This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/TCSVT.2023.3284493, IEEE Transactions on Circuits and Systems for Video Technology

SSRL: Self-supervised Spatial-temporal Representation Learning for 3D Action recognition

Zhihao Jin, Yifan Wang, Qicong Wang, Yehu Shen, Hongying Meng, *Senior Member, IEEE*

***Abstract*—For 3D action recognition, the main challenge is to extract long-range semantic information in both temporal and spatial dimensions. In this paper, in order to better excavate long-range semantic information from large number of unlabelled skeleton sequences, we propose Self-supervised Spatial-temporal Representation Learning (SSRL), a contrastive learning framework to learn skeleton representation. SSRL consists of two novel inference tasks that enable the network to learn global semantic information in the temporal and spatial dimensions, respectively. The temporal inference task learns the temporal persistence of human actions through temporally incomplete skeleton sequences. And the spatial inference task learns the spatially coordinated nature of human action through spatially partially skeleton sequence. We design two transformation modules to efficiently realize these two tasks while fitting the encoder network. To avoid the difficulty of constructing and maintaining high-quality negative samples, our proposed framework learns by maintaining consistency among positive samples without the need of any negative sample. Experiments demonstrate that our proposed method can achieve better results in comparison with state-of-the-art methods under a variety of evaluation protocols on NTU RGB+D 60, PKU-MMD and NTU RGB+D 120 datasets.**

***Index Terms*—self-supervised learning, contrastive learning, skeleton action recognition.**

I. Introduction

IN the field of computer vision, 3D action recognition as a fundamental research topic, is closely related to people's lives, and has attracted more and more research attention. Skeleton-based action recognition has become the focus of research due to the robustness and excellent action representations of skeleton data. In recent years, the rapid development of sensors such as Kinect [1] also makes it more convenient to obtain skeleton data, promoting the research of skeleton-based action recognition. In early years, many skeleton-based supervised action recognition methods have been proposed based on manual feature [2], [3]. In the past decade, more deep learning based method utilizing RNN [4], [5], [6], CNN [7], [8], [9] and GCN [10] also have been developed, among which GCN-based methods have shown remarkable performance and gain more and more attention. The variants of GCN-based methods [11], [12], [13], [14], [15], [16] achieve state-of-the-art results on many large-scale datasets.

Z. Jin, Y. Wang and Q. Wang are with the Department of Computer Science and Technology, Xiamen University, Xiamen 361000, China. (e-mail:qcwang@xmu.edu.cn)

Y. Shen is with the College of Mechanical Engineering, Suzhou University of Science and Technology, Suzhou 215009, China.

H. Meng is with the Department of Electronic and Electrical Engineering, Brunel University London, Uxbridge UB8 3PH, UK. (email: hongying.meng@brunel.ac.uk)

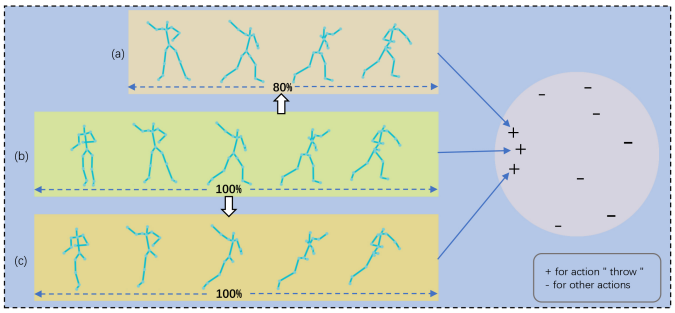

Fig. 1. **Sequence (a) is the temporally incomplete sequence obtained by removing 20% frames from the original skeleton sequence (b). Sequence (c) is the spatially incomplete sequence obtained by removing one body part from (b). Although incomplete in temporal and spatial dimension they still have commonalities with the original sequences. By learning to pull close the incomplete sequence and the complete sequence in feature space, the network can extract more high-level semantic information of human actions.**

However, regardless of the structure used, these methods have to use numerous labelled data to learn skeleton representation. Fully supervised methods inevitably rely on a large amount of annotated data, which is time-consuming, labor-intensive, and resource-intensive. So, how to learn feature representation from large-scale skeleton data without manual annotation becomes an important problem. In recent years, methods are proposed to learn representation in self-supervised manner by designing pretext tasks [17], such as reconstruction [18], auto-regression [19], etc. However the quality of the learnt representation depends heavily on the design of the pretext task. It can also be noted that feature representation learned from pretext tasks is not necessarily favourable for downstream tasks. Lately, since contrastive learning [20], [21], [22], [23], [24], [25] has made good progress in self-supervised learning field, some works [26], [27], [28] leverage contrastive framework to learn skeleton representation. In contrastive learning methods, the network is trained to pull closer different views of the same sample (positive sample) and to push away views of other samples (negative sample). Based on this procedure, the network can learn discriminative representations.

Although existing constrastive learning methods can extract better feature representation to some extent, we argue that there are two issues worthy of attention in previous works: (1) Existing contrastive learning methods mainly help the network extract information by applying various data augmentations. However, the high-level spatio-temporal information of the skeleton is difficult to be reflected by these data augmentation

methods at the coordinate level. The rich semantic information contained in human action is rarely explored. (2) These methods rely on the quality of negative samples and they should be treated carefully. It can be noted that for skeleton action tasks, there are fewer categories in total than that in image tasks, making it harder to guarantee the quality of negative samples. Therefore, the feature representation learned still lacks of discrimination and generality for skeleton-based action recognition.

In order to address the above issues, a new framework of Self-supervised Spatial-temporal Representation Learning (SSRL) is proposed in this paper to learn skeleton representation. As human actions are highly consistent and persistent in temporal sequence, the action recognition task can be performed by only partial sequences in temporal dimension, i.e. temporal partial action sequences contain information that is already sufficient for the action recognition task. The information contained in the complete sequence is redundant, whereas the information shared in temporal partial sequences is more discriminative for action recognition. If the network can be made to learn to mine this property of the skeletal sequence data, the network can learn more essential features of human actions in the temporal dimension. In short, due to the strong continuity of human action in time, the action itself can be inferred given part of the action sequence. Utilizing this property of human actions, we randomly remove part of frames from the skeleton sequence to obtain temporally partial sequences. And we train the network to pull closer temporally complete and temporally partial skeleton sequence from the same action, by minimizing the $l2$ distance between them in feature space, enabling the network to extract high-level temporal semantic information. Moreover, to fully utilize the natural structure of skeleton, we extend the inference task in the temporal dimension to the spatial dimension. The human skeleton has a strong connection among the various parts of the body that interact and co-ordinate with each other in space to form the human action. Similar to the findings in temporal dimension, spatially incomplete skeletal sequences contain a wealth of information that is sufficient for the task of action recognition, and we focus on uncovering information in spatially incomplete skeletal sequences. We divide the skeleton into five parts according to the natural physiological structure of the human body. In order to obtain a partial skeleton sequence in spatial dimension, we randomly remove one part from the sequence. And we train the network to minimize the $l2$ distance between the encoded feature from spatial complete and spatial partial sequence. If the network learns the consistency of spatially partially skeleton sequences and complete sequences, the network can gain more understanding of the role of each part of the human body in human actions. Figure 1 briefly shows how our proposed two inference tasks work. Furthermore, we introduce a contrastive learning framework to implement these two tasks without the need of any negative samples. Particularly, we combine our proposed inference task with the contrasive learning framework in such way that the network can coordinately learn both basic and high-level semantic information.

Our contributions can be summarized as follows:

1) We developed two inference tasks in the temporal and spatial dimension to learn temporal persistence and spatial coordination of human actions in an unsupervised manner, enabling the network to capture more discriminative spatial-temporal feature representations.

2) We propose SSRL, a contrasive learning framework to learn skeleton representation for 3D human action recognition. Our proposed framework works in the manner that only positive samples are needed for pairwise comparison, avoiding the difficulties of constructing high quality negative samples.

3) We validate the representation learned by SSRL on NTU-60, PKU-MMD and NTU-120 under several self-supervised settings such as linear evaluation, semi-supervised evaluation and finetune evaluation protocol, and achieve state-of-the-art results.

