid overfitting, we use dense sampling, data augmentations, and early stopping strategies. Under our experimental setups, the training times on different datasets are as follows. On the MSRAction 3D dataset, the unsupervised pre-training takes about 0.5 days. Subsequently, fine-tuning with the pre-trained encoder takes approximately 3 hours. On the NTU-RGBD dataset, unsupervised pre-training required about 5 days. The fine-tuning experiments on the entire dataset take about 4 days. On the Synthia 4D dataset, the unsupervised pre-training and fine-tuning times are about 2 days and 1.5 days, respectively.

C. Evaluation Metrics

Two evaluation metrics are used to assess the effectiveness of the proposed method, including accuracy and mean Intersection over Union (mIoU). In action recognition, accuracy represents the proportion of correct predictions out of all results, and is calculated as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}, \tag{14}$$

TABLE II
ACTION RECOGNITION ACCURACY (%) ON NTU-RGBD DATASET. THE VALUES WITHIN THE PARENTHESES INDICATE THE IMPROVEMENT BROUGHT BY OUR PRE-TRAINING COMPARED TO THE BASELINE

Methods	Input	NTU-RGBD
PointNet++ [10]	point	80.1
3DV [8]	voxel	84.5
3DV-PointNet++ [8]	voxel + point	88.8
P4Transformer [5]	point	90.2
PST-Transformer [7]	point	91.0
PSTNet [3] (*Baseline**)	point	90.5
PSTNet + CPR [15]	point	91.0
PSTNet + Query-CP (Ours)	point	**91.8** (1.3↑)

where TP represents the number of positive samples correctly identified, TN represents the number of negative samples correctly predicted, FP represents the number of samples incorrectly predicted as positives, and FN represents the number of samples incorrectly predicted as negatives.

In dynamic semantic segmentation, IoU represents the ratio of the intersection to the union of the predicted results and the ground truth for a specific class, and it is calculated as follows:

$$\text{IoU} = \frac{TP}{TP + FP + TN}. \tag{15}$$

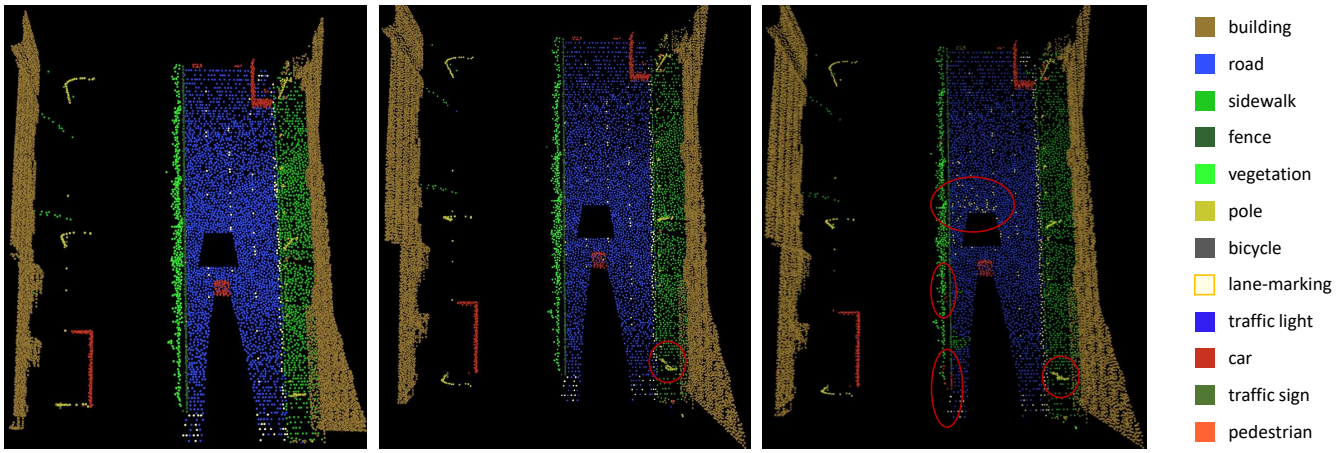
mIoU represents the average of the IoU values for multiple classes:

$$\text{mIoU} = \frac{1}{Class} \sum_{i=1}^{Class} \text{IoU}_i, \tag{16}$$

where $Class$ represents the number of semantic categories.

