Contrastive 3D Human Skeleton Action Representation Learning via CrossMoCo with Spatiotemporal Occlusion Mask Data Augmentation

Qinyang Zeng, Chengju Liu, Ming Liu *Senior Member, IEEE,* Qijun Chen *Senior Member, IEEE*

***Abstract*—Self-supervised learning methods for 3D skeleton-based action recognition via contrastive learning have obtained competitive achievements compared to classical supervised methods. Current researches show that adding a Multilayer Perceptron (MLP) to the top of the base encoder can extract high-level and global positive representations. Using a negative memory bank to store negative samples dynamically can balance the ample storage and feature consistency. However, these methods need to consider that the MLP lacks accurate encoding of fine-grained local features, and a memory bank needs rich and diverse negative sample pairs to match positive representations from different encoders. This paper proposes a new method called Cross Momentum Contrast (CrossMoCo), composed of three parts: ST-GCN encoder, ST-GCN encoder with MLP encoder (ST-MLP encoder), and two independent negative memory banks. The two encoders encode the input data into two positive feature pairs. Learning the cross representations of the two positive pairs is helpful for the model to extract both the global and the local information. Two independent negative memory banks update the negative samples according to different positive representations from two encoders, diversifying the negative samples' distribution and making negative representations close to the positive features. The increasing classification difficulty will improve the model's ability of contrastive learning. In addition, the spatiotemporal occlusion mask data augmentation method is used to enhance positive samples' information diversity. This method takes the adjacent skeleton joints that can form a skeleton bone as a mask unit, which can reduce the information redundancy after data augmentation since adjacent joints may carry similar spatiotemporal information. Experiments on the PKU-MMD Part II dataset, the NTU RGB+D 60 dataset, and the NW-UCLA dataset show that the CrossMoCo framework with spatiotemporal occlusion mask data augmentation has achieved a comparable performance.**

***Index Terms*—Cross contrastive learning, spatiotemporal occlusion mask, human skeleton action recognition.**

I. Introduction

HUMAN action recognition is a promising field, widely used in video surveillance [1], smart home [2] and human-computer interaction [3]–[5]. 3D human skeleton recognition has recently been widespread because of its low computer calculation consumption and strong robustness. Many supervised algorithms have been proposed in recent years [6]–[9]. These algorithms have achieved a high recognition accuracy with sufficient labels. However, data annotations are so expensive and time-consuming that self-supervised learning methods for action recognition have recently become a research hotspot. Contrastive learning methods based on instance discrimination have provided an effective way for self-supervised 3D skeleton action recognition. Positive samples are firstly processed into data with different information by data augmentation, then embedded into high-level semantic representation vectors by encoders. Contrastive learning makes the positive embedding features close and far away from the negative representations in the high-level vector space. However, the low amount of information and sparse skeleton sequence make it difficult for current models to sufficiently extract and discriminate the latent critical spatiotemporal representations, affecting the accuracy of self-supervised 3D skeleton action recognition. In order to improve the ability of the model to extract and discriminate information, we propose the Cross Momentum Contrast (CrossMoCo), mainly including three innovations: two encoders to crosswise learn representations, two independent negative memory banks and a new spatiotemporal occlusion mask data augmentation. These three innovative points specifically address the following three scientific issues.

3D skeleton sequences are low-information and easily affected by the perspective, which leads to the difficulty of extracting critical spatiotemporal information without label guidance. Extracting and Fusing global and local representations will be helpful for the model to locate the critical spatiotemporal information. Most existing methods use a single base encoder to extract representations. The base encoders, such as [6]–[8], can sufficiently extract the local spatiotemporal representations by establishing the adjacency matrix between skeleton joints. However, they cannot accurately extract global representations with a few annotations, leading to inconsistencies between global and local representations. Considering the problem, we add an MLP project head to the top of the base encoder referred to [10], [11]. MLP can capture global features by global mapping from its full connection layers, while it will weaken the model's ability to extract fine-grained features. Combining these methods may be helpful in extracting global and local features. However, simply combining

The work has been financially supported by the National Natural Science Foundation of China under Grant U1713211, Grant 62073245, and Grant 61733013. *(Corresponding author:Chengju Liu.)*

Qinyang Zeng, Chengju Liu and Qijun Chen are with the College of Electrics and Information Engineering, Tongji University, Shanghai, 201804, China. Chengju Liu is also a Chair Professor of Tongji ResearchInstitute of Artificial Intelligence (Suzhou), Suzhou, 215300, China (e-mail: qinyangz@163.com,liuchengju@tongji.edu.cn,qjchen@tongji.edu.cn).