II. Related Work

A. Self-supervised representation learning

Self-supervised representation learning is to learn feature representation from a large amount of unlabeled data, usually by designing pretext tasks to generate supervision. In the field of image self-supervised representation learning, many pretext tasks have been designed to learn effective features, such as jigsaw puzzles [29], [30], rotation prediction [31], etc. For sequence data such as video, pretext task based methods on temporal sequence prediction [32], [33], spatiotemporal sequence prediction [34] and speed perception [35] are also proposed. But these methods rely heavily on the quality of the pretext tasks. In recent years, self-supervised methods based on contrastive learning have been proposed for feature learning, which has become a research focus. The main idea of contrastive learning is instance identification, and generally it is realized by pulling in the same sample from different perspectives, and pushing away different samples [20], [21], [22]. Ideally, all samples that are different from the current sample should be compared, which is difficult to achieve, and corresponding solutions have emerged. He et al. [24] and Chen et al. [25] proposed to use a memory bank to store the encoded negative samples, and a momentum update mechanism is used. SimCLR [23] adopted a much larger batch size to compute embeddings of negative samples in real-time. Pan et al. [36] proposed VideoMoCo based on MoCo [24] for unsupervised video representation learning. Dorkenwald et al. [37] combined a shuffling pretext task with the contrastive learning framework. Some researchers [38], [39] have also proposed that negative samples are not necessarily needed in contrastive learning. Caron et al. [38] propose to compare cluster assignments under different views instead of directly comparing features. Grill et al. [39] add a prediction head to the siamese network structure and learn representation by pulling closer the features of the positive samples without using any negative sample.

B. Supervised skeleton-based action recognition

Early skeleton-based action recognition methods are generally based on hand-crafted features to extract information [40],

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/TCSVT.2023.3284493, IEEE Transactions on Circuits and Systems for Video Technology

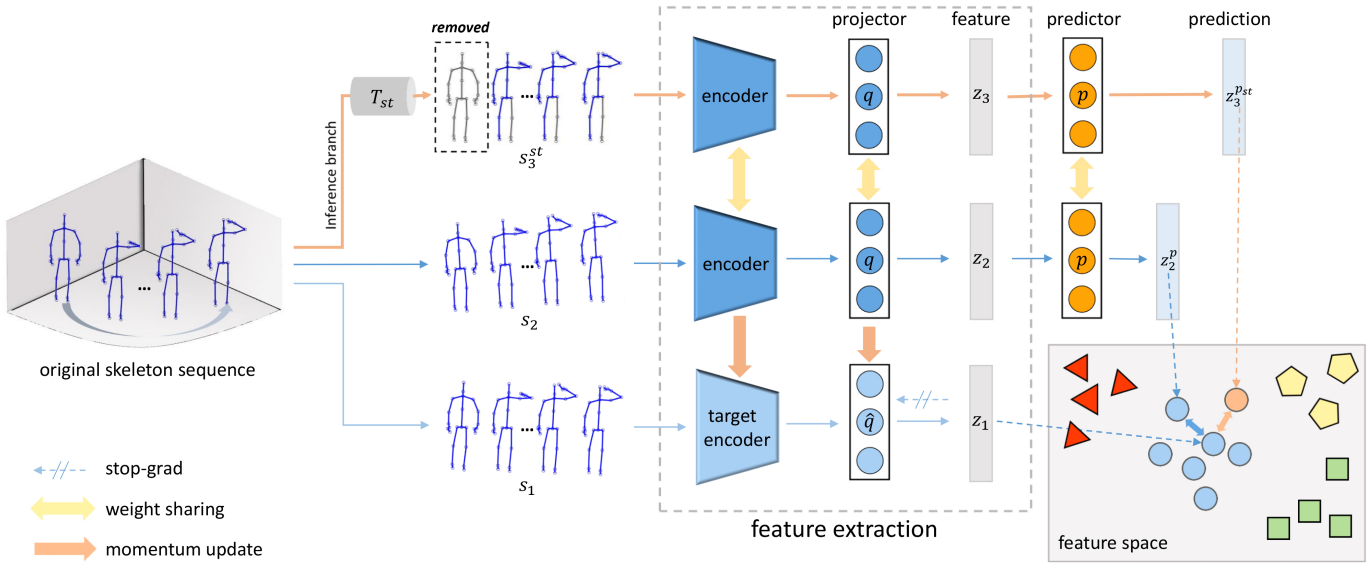

Fig. 2. **Total pipeline of SSRL. Skeleton sequence s_1 and s_2 are obtained by data augmentation from original sequence. Aside from augmentation, we apply spatial-temporal transformations T_{st} to get incomplete sequence s_3^{st}, the grey part of the skeleton s_3^{st} represents the part removed by the transformation. s_2 and s_3^{st} are then fed into the encoder and the projector q to obtain feature z_2 and z_3. The momentum updated target encoder and projector $\hat{q}$ extract feature z_1 from s_1. Instead of directly making comparison between features, we use a predictor p to further generate z_2^p and $z_3^{p_{st}}$ from z_2 and z_3. The final loss function $\mathcal{L}$ consists of $\mathcal{L}^{basic}$ calculated by comparing between z_1 and z_2^p, and $\mathcal{L}^{infer}$ calculated by comparing between $z_3^{p_{st}}$. By introducing additional spatial-temporal inference tasks, SSRL is able to extract high-level semantic features in both the spatial and temporal dimensions while not requiring any negative samples.**

[2], [3], [41]. With the rapid development of deep learning, researchers began to use methods based on deep neural networks [4], [5], [6], [7], [8], [9]. Considering the sequence structure of skeleton data, some RNN-based skeleton action recognition methods emerged to better extract temporal information [4], [5], [6]. Since recurrent neural networks suffer from gradient vanishing [42], some treat skeleton data as image-like data, and use CNN-based methods to fulfill skeleton action recognition tasks [7], [8], [9]. In recent years, due to the characteristics of graph convolutional networks that can effectively aggregate the spatial-temporal features of skeleton sequences, researchers began to use graph convolutional networks to extract features from skeleton data [10]. Based on the graph convolutional network structure, multi-stream structure was proposed to aggregate skeleton information from different views [11]. In order to better model the spatiotemporal information of the skeleton, many variants of graph convolutional networks have been developed [12], [13], [14], [15], [16]. Li et al. [12] dynamically models the structure of human skeleton graph to enhance the flexibility of the network. Liu et al. [15] expand convolution range in time dimension to extract more spatial-temporal information. In addition, based on modalities such as RGB images and optical flow, there are some researches [43], [44] into the real-time application for action recognition. Luvizon et al. [45] proposed a multi-task real-time framework using RGB images for both pose estimation and action recognition. However, we focus on skeleton-based action recognition in unsupervised manner. In this paper, we adopt the widely-used ST-GCN [10] as our backbone network to extract skeleton feature.

C. Self-supervised skeleton-based action recognition

Although fully supervised skeleton-based action recognition methods have shown promising performance, the expensive cost of annotating numerous data can not be neglected. To overcome this problem, efforts have been made for skeleton-based action recognition methods under unsupervised manner. Zheng et al. [18] proposed to use GAN for sequence reconstruction and an encoder-decoder architecture to learn skeleton representation. Su et al. [19] proposed to strengthen the capability of the encoder by weakening the decoder. Based on a multi-task framework, Lin et al. [17] designd two pretext tasks and add a contrastive loss for representation learning. Yang et al. red[46] introduced motion priors and attention module to help the encoder extract information. Recently, Li et al. [26] introduced the contrastive learning framework into skeleton representation learning and designed a new self-supervised skeleton action recognition framework CrosSCLR. In CrosSCLR, a memory bank is introduced to store negative embeddings and multi-stream information is aggregated for more comprehensive information. Thoker et al. [47] used inter-skeleton contrastive learning to aggregate information from different skeleton sequences. Guo et al. [28] proposed to apply more data augmentations to explore more movement pattern in 3D action. However, existing contrastive learning methods rarely manage to exploit high-level spatial-temporal information contained in action which is crucial for action recognition, and need careful treatment of negative samples. So, to learn more discriminative feature we propose SSRL framework, without the need of complicated data augmentations and any negative sample.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/TCSVT.2023.3284493, IEEE Transactions on Circuits and Systems for Video Technology