D. Results and Analysis

1) MSRAction 3D: As shown in Table I, under multiple input lengths, we compare the performance of our Query-CP with the supervised methods and the recent pre-training method that uses the same encoder. Firstly, compared to the baseline PSTNet [3], our Query-CP introduces accuracy improvements of 6.4%, 4.16%, 3.03%, and 3.23% when using 8, 12, 16, and 24 frames as inputs. Secondly, Query-CP also outperforms all other supervised methods, particularly PST-Transformer [7], which utilizes an advanced attention-based



(a) Ground Truth (b) Ours (c) Baseline

Fig. 5. **Semantic segmentation visualization on the Synthia 4D dataset.** Compared to the baseline P4Transformer [5], our proposed Query-CP achieves more accurate segmentation, especially on sidewalks and lane-markings.

TABLE III
DYNAMIC SEMANTIC SEGMENTATION ACCURACY (%) ON SYNTHIA 4D DATASET. THE VALUES WITHIN THE PARENTHESES INDICATE THE IMPROVEMENT BROUGHT BY OUR SELF-SUPERVISED PRE-TRAINING COMPARED TO THE BASELINE

Methods	Frame	Bldn	Road	Sdwlk	Fence	Vegittn	Pole	Car	T. Sign	Pedstrn	Bicycl	Lane	T. Light	mIoU
Minkowski [22]	3	90.13	98.26	73.47	87.19	**99.10**	97.50	94.01	79.04	**92.62**	0.00	50.01	68.14	77.46
MeteorNet [9]	3	**98.10**	97.72	88.65	94.00	97.98	97.65	93.83	84.07	80.90	0.00	71.14	77.60	81.80
PSTNet [3]	3	96.91	98.33	90.83	95.00	96.96	97.61	95.15	77.45	85.68	0.00	75.71	77.28	82.24
P4Transformer [5] (*Baseline**)	3	96.73	98.35	94.03	95.23	98.28	98.01	**95.60**	81.54	85.18	0.00	75.95	79.07	83.16
P4Transformer + Query-CP (Ours)	3	97.31	**98.58**	**94.84**	**95.70**	98.99	**98.34**	95.18	**87.18**	88.73	0.00	**78.49**	**85.38**	**84.81**(1.65↑)

network. Thirdly, compared with pre-training based methods, Query-CP also achieves state-of-the-art results. Especially, our Query-CP achieves a 1.75% improvement in accuracy compared to PointCPSC [13] when using a 24-frame point cloud video as input. The results demonstrate the effectiveness of our self-supervised pre-training framework, significantly enhancing the performance in action recognition.

2) NTU-RGBD: In Table II, we evaluate the performance of our Query-CP against the supervised methods and the pre-training method CPR [15]. By pre-training with our three collaborative pretext tasks, the proposed Query-CP improves the accuracy of the baseline PSTNet [3] by 1.3%. Compared with the supervised method P4Transformer [5] which utilizes self-attention for global modeling, the accuracy of the baseline encoder increases by 1.6% through our pre-training. The results demonstrate that the prior information obtained by self-supervised pre-training is beneficial for action recognition on point cloud videos.

3) Synthia 4D: In Table III, we compare our Query-CP with the state-of-the-art methods for dynamic semantic segmentation. For fair comparisons, we use 3 frames during fine-tuning following other methods. After pre-training, our method enhances the accuracy of the baseline P4Transformer [5] by 1.65%. Query-CP achieves the highest mIoU and outperforms other methods in seven categories, such as roads, sidewalks, and lanes. Specifically, compared with the baseline, our Query-CP improves the IoU on challenging objects such as traffic signs, lane markings, and traffic lights by 5.64%, 2.54%, and 6.31%. This indicates that our pre-training significantly improves the accuracy of small targets through fine-grained pretext tasks. Overall, the knowledge learned through pre-training benefits semantic segmentation.