Ming Liu is with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong, 999077, China (e-mail: eelium@ust.hk).

Manuscript received July 16, 2022; revised Jan 11, 2023.

these features tends to cause inconsistency between different features. Considering the idea, we design a feature cross-learning method to sufficiently integrate extracted representations, which crosswise compares query-key pairs embedded by different encoders. Specifically, the query representations embedded by one encoder are compared with the key representations from another encoder to learn positive representations. Our work uses the classic graph convolution network ST-GCN [6] as the base encoder to extract local representations. In addition, we use ST-MLP, connected by the base encoder ST-GCN and MLP, to extract the global representations of the skeleton sequences. Cross-learning representations from the two encoders can make the network effectively integrate their global and local spatiotemporal information. It is helpful for our model to extract essential spatiotemporal information.

Maintaining the consistency between the negative and the positive representations is challenging when the positive sample representations are diverse. Although the above cross-learning global and local representations proposed in our paper are effective, the negative samples with single representations cannot improve the ability of the model to discriminate information when the positive representations are diverse. Improving the similarity between positive and negative representations in the self-supervised training process can increase the difficulty of discrimination, promoting the model to discriminate representations. There are two main ways to store negative samples. One method is using a large batch size to store negative samples [11], which can ensure the negative representations' stability and consistency. However, it requires a large number of storage resources. In order to balance the ample storage and features' stability, another method represented by MoCo [12] has recently been proposed to dynamically store negative samples by a memory bank with the First-In-First-Out stack strategy. The single memory bank in MoCo is challenging to provide high-quality negative samples similar to positive representations with global and local spatiotemporal characteristics. We use two independent negative memory banks to store negative samples, respectively updated by the key representations embedded by the two encoders via the same stack strategy with MoCo. The method effectively improves the similarity of positive and negative representations and increases the difficulty of model discrimination information in the training process.

Common data augmentation methods [13] tend to produce positive skeleton samples with redundant information. They deal with discrete skeleton joints without considering their adjacent skeleton joints. 3D skeleton joints always carry lots of action information related to their adjacent joints that can form skeleton bones in the human skeleton topology through the adjacency matrix. Redundant samples make the information distribution inhomogeneous, which will affect the model to extract and discriminate critical representations. The proposed spatiotemporal occlusion mask data augmentation in our work can remove redundancy. It takes the adjacent skeleton joints as a mask unit. We occlude these mask units in random spatial positions and temporal frames with the random mask proportion. When a joint is occluded, its adjacent joints in the mask unit will also be occluded, removing the information related to the skeleton joint after data augmentation and ensuring the independence and uniformity of the augmented information. High-quality positive samples produced by data augmentation can help extract and discriminate critical representations.

The experimental results on the PKU-MMD Part II dataset [14], the NTU RGB+D 60 dataset [15], and the NW-UCLA dataset [16] show that our proposed CrossMoCo can improve the accuracy of self-supervised 3D skeleton action recognition. We summarize our contributions as follows:

- CrossMoCo framework is proposed. CrossMoCo features the crosswise learning of two positive representation pairs embedded by the two encoders and the two independent negative memory banks, which enhance the ability to extract representations as well as the diversity of negative samples' feature distribution and consistency with the positive representations from different encoders.
- We propose to use the base encoder ST-GCN and the ST-MLP encoder composed of ST-GCN and MLP to generate two different positive query-key pairs, which are used to learn similar features from the positive samples by cross-matching. Crosswise learning can help the model extract critical spatiotemporal information by fusing global and local representations. Two independent negative memory banks are updated according to the two pairs of positive key representations, respectively.
- We propose spatiotemporal occlusion mask data augmentation to mask the skeleton data with a mask unit composed of the occluded joints and their adjacent joints that can form a skeleton bone in the human skeleton topology. Compared with each skeleton joint's independent mask or perturbation data augmentation, the spatiotemporal occlusion mask method can make the generated data carry less redundant information to diversify the features of positive samples.
- Experiments on the PKU-MMD Part II dataset, the NTU RGB+D 60 dataset, and the NW-UCLA dataset show that our CrossMoCo achieves a comparable result.

The remainder of this paper is organized as follows. Section II describes related works on Supervised 3D Skeleton Learning, Self-supervised Contrastive Learning and Self-supervised 3D skeleton human action recognition. Section III represents our proposed CrossMoCo method. Section IV shows the experimental details and results. Section V summarizes our work.