TABLE I
NOTATIONS AND DEFINITIONS

Notations	Definitions
$\mathcal{D}$	the train set
N	mini batch size
K	the number of optimization steps
s	skeleton sequence
x_i	i-th frame in sequence
$\mathcal{T}$	data augmentation
c	the truncaction factor
L	linear interpolation
T_{st}	transformation for spatial-temporal inference
f	the encoder
q	the projector
p	the predictor
θ	the parameters of f, q and p
$\hat{f}$	the target encoder updated by momentum
$\hat{p}$	the momentum updated projector
$\hat{q}$	the momentum updated predictor
$\hat{\theta}$	the parameters of $\hat{f}$, $\hat{q}$ and $\hat{p}$
z	the encoded feature
z^p	the prediction
z^p_{st}	the prediction for spatial-temporal inference
α	momentum hyper-parameter
$\mathcal{L}^{basic}_{\theta,\hat{\theta}}$	the loss function for basic branch
$\mathcal{L}^{infer}_{\theta,\hat{\theta}}$	the loss function for inference tasks
$\mathcal{L}$	the total loss
λ	the weight parameter

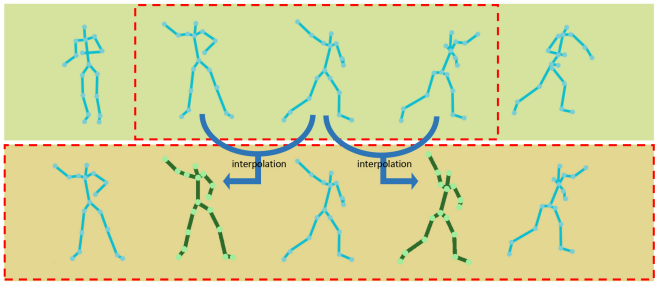

Fig. 3. **The transformation for temporal inference task. The upper dashed box represents the truncated part from the original sequence, and the lower sequence is the new sequence after linear interpolation. The skeleton in dark green is the skeleton we inserted by linear interpolation. The processed sequence still maintains the continuity of the action and contains rich information.**

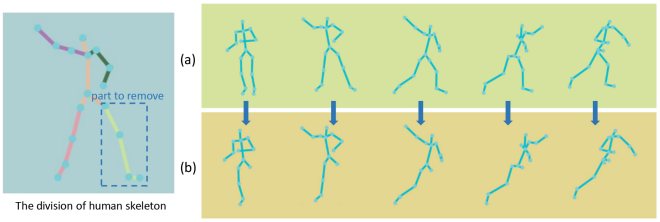

Fig. 4. **The transformation for spatial inference task. The human skeleton on the left shows how we divide the body parts, with each color representing a part. We select a random part and remove the selected part from each frame of the original skeleton sequence a, producing a spatially incomplete sequence b.**

III. METHOD

In this section, we represent our proposed self-supervised learning framework SSRL. Firstly, we give a brief overview of our proposed contrastive framework for skeleton-based action recognition. Then, we introduce our proposed inference tasks in temporal and spatial dimension respectively. Finally, we describe how we combine these two tasks with our framework and the training of the network. The most important symbols are summarized in Table I.

A. Overview

Though self-supervised skeleton representation learning methods have made progress. The existing contrastive learning framework cannot fully utilize the characteristics of skeleton data, and its potential in the field of action recognition remains to be tapped.

Since long-term spatial-temporal information is very important for action recognition task, we expect to exploit them by pursuing consistency between transformed samples. As shown in Figure 2, SSRL mainly contains two modules: 1) Inference branch: it conveys high-level spatial-temporal information for skeleton sequence 2) Basic learning framework: it aims to learn low-level information though simple augmentations. Finally, combining the information of these two modules, we can get more discriminating skeleton representations.

B. Temporal inference task

Inspired by the observation that it is easy to infer action class from temporally incomplete action sequences, we propose a temporal inference task to help the network learn long-range temporal information.

To formulate temporal inference task, firstly a set of data augmentation $\mathcal{T}$ is applied to the original skeleton sequence $s = \{x_1, ..., x_i\}$, and the augmented skeleton sequence is $s^a = \{x^a_1, ..., x^a_i\}$. Then we truncate c% of the augmented sequence, discarding the rest of the sequence. The truncated skeleton sequence is $\overline{s^a} = \{0, ..., , 0, x^a_j, ..., x^a_k, 0, ..., 0\}, k - j = c\% * n$, where n is the total frame number of the original skeleton sequence and c is the truncation factor.

It should be recalled that we adopt ST-GCN as our backbone network, and the input skeleton data of ST-GCN need to be a uniform fixed-length sequence data. After the truncation, there would be some empty frames in the skeleton sequence which can be disadvantageous. So we apply linear interpolation L to fill the entire sequence. The whole transformation process is shown in Figure 3. We can represent the whole transformation as T_{tem}

$$T_{tem} = L(trun(s = \{x_1, ..., x_i\})) \tag{1}$$

where $trun$ represent the truncation operation. Though the interpolated action sequence will be changed to a certain extent in terms of timing rate, but since the coherence of the action itself is not destroyed, the information is still preserved well enough for action recognition task. Finally, the sequence is encoded into feature representation. By learning that the temporally incomplete sequence and the original complete sequence belong to the same action, the network can extract more long-range temporal information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/TCSVT.2023.3284493, IEEE Transactions on Circuits and Systems for Video Technology

It should be mentioned here that our design has some similarities with action prediction task. In fact, the action prediction task can be regarded as a special case of our temporal inference task. The sequence used for action prediction is a partial time series starting from the occurrence of an action, whereas the sequence used by our proposed inference task is a partial time series starting at an arbitrary position. Both the action prediction task and our proposed temporal inference task share the consensus that long-range global information in temporal dimension is one of the key factors for action recognition.

C. Spatial inference task

Although skeleton data and video data are both sequence data, the difference is that the skeleton data itself has an obvious regular structure in the spatial structure. The structure of the skeleton data itself is from the physical structure of the human body, and each part of the body needs to abide by certain physical rules, that is to say, there are mutual constraints and constraints between the various parts of the skeleton data; at the same time, the human action itself is achieved through the cooperation of various parts. In order to make the action happen, there is also a cooperative relationship between the various parts of the skeleton data. Combining the above two points, we can conclude that in the process of human action, each part of the skeleton data is closely related to other parts. From another perspective, the action information of each body part itself implies certain information of other parts.

Based on the above considerations and observations, we further analogize the inference task in time to space, and propose an inference task in the spatial dimension. In our work, the network is to make inferences through spatial incomplete skeleton sequence. To generate spatial incomplete samples, we divide the 3D skeleton into five parts from the spatial structure of the limbs and the torso, which are the main parts of the human body movement and can reflect the human body movement more concisely. Then we randomly pick one of these part and mask all the joints of the picked body part with zero. Figure 4 shows how we divide skeleton structure into five parts and the whole process of the transformation T_{spa}. Our purpose is to enable the network to learn and use the information of the four parts existed under such conditions to understand how the various parts of the human body work together in human action.

D. Full scheme of SSRL

Since negative samples are not necessary in contrastive learning, our proposed method is based on a siamese network structure without negative samples. At the same time, for our proposed inference tasks, since the skeleton data is unlabelled, we can not make inference at category level directly, so we propose to perform the inference tasks in the feature space.