TABLE IV
STUDIES ON TOKEN-LEVEL CONTRASTIVE LEARNING UNDER 16 FRAMES ON MSRACTION 3D

Pretext Task	**Accuracy (%)**
Interpolation-based Contrastive Learning	91.92
Query-based Contrastive Learning (Ours)	**92.93**

TABLE V
ARCHITECTURE DESIGN UNDER 16 FRAMES ON MSRACTION 3D

Pre-training Framework	**Accuracy (%)**
Query-based contrast	91.58
Query-based contrast + Spatial branch	91.93
Query-based contrast + Spatial branch + Temporal branch	**92.93**

As shown in Fig. 5, we visualize the semantic segmentation results on the Synthia 4D dataset. We compare the results between the baseline and our proposed Query-CP. The red circles indicate the locations with incorrect predictions. In Fig. 5(c), the baseline predicts the land-marking as a cluster of scattered points. Due to the lack of geometric information, segmenting the land-marking category is particularly challenging. Moreover, some points of sidewalks are predicted as cars. Overall, our proposed Query-CP exhibits fewer incorrect predictions and achieves better segmentation performance.

TABLE VI
SPATIAL PREDICTION TARGETS UNDER 16 FRAMES ON MSRACTION 3D

Target	Accuracy (%)
Stage1 (Ours)	**92.93**
Stage2	91.25
Stage3	90.57

E. Ablation Experiments

1) Token-level Contrastive Learning: We compare the pretext tasks of interpolation-based and query-based contrastive learning in Table IV. Except for the contrastive task, the rest of the networks use those in our Query-CP. The interpolation-based contrast manually constructs sample pairs, while our query-based contrast adaptively aligns predictions and targets using a position-guided autoregressor. We can see that query-based contrastive learning surpasses interpolation-based one by 1.01%. This is because our learnable mechanism makes the contrast more accurate, effectively capturing high-level features by pre-training.

2) Architecture Design: The ablation studies on architecture design are presented in Table V. Without pre-training, the baseline PSTNet [3] achieves an accuracy of 89.90% with 16 frames. While only performing pre-training using query-based contrast, the fine-tuning accuracy is 91.58%. The two branches of the decoupled prediction introduce respective performance gains. Notably, the temporal branch introduces a 1% improvement in fine-tuning accuracy by explicitly learning temporal information. Meanwhile, it also enhances the modeling of temporal embeddings, and the shared temporal embeddings further benefit the spatial branch. The above results indicate that all pretext tasks contribute to performance improvements.

3) Prediction Targets of Spatial Branch: Table VI presents the accuracy achieved using different spatial prediction targets. The stages in this experiment are derived from the baseline PSTNet [3], which comprises four stages. Stage 1 only conducts spatial convolution, while Stage 2 and Stage 3 perform both spatial and temporal convolutions. When features from Stage 1 are used as the target, the accuracy is highest, providing complementary information to the query-based contrastive learning. The features from Stage 2 and Stage 3 embed spatio-temporal semantics but lack low-level structures and geometric information, resulting in inferior accuracy.

4) The Execution Time of One Sample on the MSRAction 3D Dataset: We evaluate the execution time of one sample on one Nvidia 2080Ti GPU. For the experiments on the MSRAction 3D dataset, we use PSTNet [3] as the encoder, and the execution time is 0.61s. Besides, the execution time for the P4Transformer [5] and PST-Transformer [7] is 0.58s and 0.76s, respectively. Our method achieves higher accuracy with a similar execution time.