II. Related Works

Supervised 3D Skeleton Learning. Early action recognition algorithms are mainly based on handcraft features [17]–[19]. Deep learning methods are characterized by end-to-end learning and have recently attracted much attention. Algorithms based on RNN can extract spatiotemporal features of successive skeleton frames [20]–[22], but it is likely to suffer from gradient disappearance or gradient explosion as well as colossal calculation. Methods based on CNN algorithm have attracted extensive attention [23]–[25], while they need regular spatiotemporal skeleton data. Methods based on Graph Convolution Network (GCN) can well model irregular skeleton data by establishing an adjacency matrix, making 3D skeleton-based action recognition achieve high accuracy yan2018spatial,li2019actional,shi2019two,liu2020disentangling. In recent years, researchers further develop many GCN-based methods by integrating attention mechanism and multi-streams fusion mechanism [22], [26], [27]. These methods are of high precision and lightweight. In our paper, ST-GCN [6] is taken as the base encoder.

Self-supervised Contrastive Learning. In recent years, many contrastive learning algorithms based on instance discrimination have been proposed, which embed the positive and negative sample features into a high-dimensional space for discrimination. The representations of input data are treated as the anchor. Only the representations of the input samples after data augmentation are positive features. The positive features are pulled close, and the negative features are put away in the feature space. There are many algorithms to enrich positive pairs. Su et al. [28] proposed the encode and decode structures to reconstruct features. Gao et al. [29] proposed to design different data augmentation methods to improve the quantity and quality of positive sample pairs. Improving negative samples is also the research hotspot. Chen et al. [30] proposed to use a large size to compute negative embeddings. He et al. [12] used the memory bank to dynamically store negative embeddings via the First-In-First-Out strategy, which balances the features' stability and diversity. These methods have already achieved excellent performances in the fields of self-supervised image reconstruction and image classification.

Self-supervised 3D Skeleton Human Action Recognition. In recent years, many self-supervised learning methods based on 3D skeleton action recognition have been proposed, such as 3s-CrosSCLR [31] and Skeleton-Contrastive [32], which increase the positive sample pairs via data augmentation and design different encoders to improve the ability of feature representations learning. VideoMoCo [33] improves the memory bank's storage for negative sample pairs. Some popular algorithms improve instance discrimination by enhancing the feature learning ability of the encoder and decoder, such as P & C [28], LongT GAN [34], and MS^{2}L [10]. These methods unilaterally consider the feature extraction of positive pairs or the storage of negative samples, which cannot improve the quality of positive and negative representations at the same time. Besides, they cannot combine the advantages of different encoders to further improve the network's encoding capability. Based on these ideas, we propose the CrossMoCo for self-supervised 3D skeleton action recognition based on 3s-CrosSCLR [31], whose framework is inspired by MoCo [12].

III. CrossMoCo Method

The whole architecture is shown in Fig.1(b). There are two encoders: the base encoder ST-GCN and the combined encoder ST-MLP composed of ST-GCN and MLP in series. Two encoders simultaneously encode positive samples to generate two kinds of query-key feature pairs, and they are cross-dot multiplied to learn the positive representations. Unlike 3s-CrosSCLR referred to the MoCo architecture [12], shown

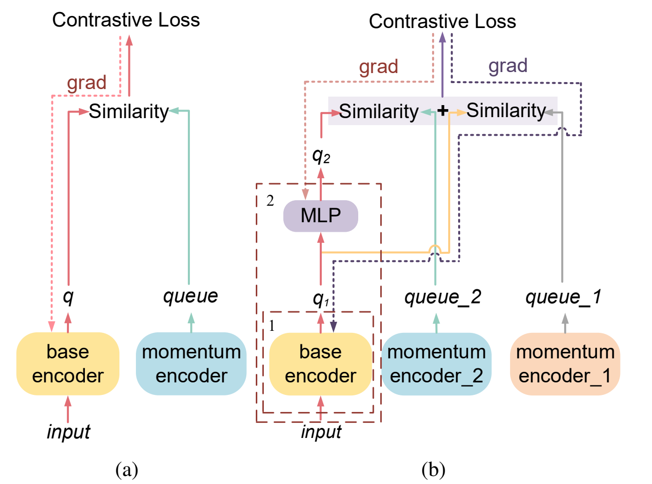


Fig. 1. The illustration of MoCo and CrossMoCo architectures. (a) MoCo architecture. A base encoder embeds the input data into positive query-key pairs. A memory bank, the momentum encoder, stores the negative samples. The encoder's parameters are updated by gradient descent. (b) CrossMoCo architecture. Two encoders are used to produce different positive query-key pairs, a base encoder and a base encoder combined with MLP, which are circled with red dotted lines and respectively marked with numbers 1 and 2 to distinguish. Two memory banks, the momentum encoder_1 and encoder_2, store the negative samples. The CrossMoCo model learns the similarity of two groups of positive and negative pairs. The parameters of the two encoders are updated by gradient descent.