Specifically, given a skeleton sequence s, we first apply random augmentation $\mathcal{T}$ to generate s_1, s_2, s_3, using shear and random rotation. We feed s_2 into the encoder and get representation $f(s_2)$, then we project the representation into lower dimension feature space through a MLP projector q and get representation z_2. Likewise, we feed the view s_1 into the target encoder branch to produce representation z_1. We aim to maximize the similarity between these two representations from different views of the same sample. Instead of directly comparising between z_1 and z_2, we use an additional predictor to further reproduce z_2 into z_2^p

$$z_2^p = p(q(f(s_2;\ \theta)),\ z_1 = \hat{q}(\hat{f}(s_1;\ \hat{\theta})) \tag{2}$$

where θ,$\hat{\theta}$ are the parameters corresponding to the encoder f and the target encoder $\hat{f}$, p represents the predictor. $\hat{f}$ does not do back-propagation and is momentum updated by f

$$\hat{\theta} \leftarrow \alpha\hat{\theta} + (1-\alpha)\theta, \alpha \in [0,1] \tag{3}$$

where α is the momentum hyper-parameter. In SSRL, we use a two-layer MLP for the projector and the predictor. For the third view s_3, before feeding it into the encoder, we apply our transformation module T_{st} on it and generate s_3^{st}, where T_{st} is the combination of T_{tem} and T_{spa}. After the transformation, the information of s_3^{st} in either spatial or temporal dimension has been partially removed. Then we can get the representation $z_3^{p_{st}}$ in the same way as the other two views.

$$z_3^{p_{st}} = p(q(f(T_{st}(s_3);\ \theta))) \tag{4}$$

Our proposed SSRL aims to train the network to learn both common-level information and long-term spatial-temporal information simultaneously, and our optimization objective consists of two parts correspondingly. Firstly the network minimizes the normalized l_2 distance between z_2^p and z_1 [48]

$$\begin{aligned}\mathcal{L}_{\theta,\hat{\theta}}(z_1, z_2) &\triangleq \|\overline{z_2^p} - \overline{z_1}\|_2^2 \\ &= (\overline{z_2^p} - \overline{z_1})^T(\overline{z_2^p} - \overline{z_1}) \\ &= \overline{z_2^p}^T\overline{z_2^p} + \overline{z_1}^T\overline{z_1} - 2\overline{z_2^p}^T\overline{z_1} \\ &= 2 - 2\cdot CosineSimilarity(\overline{z_2^p}, \overline{z_1}) \\ &= 2 - 2\cdot\frac{\langle z_2^p, z_1\rangle}{\|z_2^p\|_2\cdot\|z_1\|_2}\end{aligned} \tag{5}$$

Symmetrically, we separately feed z_1 into the encoder branch and feed z_2 into the target encoder branch to compute $\mathcal{L}_{\theta,\hat{\theta}}(z_2, z_1)$. The loss function of the basic part is obtained by

$$\mathcal{L}_{\theta,\hat{\theta}}^{basic} = \mathcal{L}_{\theta,\hat{\theta}}(z_1, z_2) + \mathcal{L}_{\theta,\hat{\theta}}(z_2, z_1) \tag{6}$$

For the second part, we aim to accomplish the inference task by pulling close the representations of the complete sequence and the incomplete sequence. The network is trained to minimize the difference between z_1 and $z_3^{p_{st}}$, and the difference between z_2 and $z_3^{p_{st}}$. The loss of the inference task can be represented as

$$\mathcal{L}_{\theta,\hat{\theta}}^{infer} = \|\overline{z_3^{p_{st}}} - \overline{z_1}\|_2^2 + \|\overline{z_3^{p_{st}}} - \overline{z_2}\|_2^2 \tag{7}$$

The total loss function of our network can be formulated as

$$\mathcal{L} = \mathcal{L}_{\theta,\hat{\theta}}^{basic} + \lambda\mathcal{L}_{\theta,\hat{\theta}}^{infer} \tag{8}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/TCSVT.2023.3284493, IEEE Transactions on Circuits and Systems for Video Technology

Algorithm 1 Training for SSRL

Input:
set of skeleton data $\mathcal{D}$
parameters of encoder network θ, encoder f projector q and predictor p
parameters of target encoder network $\hat{\theta}$, target encoder $\hat{f}$ and target projector $\hat{q}$
momentum hyper-parameter α, weight parameters λ_1 and λ_2
number of optimization steps K and batch size N
Initialization:
randomly initialize θ and copy θ to $\hat{\theta}$
for $k = 1$ **to** K **do**
$\mathcal{B} \leftarrow \{s_i \in \mathcal{D}\}_{i=1}^{N}$
for $x_i \in \mathcal{B}$ **do**
$s_1 \sim \mathcal{T}, s_2 \sim \mathcal{T}, s_3 \sim \mathcal{T}$
transform for inference tasks $s_3^{st} = T_{st}(s_3)$
$z_2^p = p(q(f(s_2;\ \theta)),\ z_1 = \hat{q}(\hat{f}(s_1;\ \hat{\theta}))$
$z_1^p = p(q(f(s_1;\ \theta)),\ z_2 = \hat{q}(\hat{f}(s_2;\ \hat{\theta}))$
$z_3^{p_{st}} = p(q(f(T_{st}(s_3);\ \theta)))$
$\mathcal{L}_{\theta,\hat{\theta}}^{basic} = \|\overline{z_2^p} - \overline{z_1}\|_2^2 + \|\overline{z_1^p} - \overline{z_2}\|_2^2$
$\mathcal{L}_{\theta,\hat{\theta}}^{infer} = \|\overline{z_3^{p_{st}}} - \overline{z_1}\|_2^2 + \|\overline{z_3^{p_{st}}} - \overline{z_2}\|_2^2$
$\mathcal{L} = \lambda_1 \mathcal{L}_{\theta,\hat{\theta}}^{basic} + \lambda 2 \mathcal{L}_{\theta,\hat{\theta}}^{infer}$
end
update θ by back-propagation
momentum update $\hat{\theta}$
end
Output: encoder f

where λ is the weight parameter used to reflect the proportion of basic contrastive learning tasks and spatial-temporal inference tasks in the entire optimization process. The Algorithm 1 provides the training procedure of our proposed framework.

IV. Experimental Results

A. Dataset

NTU RGB+D 60 Dataset [49] is a large-scale benchmark for action recognition including 56,578 videos with 60 action labels and 25 joints for each skeleton body. There are two recommended evaluation protocols: cross-subject (xsub) and cross-view (xview). For xsub setting, the subject IDs of training subjects are: 1, 2, 4, 5, 8, 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31, 34, 35, 38, and the remaining half are used as testing set. In xview setting, samples captured by camera 2 and camera 3 are used for training, and those captured by camera 1 are used for testing.

PKU-MMD Dataset [50] is a new large-scale benchmark for 3D human action analytics. It consist of almost 20,000 action instance and more than 5 million frames in 51 action classes. For skeleton sequence data, each skeleton sample contains 25 joints. PKU-MMD comprises two subsets under different settings Part I and Part II. Compared to Part I, Part II is more challenging due to more complex views. We conduct experiments under the cross subject protocol on the two subsets respectively.

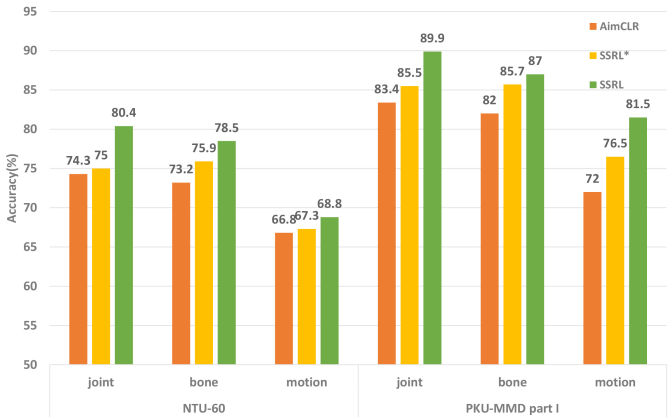

Fig. 5. **Ablation study on the effects of the proposed framework. SSRL* represents a variant of SSRL without our proposed inference tasks, and it applies the data augmentations in AimCLR[28].**

NTU RGB+D 120 Datase[51] is an extended version of NTU RGB+D 60 Dataset and is the largest benchmark for 3D action recognition currently. It contains 113, 945 skeleton sequences in 120 action categories. There are two evaluation protocols recommended: cross-subject (xsub) and cross-set (xset). In xsub, 53 subjects are used for training and the rest are for testing. In xset setting, 32 sets are divided evenly into two parts for training and testing respectively.

B. Experimental Settings

All the experiments are conducted on the PyTorch [52] framework. We pre-process the skeleton data following CrosS-CLR [26] and AimCLR [28] for fair comparison. We train the network on one NVIDIA RTX 3090 GPU with a mini-batch size of 128. The number of model parameters is 2.007M, and GFLOPs is 2.8. For all datasets and evaluation protocols, we report the top-1 accuracy.