F. Limitation

In our experiments, due to GPU memory limitations, we did not explore the impact of pre-training with larger batch size on performance. To address this issue, it is necessary to further investigate more lightweight and efficient architectures for pre-training. In addition, our proposed method solely utilizes point cloud videos during pre-training. The image sequences can capture the appearance and texture details, which are complementary to point clouds. For better perception of complex scenes, multi-modal self-supervised pre-training based on images and point clouds can be explored.

IV. CONCLUSION

In this paper, we propose a self-supervised pre-training framework that integrates query-based contrastive learning and decoupled prediction for point cloud video understanding. The query-based method constructs positive pairs accurately by combining learnable queries and positional embeddings. This facilitates the pretext task based on contrastive learning. In addition, the decoupled spatial and temporal prediction further promote the learning of geometry and dynamics, respectively. Extensive experiments and ablation studies conducted on two downstream tasks show the effectiveness of our method.

In the future, the establishment of more diverse point cloud video datasets can facilitate exploration into the generalization of existing methods and their adaptability to various application scenarios, such as for embodied intelligence. Furthermore, we will explore more effective pretext tasks to enhance sequential modeling for point cloud videos.

REFERENCES

[1] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, and A. Markham, "RandLA-Net: Efficient semantic segmentation of large-scale point clouds," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 108–11 117, doi:10.1109/CVPR42600.2020.01112.

[2] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3D point clouds: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 12, pp. 4338–4364, 2020, doi:10.1109/TPAMI.2020.3005434.

[3] H. Fan, X. Yu, Y. Ding, Y. Yang, and M. Kankanhalli, "PSTNet: Point spatio-temporal convolution on point cloud sequences," *arXiv preprint arXiv:2205.13713*, 2022, doi:10.48550/arXiv.2205.13713.

[4] H. Fan, X. Yu, Y. Yang, and M. Kankanhalli, "Deep hierarchical representation of point cloud videos via spatio-temporal decomposition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 9918–9930, 2021, doi:10.1109/TPAMI.2021.3135117.

[5] H. Fan, Y. Yang, and M. 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*, 2021, pp. 14 204–14 213, doi:10.1109/CVPR46437.2021.01398.

[6] Y. Wei, H. Liu, T. Xie, Q. Ke, and Y. Guo, "Spatial-temporal transformer for 3D point cloud sequences," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 1171–1180, doi:10.1109/WACV51458.2022.00073.

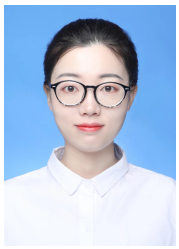
[7] H. Fan, Y. Yang, and M. Kankanhalli, "Point spatio-temporal transformer networks for point cloud video modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 2181–2192, 2022, doi:10.1109/TPAMI.2022.3161735.

[8] Y. Wang, Y. Xiao, F. Xiong, W. Jiang, Z. Cao, J. T. Zhou, and J. Yuan, "3DV: 3D dynamic voxel for action recognition in depth video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 511–520, doi:10.1109/CVPR42600.2020.00059.

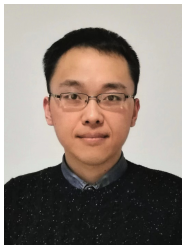
[9] X. Liu, M. Yan, and J. Bohg, "MeteorNet: Deep learning on dynamic 3D point cloud sequences," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9245–9254, doi:10.1109/ICCV.2019.00934.

[10] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in Neural Information Processing Systems*, pp. 5099–5108, 2017, doi:10.48550/arXiv.1706.02413.