in Fig.1(a), we use two independent negative memory banks to dynamically store negative samples, which are updated according to positive key representations embedded by two encoders, respectively. Fig.1(b) shows that q_1 and q_2 are the positive pairs' embeddings generated by the two encoders, respectively. They are compared with the negative embeddings from the two independent negative memory banks to form two similarity functions, consisting in the final objective function. The parameters of ST-GCN and MLP are updated with gradient descent.

A. *MoCo Architecture Review*

MoCo features a memory bank to dynamically store negative pairs following the First-In-First-Out strategy, which greatly makes use of memory storage. Besides, the parameters of the key encoder are updated by the query encoder without participating in gradient backpropagation. The expression is shown as follows [12]:

$$\theta_k \leftarrow \theta_k + (1-m)\theta_q \tag{1}$$

where $m \in [0,1)$ is the momentum coefficient, and m is vital to balance the feature representations' update speed and stability. The MoCo's contrastive loss function is written as follows [12]:

$$L = -\log \frac{\exp(z \cdot \hat{z}/\tau)}{\exp(z \cdot \hat{z}/\tau) + \sum_{i=1}^{M} \exp(z \cdot m_i/\tau)} \tag{2}$$

where z and $\hat{z}$ are respectively the input data's query feature embeddings and key feature embeddings that are encoded by the two encoders after data augmentation, dot product is used to compute the similarity of embeddings, m_i is the negative

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/TMM.2023.3253048

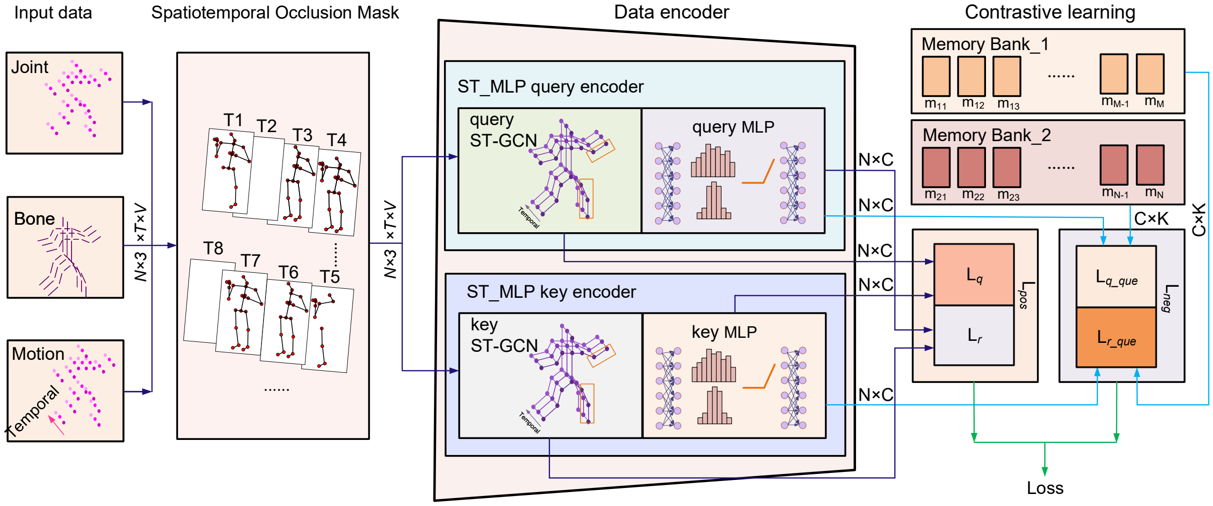


Fig. 2. The architecture diagram of CrossMoCo training original skeleton data. Three data streams composed of the skeleton joint data, the skeleton motion data and the skeleton bone data are respectively input into the network for contrastive learning. The spatiotemporal occlusion mask is used for data augmentation. The encoder can embed the input data into the vector space and generate feature embedding representations, whose dimensions are $N \times 3 \times T \times V$. There are two encoders: the base encoder ST-GCN and the combined encoder ST-MLP, composed of the base encoder and MLP in series. Two encoders simultaneously embed positive sample pairs to generate two kinds of query-key pairs with $N \times C$ dimensions, which are crosswise multiplied and form L_q and L_r. L_{pos}, the similarity of positive feature representations, is the sum of L_q and L_r. Query features generated by ST-GCN and negative samples with $C \times K$ dimensions from memory bank_2 consist in L_{q_que}. L_{r_que} is the similarity of ST-MLP query features and the negative samples from memory bank_1. L_{neg} is the sum of L_{r_que} and L_{q_que}. The contrastive learning loss function is formed by L_{pos} and L_{neg} whose iteration process is similar to MoCo. The query encoder's parameters of ST-GCN and MLP are updated by gradient descent. Parameters in the key encoder are updated according to formula (1).