Self-supervised Pretext Training We adopt ST-GCN [10] as our encoder, and reduce the number of the channels in each layer into 1/4 of the original setting following CrosSCLR [26] and AimCLR [28]. For data augmentation setting, we set shear amplitude β = 0.5. The weight parameters λ is set to 1. For optimization, we use SGD with momentum (0.9) and weight decay (0.0001). We use 0.99 for the momentum hyper-parameter α. The model is trained for 300 epochs with learning rate of 0.1 with no learning rate adjustment, and it takes about 460 minutes to pre-train the model. We also generate bone and motion stream data from the skeleton sequence to adopt three-stream fusion. For our reported three-stream results, we use the weights of [0.6, 0.6, 0.4] for stream fusion following other GCN-based multi-stream methods. For temporal inference task, we use 0.8 as the truncation factor. For spatial inference task on motion stream, we reverse the temporal order of the randomly selected body part instead of setting the coordinates to zero.

Linear Evaluation Protocol The models are verified by linear evaluation for action recognition task. Specifically, we freeze the weights of the encoder and train a linear classifier

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/TCSVT.2023.3284493, IEEE Transactions on Circuits and Systems for Video Technology

TABLE II
LINEAR EVALUATION RESULTS COMPARED WITH AIMCLR ON NTU-60, PKU-MMD, AND NTU-120 DATASET. "3S" MEANS USING THREE STREAM FUSION.

Method	Stream	NTU-60(%)				PKU(%)		NTU-120(%)			
		xsub		xview		part I		xsub		xset	
		acc.	gain	acc.	gain	acc.	gain	acc.	gain	acc.	gain
AimCLR(AAAI 22)[28]	joint	74.3		79.7		83.4		63.4		63.4	
SSRL(ours)		**80.4**	6.1↑	**82.0**	2.3↑	**89.9**	6.5↑	**68.0**	*4.6*↑	**68.6**	5.2↑
AimCLR(AAAI 22)[28]	bone	73.2		77.0		82.0		62.9		63.4	
SSRL(ours)		**78.5**	5.3↑	**80.5**	3.5↑	**87.0**	5.0↑	**65.1**	2.2↑	**65.1**	1.7↑
AimCLR(AAAI 22)[28]	motion	66.8		70.6		72.0		57.3		54.4	
SSRL(ours)		**68.8**	2.0↑	**76.7**	6.1↑	**81.5**	9.5↑	**58.6**	1.3↑	**58.3**	3.9↑
AimCLR(AAAI 22)[28]	three stream	78.9		83.8		87.8		68.2		68.8	
SSRL(ours)		**81.6**	2.7↑	**85.1**	1.3↑	**90.9**	3.1↑	**69.2**	1.0↑	**71.5**	2.7↑

TABLE III
LINEAR EVALUATION RESULTS OF DIFFERENT SETTING, W MEANS USING THE MODULE

w/TI	w/SI	NTU-60(%)	
		xsub	xview
		66.1	71.0
✓		76.4	79.0
	✓	77.5	80.6
✓	✓	**80.4**	**82.0**

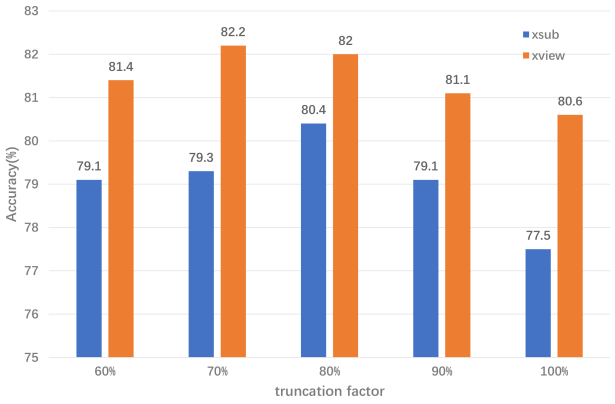

Fig. 6. **Results of pre-trained SSRL with various truncation factors on NTU-60. The models are linear evaluated on joint stream.**

TABLE IV
ABLATION STUDY OF SSRL WITH VARIOUS SPATIAL TRANSFORMATION METHODS ON NTU-60

spatial transformation	NTU-60(%)	
	xsub	xview
randomly remove 5 joints	75.1	79.1
randomly remove 6 joints	75.9	79.8
randomly remove 7 joints	76.2	79.3
randomly remove 1 part	**80.4**	**82.0**

TABLE V
ABLATION STUDY OF SSRL WITH TEMPORAL TRANSFORMATION METHODS ON NTU-60

temporal transformation	NTU-60(%)	
	xsub	xview
without linear interpolation	77.0	80.5
with linear interpolation	**80.4**	**82.0**

TABLE VI
ABLATION STUDY ON MOMENTUM PARAMETER α AND WEIGHT PARAMETER λ ON NTU-60

α	λ	NTU-60(%)
0.9	1.0	77.25
0.99	1.0	**80.4**
0.995	1.0	79.5
0.999	1.0	74.1
0.99	0.6	76.8
0.99	0.8	79.7
0.99	1.2	78.7
0.99	1.4	79.6

composed of a fully-connected layer and a softmax layer. We train the classifier for 100 epochs with learning rate 3 (multiplied by 0.1 at epoch 80).

Finetune Protocol We append a linear classifier to the trainable encoder, and train the whole model for action recognition task in comparison with fully-supervised methods.

Semi-supervised Evaluation Protocol After pre-training with all data, we finetune the model with only 1% or 10% randomly selected data for action recognition task.

C. Ablation Study

We conduct ablation studies to verify the effectiveness of different components of our method.

The effectiveness of SSRL To verify the effectiveness of our proposed method against recent contrastive learning method AimCLR [28], we conduct experiments for all three streams on three benchmarks. As shown in TABLE II, for all the three streams of the three datasets, SSRL outperforms the AimCLR. The performance of our SSRL in joint stream is outstanding and can achieve the performance of AimCLR using

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/TCSVT.2023.3284493, IEEE Transactions on Circuits and Systems for Video Technology

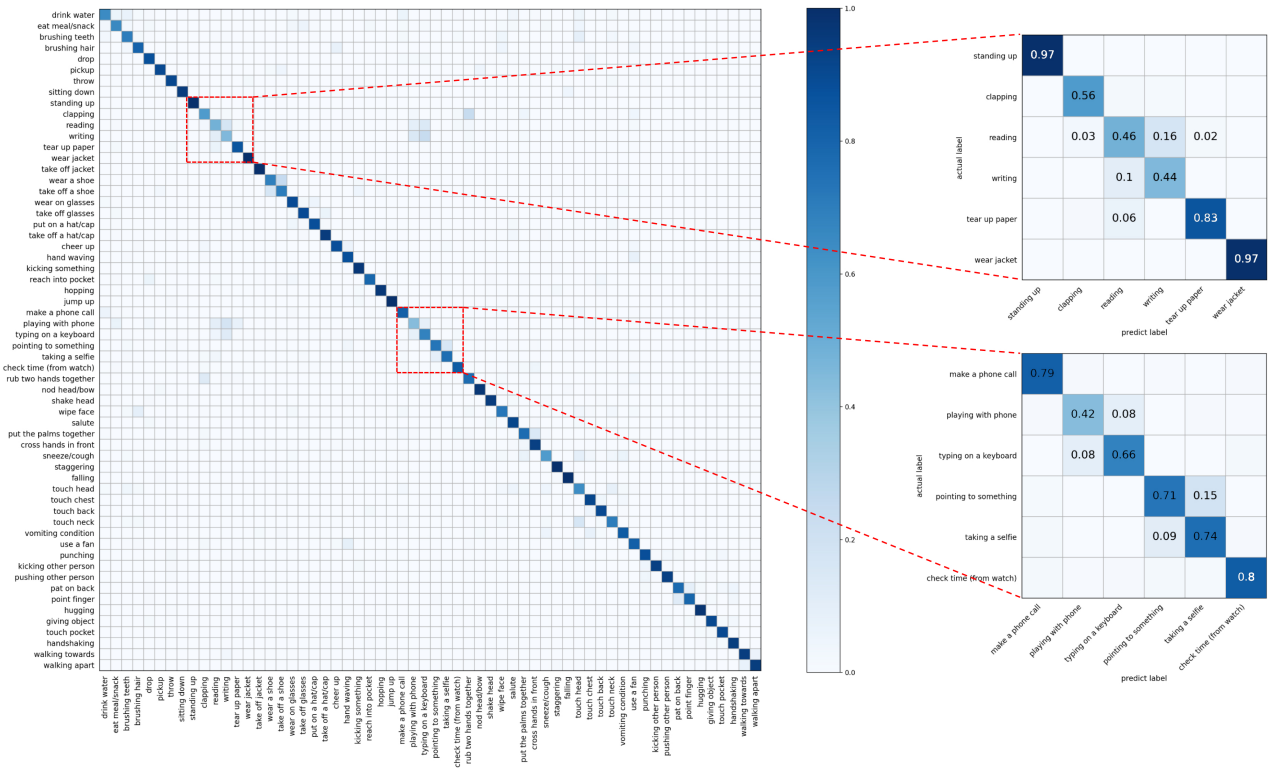

Fig. 7. **Confusion matrix on NTU-60 under xsub setting.**

three streams. It should be noted that AimCLR uses 8 different data augmentations to help learn skeleton representations. In our SSRL, we simply use two basic data augmentations and it is the inference tasks that effectively help the network learn more advanced semantic information.