[11] H. Wang, L. Yang, X. Rong, J. Feng, and Y. Tian, "Self-supervised 4D spatio-temporal feature learning via order prediction of sequential point cloud clips," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2021, pp. 3762–3771, doi:10.1109/WACV48630.2021.00381.
[12] Y. Dong, Z. Zhang, Y. Liu, and L. 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*, 2023, pp. 17 661–17 670, doi:10.1109/CVPR52729.2023.01694.
[13] X. Sheng, Z. Shen, G. Xiao, L. Wang, Y. Guo, and H. 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*, 2023, pp. 16 469–16 478, doi:10.1109/ICCV51070.2023.01514.
[14] Z. Shen, X. Sheng, H. Fan, L. Wang, Y. Guo, Q. Liu, H. Wen, and X. 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*, 2023, pp. 16 580–16 589, doi:10.1109/ICCV51070.2023.01520.
[15] X. Sheng, Z. Shen, and G. Xiao, "Contrastive predictive autoencoders for dynamic point cloud self-supervised learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023, pp. 9802–9810, doi:10.1609/aaai.v37i8.26170.
[16] Z. Shen, X. Sheng, L. Wang, Y. Guo, Q. Liu, and X. 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*, 2023, pp. 1212–1222, doi:10.1109/CVPR52729.2023.00123.
[17] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End object detection with transformers," in *Proceedings of the European conference on computer vision*, 2020, pp. 213–229, doi:10.1007/978-3-030-58452-8_13.
[18] B. Cheng, A. G. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," in *Advances in Neural Information Processing Systems*, 2021, pp. 17 864–17 875, doi:10.48550/arXiv.2107.06278.
[19] J. Li, D. Li, S. Savarese, and S. Hoi, "BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*, 2023, pp. 19 730–19 742, doi:10.48550/arXiv.2301.12597.
[20] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2010, pp. 9–14, doi:10.1109/CVPRW.2010.5543273.
[21] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1010–1019, doi:10.1109/CVPR.2016.115.
[22] C. Choy, J. Gwak, and S. Savarese, "4D spatio-temporal convnets: Minkowski convolutional neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3075–3084, doi:10.1109/CVPR.2019.00319.
[23] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008, doi:10.48550/arXiv.1706.03762.
[24] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748, 2018*, 2018, doi:10.48550/arXiv.1807.03748.
[25] J.-X. Zhong, K. Zhou, Q. Hu, B. Wang, N. Trigoni, and A. Markham, "No Pain, Big Gain: Classify dynamic point cloud sequences with static models by fitting feature-level space-time surfaces," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 8500–8510, doi:10.1109/CVPR52688.2022.00832.
[26] H. Wen, Y. Liu, J. Huang, B. Duan, and L. Yi, "Point primitive transformer for long-term 4D point cloud video understanding," in *Proceedings of the European Conference on Computer Vision*, 2022, pp. 19–35, doi:10.1007/978-3-031-19818-2_2.



Xiaoxiao Sheng is currently pursuing the Ph.D. degree in Shanghai Jiao Tong University, China. She received the master's degree with the School of Control Science and Engineering, Shandong University, China, in 2020. Her research interests include action recognition and video understanding.



Zhiqiang Shen is currently pursuing the Ph.D. degree in Shanghai Jiao Tong University, China. He received the master's degree with the School of Control Science and Engineering, Shandong University, China, in 2018. His current research interests include self-supervised representation learning and point cloud understanding.

Longguang Wang received the B.E. degree in Electrical Engineering from Shandong University (SDU), Jinan, China, in 2015, and the Ph.D. degree in Information and Communication Engineering from National University of Defense Technology (NUDT), Changsha, China, in 2022. His current research interests include low-level vision and 3D vision.

Gang Xiao received the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 2005. He is currently a full professor with the school of aeronautics and astronautics, Shanghai Jiao Tong University, director of Advanced Avionics and Intelligent Information Laboratory. His current research interests include image fusion and target tracking, avionics integration and simulation. From 2008 to 2016, he had published 40 papers and 2 books. He received the title of Shanghai Pujiang talent in 2016. He is a member of China aviation society information fusion branch. He was a Visiting Scholar with Cranfield University, UK (2006), University of California, San Diego, USA (2010), Southern Illinois University Edwardsville, USA (2014-2015), respectively.