tensor from the memory bank, and the τ is the temperature hyperparameter. In our work, we propose a new architecture, CrossMoCo, which improves MoCo by introducing query-key features crosswise learning and two independent negative memory banks for storing negative samples. Consecutive skeleton data after data augmentation are sent to two different encoders to produce two sets of query-key pairs, and then they are multiplied crosswise to form the similarity functions of positive pairs. The similarity functions of the two positive pairs and the negative representations from two memory banks are also calculated. These similarity functions are formed into the final similarity loss function.

B. Crosswise Learning Representations & Two Independent Negative Memory Banks

We propose a method of cross-learning representations to improve feature learning. The architecture diagram of CrossMoCo training original skeleton data is shown in Fig. 2. One encoder is used ST-GCN as the base encoder to embed the augmentation skeleton data into the vector space. The other encoder, ST-MLP, consists of ST-GCN and MLP connected in series. The original skeleton joint data are processed into three kinds of data streams as input, i.e., joint data, motion data and bone data, which are put into the data augmentation module to form various positive samples. After spatiotemporal occlusion mask data augmentation, the tensor dimension remains $N \times 3 \times T \times V$. N represents the batch size, 3 represents the number of channels, T is the number of temporal frames, and V is the number of skeleton joints in each frame. The positive data are respectively embedded with the two encoders to generate two kinds of query-key feature pairs, both $N \times C$ tensors. C is the channel number. The query encoder's parameters are updated by gradient descent, while the key encoder's parameters are updated by formula (1). The contrastive learning loss function is composed of L_{pos} and L_{neg}, where L_{pos} is the sum of L_q and L_r. L_q is used to measure the similarity between the positive samples' query features encoded by ST-GCN and the positive samples' key features encoded by ST-MLP. L_r is used to measure the similarity between the positive samples' query features encoded by ST-MLP and the positive sample's key features encoded by ST-GCN. L_{neg} is the sum of L_{q_que} and L_{r_que}. L_{q_que} is used to measure the similarity between the positive samples' query features encoded by ST-GCN and the negative features from a memory bank, updated according to the key features embedded by the ST-MLP. L_{r_que} is used to measure the similarity between the positive samples' query features encoded by ST-MLP and the negative features from the other memory bank, updated by the key features embedded by the ST-GCN encoder. The tensors' dimensions of negative samples from two banks are both $C \times K$. The MLP in ST-MLP is composed of two linear layers of Full Connection (FC), shown in Fig. 3. Normalization and activation functions are used between two FC layers. The inputs x are the positive samples after data augmentation, which are encoded by the two encoders ST-GCN and ST-MLP. Two positive query-key pairs are respectively produced. i.e., $(Z_q^{ST-GCN}(x), Z_k^{ST-GCN}(x))$ and $(Z_q^{ST-MLP}(x), Z_k^{ST-MLP}(x))$. $Z_q^{ST-GCN}(x)$ and $Z_q^{ST-MLP}(x)$ are the query features. $Z_k^{ST-GCN}(x)$ and

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2023.3253048

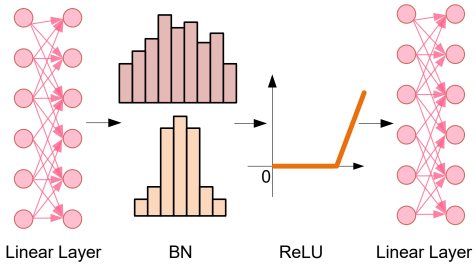


Fig. 3. MLP architecture. MLP is composed of two linear layers of Full Connection (FC). Batch Normalization and ReLU activation functions are used between two FC layers.