We also conduct experiments to analyse the effectiveness of our proposed network framework without negative samples. For fair comparison, we remove our proposed inference tasks and replace the data augmentations in SSRL with those in AimCLR [28], which is named SSRL*. The linear evalution results on NTU-60 and PKU-MMD part I are shown in Figure 5. It can be seen that SSRL* without inference tasks still outperforms AimCLR in all three streams on both benchmarks. This proves that it is necessary to remove negative samples for the skeleton-based action recognition task. In addition, the performance of SSRL is higher than that of SSRL*, indicating a further improvement in the network's ability to extract discriminative features with the help of the inference tasks.

We plot confusion matrix results on NTU-60 and PKU-MMD part I datasets respectively in Figure 6 and Figure 7. It can be seen from the confusion matrix, recognition errors are mainly concentrated in hand movements such as reading, writing, using a mobile phone and using a keyboard, which are more difficult to recognize for subtle movements with little body variation, while our proposed method achieves good accuracy in the remaining categories.

The effectiveness of the inference tasks To verify the effect of our proposed inference tasks, we conduct linear evaluation on NTU-60 on several settings. From Table III, the performance of the network without any inference tasks drop severely, because the network learns only at the underlying coordinate level which does not contain enough semantic information. The models with temporal inference task and spatial inference task are allowed to learn high-level semantic information and the effects improve significantly. It should be noted that the effect of the model with only spatial inference task outperforms that with only temporal inference task both in xsub and xview, we suppose the spatial inference task itself contains certain temporal information which gives some edge. Combined with both these two inference tasks, the performance of our SSRL improves over the performance of the model with single task by about 4%. In summary, the proposed inference tasks can help the network to learn more semantic information in temporal and spatial dimension and extract more discriminating representations.

The effects of transformation methods We conduct experiments on the spatial transformation and the temporal transformation. For spatial transformation, we try to remove some joints in a completely random way. It is shown in Table IV, compared to randomly removing one part as in SSRL, the performance of randomly removing some joints from the skeleton are lower of about 5% and 3% respectively in xsub and xview setting. And from Table V, we can see when applying temporal inference task without linear interpolation, the results decreases since there are empty frames in the input sequence which may influence the performance of the backbone. It can be concluded that our proposed transformation

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/TCSVT.2023.3284493, IEEE Transactions on Circuits and Systems for Video Technology

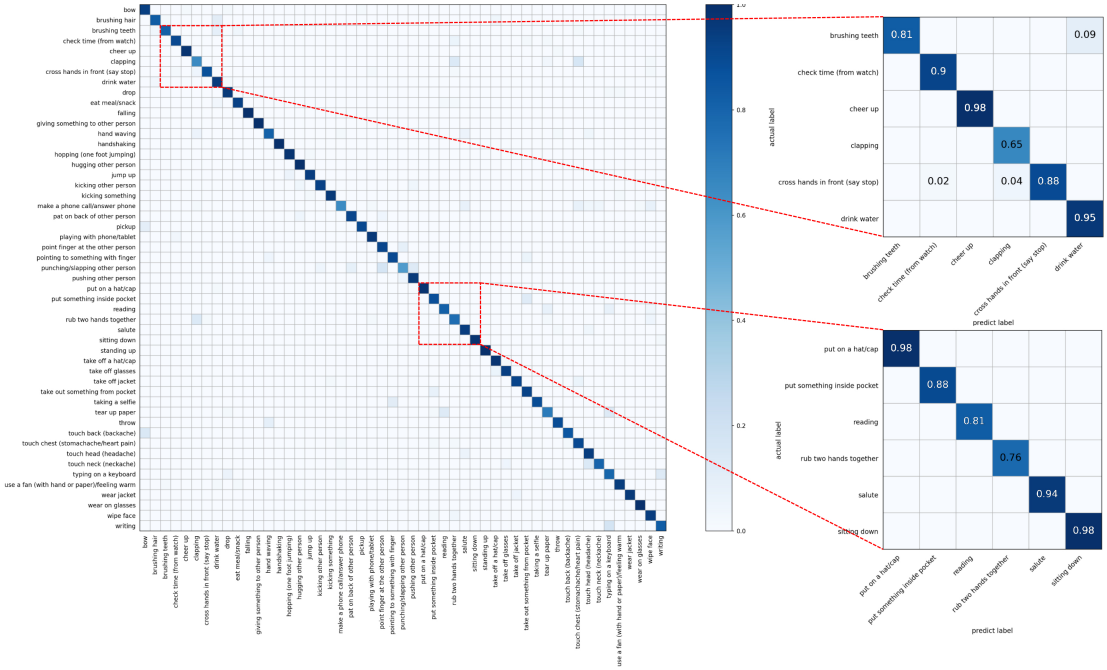

Fig. 8. **Confusion matrix on PKU-MMD part I.**

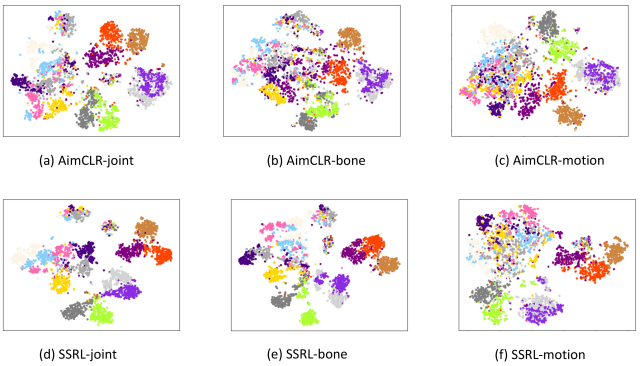

Fig. 9. **The t-SNE visualization of embeddings on NTU-60 xsub.**

methods generate better inputs for the downstream task.

The effects of truncation factor Hyper-parameter c determines in what proportion of the sequence is truncated, and the selection of this parameter directly affects whether the network can learn enough temporal information from the sequence. Too large parameter will degrade the network into that with only spatial inference task, while too small parameter will cause too much timing information to be removed. From existed works on action prediction [53], [54], [55], [56], it can be concluded that about 80% of the sequences can achieve the same classification accuracy as the complete sequences. This means that for most actions, 80% of the sequence length contains sufficient information to perform the action recognition task. Based on this finding, we conduct experiments to find the most suitable truncation factor for temporal inference task. As shown in Figure 6, the performance at 80% is the best overall.

TABLE VII
LINEAR EVALUATION RESULTS ON NTU-60 DATASET.

Method	Backbone	NTU-60(%) xsub	NTU-60(%) xview
single-stream			
LongT GAN(AAAI 18) [58]	RNN	39.1	48.1
MS2L(ACM MM 20) [17]	RNN	52.6	-
P&C(CVPR 20) [19]	RNN	50.7	76.3
SeBiReNet(ECCV 20) [59]	RNN	-	79.7
PCRP(TMM 21) [60]	RNN	53.9	63.5
AS-CAL(Information Sciences 21) [27]	RNN	58.5	64.8
CRRL(21) [61]	RNN	67.6	73.8
MG-AL(TCSVT 22) [46]	GCN	58.6	59.1
SkeletonCLR(CVPR 21) [26]	GCN	68.3	76.4
AimCLR(AAAI 22) [28]	GCN	74.3	79.7
SSRL(ours)	GCN	**80.4**	**82.0**
three-stream:			
3s-SkeletonCLR(CVPR 21) [26]	GCN	75.0	79.8
3s-Colorization(ICCV 21) [62]	CNN	75.2	83.1
3s-CrosSCLR(CVPR 21) [26]	GCN	77.8	83.4
3s-AimCLR(AAAI 22) [28]	GCN	78.9	83.8
3s-SSRL(ours)	GCN	**81.6**	**85.1**

The effects of hyper-parameters From the data in Table VI, a momentum parameter of 0.99 reaches the best performance over other settings. And for the weight parameter λ, we choose 1.0 for SSRL for a better balance of the information from different inputs.