$Z_k^{ST-MLP}(x)$ are the key features. The similarity measure of positive pairs representations, $L_{pos}(x)$, is obtained by the intersection dot product between them. It is expressed as follows:

$$\begin{aligned} L_{pos}(x) &= \alpha \exp\left(Z_q^{ST-GCN}(x)\cdot Z_k^{ST-MLP}(x)/\tau\right) \\ &+\beta \exp\left(Z_q^{ST-MLP}(x)\cdot Z_k^{ST-GCN}(x)/\tau\right) \end{aligned} \tag{3}$$

where $\alpha,\beta \in [0,1)$ are the correlation coefficients, reflecting the influence of two parts on $L_{pos}(x)$. Query features produced by two encoders are also carried on dot product with negative pairs from two independent negative memory banks to calculate the similarity loss function between the positive features and negative features, $L_{neg}(x)$, expressed as follows:

$$\begin{aligned} L_{neg}(x) &= \lambda \sum_{n_1\sim N} \exp\left(Z_q^{ST-GCN}(x)\cdot Z_{n1}(x)/\tau\right) \\ &+\varphi \sum_{n_2\sim N*} \exp\left(Z_q^{ST-MLP}(x)\cdot Z_{n2}(x)/\tau\right) \end{aligned} \tag{4}$$

where $\lambda,\varphi \in [0,1)$ are the correlation coefficients, reflecting the impact of the two kinds of similarities between query features from two encoders and negative samples from two independent negative memory banks on $L_{neg}(x)$, N and N* are the two memory banks, which are respectively updated according to the key features embedded by the ST-MLP and ST-GCN encoders. The final contrastive learning loss function $L(x)$ is expressed as follows:

$$L(x) = -\log \frac{L_{pos}(x)}{L_{neg}(x)} \tag{5}$$

The pseudocode for CrossMoCo learning is shown as Algorithm 1.

C. The Importance of Three Streams Skeleton Data for Contrastive Learning

We process the input data x as three streams' data: the skeleton joint data x_{joint}, the motion data x_{motion} and the bone data x_{bone}.

$x_{joint} \in \{X^r(x)\,|r=1,2,3,...,J;t=1,2,3,...,T\}$, where r is the number of skeleton joints and T is the temporal frame. x_{joint} can be expressed in Cartesian coordinates as follows :

$$X^r(t) = (x_t^r, y_t^r, z_t^r)^{\mathrm{T}} \in \mathbb{R}^3 \tag{6}$$

Algorithm 1 Pseudocode of CrossMoCo Pre-training

Input: The three streams data x joint, motion, bone; f, the base encoder ST-GCN; h, base encoder ST-GCN + MLP, ST-MLP; queue_1, queue_2, negatives queue in two memory banks, epoch e for the pretraining epochs
for $epoch$ in e **do**
  for x in $Bactchsize$ **do**
    $x_1, x_2 = aug(x), aug(x)$;
    $q_1, k_1 = f(x_1), f(x_2)$;
    $r_1, l_1 = h(x_1), h(x_2)$;
    compute L_{pos} by $E_q.(3)$;
    compute L_{neg} by $E_q.(4)$;
    compute L by $E_q.(5)$, $E_q.(9)$, $E_q.(10)$;
  end for
  update θ_q by backpropagation;
  update θ_k by $E_q.(1)$;
  enqueue k to $queue$;
  enqueue l to $queue2$;
end for
Output: Optimized the two encoders f and h parameters

The motion data x_{motion} is obtained from the position difference of the same skeleton joint among the adjacent temporal frames and the expression is as follows:

$$x_{motion} = X^r(t+1) - X^r(t) \tag{7}$$

x_{bone} is formed by connecting the adjacent joints in the same frame, expressed as follows:

$$x_{bone} = X^{r+1}(t) - X^r(t) \tag{8}$$

The three streams' data are independently used as the input of the contrastive loss function to produce three contrastive loss functions, i.e., L_{joint} , L_{motion} , L_{bone}. The expression is:

$$\begin{cases} L_{joint} &= L(x_{joint}) \\ L_{motion} &= L(x_{motion}) \\ L_{bone} &= L(x_{bone}) \end{cases} \tag{9}$$

The final contrastive loss function L is shown as:

$$L = aL_{joint} + bL_{motion} + cL_{bone} \tag{10}$$

where $a,b,c \in (0,1]$ are the correlation coefficients, reflecting the impact of three contrastive loss functions on the final loss function.

D. Skeleton Data Augmentation For Contrastive Learning

A critical design of the contrastive learning network is augmenting the input data to get multi-view positive samples. Diverse positive samples will obtain different view information, which is helpful for the encoders to learn abundant semantic representations. The common data augmentation methods include shear [13], crop [35], etc. These augmentation methods may not be suitable for skeleton joints because each 3D skeleton joint contains plenty of information related to adjacent joints during the iterative process. There will be much information redundancy if these methods are applied to the 3D skeleton data augmentation.

Inspired by the actual human skeleton occlusion in monitoring systems, we propose a new spatiotemporal occlusion mask data augmentation method to generate positive samples. The occlusion rate and position are random. Specially, a mask unit is composed of adjacent skeleton joints that can form a skeleton bone in the human skeleton topology rather than a random skeleton joint to reduce the redundant information. Firstly, we randomly mask position of skeleton joints with random different occlusion mask rates. The mask formula is expressed as follows:

$$Mask_j = Mask\left(RandomSampler\left(r \times N_j\right)\right) \tag{11}$$

where $Mask_j$ is the mask matrix of skeleton joints, r is the skeleton joints' spatial occlusion mask rate, N_j is the skeleton joint number, $RandomSampler\left(\cdot\right)$ is the random sampling function, which randomly extracts a certain number of skeleton joints from the complete skeleton joints, and $Mask\left(\cdot\right)$ is the mask function, which can block the selected samples. The final spatial occlusion mask formula of the skeleton joints is as follows:

$$D_{Spatial}\left(X\right) = X \odot Mask \tag{12}$$

where X is the input skeleton joint data matrix, $D_{Spatial}\left(X\right)$ is the skeleton data after the spatial occlusion mask data augment, and $\odot$ is the dot production.

Occlusion often lasts for several temporal frames, which may not be successive when occlusion events occur. Inspired by the phenomenon, we randomly mask some temporal frames for data augmentation. The temporal mask $Mask_t$ is expressed as follows:

$$Mask_t = Mask\left(RandomSampler\left(\beta \odot T\right)\right) \tag{13}$$

where β is the temporal occlusion mask rate, and T is the temporal frames. The temporal frames occlusion mask and the skeleton joints occlusion mask are combined to form the spatiotemporal occlusion mask. The formula $D_{Spatiotemporal}\left(x\right)$ is expressed as follows:

$$D_{Spatiotemporal}\left(X\right) = D_{Temporal}\left(D_{Spatial}\left(x\right)\right) \tag{14}$$

We visualize the spatiotemporal occlusion mask of three actions, i.e., drinking water, jumping up and falling down, shown in Fig.4. t_1, t_2, t_3, t_4 and t_5 represents 10 frames, 20 frames, 30 frames, 40 frames and 50 frames, respectively. The occlusion mask's body part and occlusion rate are different and random between 50 frames. Precisely, in t_1 frames, the left leg, the left foot, the right arm and the right hand are occluded. There is no occlusion in t_2 frames. Both calves and feet are occluded in t_3 frames. The left calves and the right feet are occluded in t_4 frames. The right calves and right feet are occluded in t_5 frames.

IV. Experiments

The effectiveness of spatiotemporal occlusion mask data augmentation, encoders, MLP's layers and the number of independent memory banks are first verified by the PKU MMD Part II dataset. Then our CrossMoCo is compared with the state-of-the-art algorithms on the PKU MMD part II dataset, the NTU RGB+D 60 dataset, and the NW-UCLA dataset. Section A introduces the three classical datasets. Section B shows the experiment settings. Section C clarifies the details of the ablation experiment. Section D compares our algorithm with other advanced methods on the three datasets.

A. Datasets

PKU-MMD Part II Dataset. PKU MMD dataset is a large-scale dataset covering a multi-modality 3D understanding of human actions with almost 20,000 instances and 51 action labels. It consists of two subsets. Part I is an easier version for action recognition, while Part II is more challenging with more noise caused by view variation. In our work, we choose the Part II dataset to conduct experiments and the skeletal sequences in PKU MMD Part II dataset are processed into 50 frames.

NTU RGB+D 60 Dataset. This dataset is a human behavior recognition dataset proposed by Rose Lab of Nanyang University of technology. It contains 60 kinds of actions, with a total of 56880 samples. Among them, 40 types are daily actions; 9 types are health-related actions and 11 types are interactive actions. The movements were performed by 40 people aged from 10 to 35. The dataset is collected by Microsoft Kinect V2 sensors. Three cameras with different angles are used. The collected data form includes depth information, 3D skeleton information, RGB frames and infrared sequences. The 3D skeleton data we used includes the 3D coordinates of 25 human joints in each frame. There are two evaluation protocols: cross-subject (xsub) and cross-view (xview). For the xview experiment, the training and test datasets are divided by cameras from different views. The 18960 samples collected by camera 1 are used as the test dataset, and the remaining samples are used as the training dataset. For the xsub experiment, we divide the samples into a training dataset and a test dataset according to the person ID. There are 40320 samples in the training dataset with IDs of 1, 2, 4, 5, 8, 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31, 34, 35 and 38. The rest dataset is used as the test dataset with 37920 samples. In the following experiment, we abbreviated NTU RGB+D 60 Dataset, the xivew and the xsub evaluation protocols to NTU-60 Dataset, NTU-60 xview and NTU-60 xsub, respectively. In our work, we split the datasets both in xview and in xsub experiments to 50 frames.