Qualitative Results We apply t-SNE [57] with fix settings to show the embedding distribution of SSRL and AimCLR on 300 epochs of pre-training in Figure 9. From the visualized results, embeddings of SSLR are more closely clustered than AimCLR on three streams, which indicates that SSRL can generate more discriminative features.

TABLE VIII
LINEAR EVALUATION RESULTS ON PKU-MMD DATASET.

Method	part I(%)	part II(%)
Fully-supervised:		
ST-GCN(AAAI 18)[10]	84.1	48.2
VA-LSTM(TPAMI 19)[63]	84.1	50.0
Self-supervised		
LongT GAN(AAAI 18)[58]	67.7	26.0
MS2L(ACM MM 20)[17]	64.9	27.6
3s-CrosSCLR(CVPR 21)[26]	84.9	21.2
ISC(ACM MM 21)[47]	80.9	36.0
3s-AimCLR(AAAI 22)[28]	87.8	38.5
3s-SSRL(ours)	**90.9**	**50.2**

TABLE IX
LINEAR EVALUATION RESULTS ON NTU-120 DATASET.

Method	NTU-120(%)	
	xsub	xset
P&C(CVPR 20)[19]	42.7	41.7
AS-CAL(Information Sciences 21)[27]	48.6	49.2
CRRL(21)[61]	56.2	57.0
3s-CrosSCLR(CVPR 21)[26]	67.9	66.7
ISC(ACM MM 21)[47]	67.9	67.1
3s-AimCLR(AAAI 22)[28]	68.2	68.8
3s-SSRL(ours)	**69.2**	**71.5**

TABLE X
FINETUNE RESULTS ON NTU-60 AND NTU-120 DATASET.

Method	NTU-60(%)		NTU-120(%)	
	xsub	xview	xsub	xset
single-stream				
SkeletonCLR(CVPR 21)[26]	82.2	88.9	73.6	75.3
AimCLR(AAAI 22)[28]	83.0	89.2	76.4	76.7
SSRL(ours)	**83.2**	**90.3**	**76.5**	**77.2**
three-stream				
3s-ST-GCN(AAAI 18)[10]	85.2	91.4	77.2	77.1
3s-CrosSCLR(CVPR 21)[26]	86.2	92.5	**80.5**	80.4
AimCLR(AAAI 22)[28]	86.9	92.8	80.1	80.9
SSRL (ours)	**87.0**	**93.0**	80.3	**81.7**

TABLE XI
SEMI-SUPERVISED RESULTS ON NTU-60 DATASET.

Method	NTU-60(%)			
	xsub		xview	
	(1%)	(10%)	(1%)	(10%)
LongT GAN(AAAI 18)[58]	35.2	62.0	-	-
MS2L(ACM MM 20)[17]	33.1	65.2	-	-
ISC(ACM MM 21)[47]	35.7	65.9	38.1	72.5
3s-CrosSCLR(CVPR 21)[26]	51.1	74.4	50.0	77.8
3s-Colorization(ICCV 21)[62]	48.3	71.7	52.5	78.9
3s-AimCLR(AAAI 22)[28]	54.8	78.2	54.3	81.6
3s-SSRL(ours)	**61.2**	**79.4**	**56.3**	**82.0**

D. Comparison with State-of-the-art Methods

We compare our proposed SSRL with other existed state-of-the-art methods under linear evaluation, finetune protocol and semi-supervised evaluation. We also conduct experiments on its inference speed.

Linear Evaluation Results on NTU-60 As shown in Table VII, our proposed SSRL outperforms all other methods in both single stream setting and three-stream setting. It is noted that, compared to recent contrastive learning method 3s-CrosSCLR [26] and 3s-AimCLR [28], 3s-SSRL leads by 2.7% and 1.3% respectively in xsub and xview. The performance of single-stream SSRL in xsub setting surpasses the performance of other three-stream methods, indicating the effectiveness of our proposed method.

Linear Evaluation Results on PKU-MMD From Table VIII, it can be witnessed that our proposed SSRL is ahead of other state-of-the-art methods. For PKU-MMD part I, SSRL takes a lead of 3.1%. And it is worth noting that for the more difficulty part II, the performance of SSRL reaches the level of some supervised methods, which proves that the inference tasks help the model to learn high-level semantic information in skeleton sequences.

Linear Evaluation Results on NTU-120 As shown in TABLE IX, our 3s-SSRL outperforms the other self-supervised methods on both xsub and xset settings. Compared to the advanced contrastive learning method AimCLR, our method leads 1% and 2.7% in the two settings respectively. It indicates that our proposed method can perform well on large-scale datasets with more categories.

Finetuned Evaluation Results For fair comparison, the ST-GCN [10] in Table VI has the same number of parameters as our encoder network. From Table X, for single-stream, our proposed SSRL perform better than other contrastive methods. And for three-stream, the results of SSRL are also competitive compared to state-of-the-art methods including supervised ST-GCN, indicating the effectiveness of our method.

Semi-supervised Evaluation Results We conduct experiments under semi-supervised evaluation on NTU-60 and PKU-MMD. From Table XI and Table XII, either using 1% or 10% of labelled data, our proposed method outperform other unsupervised method by a considerable margin on each benchmark. And as shown in the table, in the case of using only 1% labelled data, the improvement of our method over other methods is more significant. T his s hows t he a bility o f SSRL on extracting more discriminative skeleton representations.

Inference Speed We also evaluate the inference speed of SSRL, considering the potential need for the model to be deployed in real-world application. We used one single NVIDIA GTX 1080ti for the inference experiment, and obtained the speed of 105 frame per second for SSRL, which is more than good enough for many real-time applications. Recently there are some real-time action recognition methods in supervised manner [43], [44], [45] with excellent performance. However, we could not compare our method with these methods directly

TABLE XII
SEMI-SUPERVISED RESULTS ON PKU-MMD DATASET.

Method	PKU-MMD(%)			
	part I		part II	
	(1%)	(10%)	(1%)	(10%)
LongT GAN(AAAI 18)[58]	35.8	69.5	12.4	25.7
MS2L(ACM MM 20)[17]	36.4	70.3	13.0	26.1
ISC(ACM MM 21)[47]	37.7	72.1	-	-
3s-CrosSCLR(CVPR 21)[26]	49.7	82.9	10.2	28.6
3s-AimCLR(AAAI 22)[28]	57.5	86.1	15.1	33.4
3s-SSRL(ours)	**63.1**	**87.7**	**21.2**	**36.7**

as they used modalities such as RGB images or optical flow while we used skeleton. Of course, skeleton can be extracted from videos, but it is out of the scope of this paper. By the way, it can be noted that the decisive factor in inference speed of our proposed method is the encoder network. However, our proposed framework can be used flexibly with other encoders as well, and the inference speed can be further improved if another efficient encoder is chosen and integrated into our framework.

V. Conclusion

In this paper, we propose a contrastive learning framework for self-supervised 3D skeleton-based action representation. To mine the spatial-temporal features of human actions, it integrates two novel inference tasks to help exploiting high-level semantic information in both spatial and temporal dimension. By pursuing consistency of samples, it can learn effective skeleton representation without the need of negative samples. Experiments on a variety of evaluation protocols show that our SSRL outperforms significantly against state-of-the-art methods, verifying the high quality of skeleton representation.

References

[1] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE multimedia*, vol. 19, no. 2, pp. 4–10, 2012.

[2] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3d skeletons as points in a lie group," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 588–595.

[3] R. Vemulapalli and R. Chellapa, "Rolling rotations for recognizing human actions from 3d skeletal data," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4471–4479.

[4] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1110–1118.

[5] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "Spatio-temporal attention-based lstm networks for 3d action recognition and detection," *IEEE Transactions on image processing*, vol. 27, no. 7, pp. 3459–3471, 2018.

[6] X. Jiang, K. Xu, and T. Sun, "Action recognition scheme based on skeleton representation with ds-lstm network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 7, pp. 2129–2140, 2019.

[7] Y. Du, Y. Fu, and L. Wang, "Skeleton based action recognition with convolutional neural network," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*. IEEE, 2015, pp. 579–583.

[8] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3d action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3288–3297.

[9] A. Banerjee, P. K. Singh, and R. Sarkar, "Fuzzy integral-based cnn classifier fusion for 3d skeleton action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 6, pp. 2206–2216, 2020.

[10] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Thirty-second AAAI conference on artificial intelligence*, 2018.

[11] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 12 026–12 035.

[12] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Actional-structural graph convolutional networks for skeleton-based action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3595–3603.

[13] Y.-F. Song, Z. Zhang, C. Shan, and L. Wang, "Richly activated graph convolutional network for robust skeleton-based action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 5, pp. 1915–1925, 2020.

[14] S. Miao, Y. Hou, Z. Gao, M. Xu, and W. Li, "A central difference graph convolutional operator for skeleton-based action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[15] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang, "Disentangling and unifying graph convolutions for skeleton-based action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 143–152.

[16] Y. Chen, Z. Zhang, C. Yuan, B. Li, Y. Deng, and W. Hu, "Channel-wise topology refinement graph convolution for skeleton-based action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 13 359–13 368.

[17] L. Lin, S. Song, W. Yang, and J. Liu, "Ms2l: Multi-task self-supervised learning for skeleton based action recognition," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2490–2498.

[18] N. Zheng, J. Wen, R. Liu, L. Long, J. Dai, and Z. Gong, "Unsupervised representation learning with long-term dynamics for skeleton based action recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[19] K. Su, X. Liu, and E. Shlizerman, "Predict & cluster: Unsupervised skeleton based action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9631–9640.

[20] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[21] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3733–3742.

[22] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*. Springer, 2020, pp. 776–794.

[23] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[24] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9729–9738.

[25] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.

[26] L. Li, M. Wang, B. Ni, H. Wang, J. Yang, and W. Zhang, "3d human action representation learning via cross-view consistency pursuit," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4741–4750.

[27] H. Rao, S. Xu, X. Hu, J. Cheng, and B. Hu, "Augmented skeleton based contrastive action learning with momentum lstm for unsupervised action recognition," *Information Sciences*, vol. 569, pp. 90–109, 2021.

[28] T. Guo, H. Liu, Z. Chen, M. Liu, T. Wang, and R. Ding, "Contrastive learning from extremely augmented skeleton sequences for self-supervised action recognition," *arXiv preprint arXiv:2112.03590*, 2021.

[29] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European conference on computer vision*. Springer, 2016, pp. 69–84.

[30] C. Wei, L. Xie, X. Ren, Y. Xia, C. Su, J. Liu, Q. Tian, and A. L. Yuille, "Iterative reorganization with weak spatial constraints: Solving arbitrary jigsaw puzzles for unsupervised representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1910–1919.

[31] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," *arXiv preprint arXiv:1803.07728*, 2018.

[32] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: unsupervised learning using temporal order verification," in *European Conference on Computer Vision*. Springer, 2016, pp. 527–544.

[33] J. Huang, Y. Huang, Q. Wang, W. Yang, and H. Meng, "Self-supervised representation learning for videos by segmenting via sampling rate order prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[34] D. Kim, D. Cho, and I. S. Kweon, "Self-supervised video representation learning with space-time cubic puzzles," in *Proceedings of the AAAI*

Conference on Artificial Intelligence, vol. 33, no. 01, 2019, pp. 8545–8552.

[35] P. Chen, D. Huang, D. He, X. Long, R. Zeng, S. Wen, M. Tan, and C. Gan, "Rspnet: Relative speed perception for unsupervised video representation learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 2, 2021, pp. 1045–1053.

[36] T. Pan, Y. Song, T. Yang, W. Jiang, and W. Liu, "Videomoco: Contrastive video representation learning with temporally adversarial examples," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 205–11 214.

[37] M. Dorkenwald, F. Xiao, B. Brattoli, J. Tighe, and D. Modolo, "Scvrl: Shuffled contrastive video representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4132–4141.

[38] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020.

[39] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar *et al.*, "Bootstrap your own latent-a new approach to self-supervised learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 271–21 284, 2020.

[40] B. Ni, G. Wang, and P. Moulin, "Rgbd-hudaact: A color-depth video database for human daily activity recognition," in *2011 IEEE international conference on computer vision workshops (ICCV workshops)*. IEEE, 2011, pp. 1147–1153.

[41] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1290–1297.

[42] S. Hochreiter, Y. Bengio, P. Frasconi, J. Schmidhuber *et al.*, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," 2001.

[43] K. Liu, W. Liu, H. Ma, M. Tan, and C. Gan, "A real-time action representation with temporal encoding and deep compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 2, pp. 647–660, 2020.

[44] B. Zhang, L. Wang, Z. Wang, Y. Qiao, and H. Wang, "Real-time action recognition with deeply transferred motion vector cnns," *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2326–2339, 2018.

[45] D. C. Luvizon, D. Picard, and H. Tabia, "Multi-task deep learning for real-time 3d human pose estimation and action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 8, pp. 2752–2764, 2020.

[46] Y. Yang, G. Liu, and X. Gao, "Motion guided attention learning for self-supervised 3d human action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2022.

[47] F. M. Thoker, H. Doughty, and C. G. Snoek, "Skeleton-contrastive 3d action representation learning," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 1655–1663.

[48] S. Kan, Y. Cen, Y. Li, M. Vladimir, and Z. He, "Local semantic correlation modeling over graph neural networks for deep feature embedding and image retrieval," *IEEE Transactions on Image Processing*, vol. 31, pp. 2988–3003, 2022.

[49] 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 conference on computer vision and pattern recognition*, 2016, pp. 1010–1019.

[50] J. Liu, S. Song, C. Liu, Y. Li, and Y. Hu, "A benchmark dataset and comparison study for multi-modal human action analytics," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 2, pp. 1–24, 2020.

[51] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. C. Kot, "Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 10, pp. 2684–2701, 2019.

[52] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[53] J.-F. Hu, W.-S. Zheng, L. Ma, G. Wang, and J. Lai, "Real-time rgb-d activity prediction by soft regression," in *European Conference on Computer Vision*. Springer, 2016, pp. 280–296.

[54] A. Jain, A. Singh, H. S. Koppula, S. Soh, and A. Saxena, "Recurrent neural networks for driver activity anticipation via sensory-fusion architecture," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 3118–3125.

[55] Q. Ke, J. Liu, M. Bennamoun, H. Rahmani, S. An, F. Sohel, and F. Boussaid, "Global regularizer and temporal-aware cross-entropy for skeleton-based early action recognition," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 729–745.

[56] Q. Ke, M. Bennamoun, H. Rahmani, S. An, F. Sohel, and F. Boussaid, "Learning latent global network for skeleton-based action prediction," *IEEE Transactions on Image Processing*, vol. 29, pp. 959–970, 2019.

[57] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[58] N. Zheng, J. Wen, R. Liu, L. Long, J. Dai, and Z. Gong, "Unsupervised representation learning with long-term dynamics for skeleton based action recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[59] Q. Nie, Z. Liu, and Y. Liu, "Unsupervised 3d human pose representation with viewpoint and pose disentanglement," in *European Conference on Computer Vision*. Springer, 2020, pp. 102–118.

[60] S. Xu, H. Rao, X. Hu, J. Cheng, and B. Hu, "Prototypical contrast and reverse prediction: Unsupervised skeleton based action recognition," *IEEE Transactions on Multimedia*, 2021.

[61] P. Wang, J. Wen, C. Si, Y. Qian, and L. Wang, "Contrast-reconstruction representation learning for self-supervised skeleton-based action recognition," *arXiv preprint arXiv:2111.11051*, 2021.

[62] S. Yang, J. Liu, S. Lu, M. H. Er, and A. C. Kot, "Skeleton cloud colorization for unsupervised 3d action representation learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 13 423–13 433.

[63] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive neural networks for high performance skeleton-based human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 1963–1978, 2019.