Northwestern-UCLA Dataset. The dataset includes 10 kinds of actions with 1494 video clips, which are captured by three Kinect cameras. Each action is performed by 10 different subjects. We use the video samples from the first two cameras as training datasets and the rest are test datasets, referring to [36].

B. Experimental Settings

All our experiments are conducted on the server with the Tesla-V100 GPU. The deep learning framework in our work is Pytorch. We preprocess the original data by eliminating the missing data and reordering the rest. In our experiments, the temporal frames are all resized to 50 frames. Hyperparameters are set as follows: the batch size is set to 128; the size of each

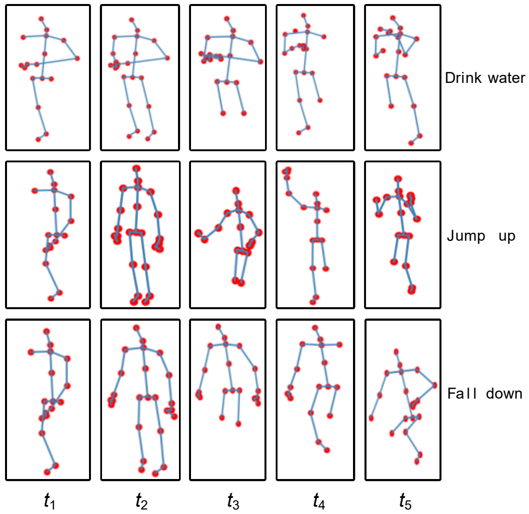


Fig. 4. Visualization of spatiotemporal occlusion mask of skeleton data on three actions: drink water, jump up and fall down. t_1, t_2, t_3, t_4, t_5 represent different temporal frames. The occluded body parts of each frame are random.

memory bank M is set to 16 K and the momentum is set to 0.999.

Data Augmentation T. Instead of traditional data augmentation such as shear and crop, we conduct the spatiotemporal occlusion mask for input skeleton data and study the effect of different spatiotemporal mask rates for data augmentation.

Self-supervised Pre-training. We follow the experiment in 3s-CrosSCLR [31]. For data augmentation, we respectively set the percent of spatial occlusion mask and temporal frames occlusion mask to 0.6 and 0.5. The model is trained for 300 epochs with the learning rate 1e-5. Specially, we cross-train our model 150 epochs.

Linear Evaluation Protocol. A linear classifier is used for the action recognition task. We freeze two encoders to prevent them from gradient descent and then train the linear classifier (a fully-connected layer followed by a softmax layer) with a supervised training mode. The training lasts for 100 epochs with a learning rate 0.1.

Semi-supervised Evaluation Protocol. We pre-train the encoder with all data and then fine-tune the whole model with only 1% or 10% randomly selected labeled data.

Fine-tuned Evaluation Protocol. We add a linear classifier to the two encoders and train them as a whole for gradient descent. We train for 100 epochs with a learning rate 1e-4 and compare it with fully-supervised methods.

C. Ablation Study

All experiments in this section are conducted on PKU MMD Part II dataset with self-supervised pre-training. We pre-train 300 epochs with Tesla-V100.

Data Augmentation. In this section, we compare the effects of spatiotemporal mask rate on data augmentation by conducting linear evaluation experiments. The results are shown in Table I. It can be seen that when the rates of temporal frame occlusion mask and skeleton joint spatial occlusion mask are respectively 0.5 and 0.6, CrossMoCo can reach the highest accuracy on the PKU MMD part II dataset. In subsequent experiments on this dataset, we choose the set of spatiotemporal mask parameters (0.5, 0.6) as hyperparameters. The accuracy curve of linear evaluation with the spatiotemporal mask occlusion is shown as Fig.5. The linear evaluation accuracy of the spatiotemporal mask is the highest. The single spatial mask performs better than the single temporal mask. It means that the positive samples' qualities from the spatial skeleton mask are higher than those of the temporal mask. The mask unit in our proposed spatial mask is the adjacent skeleton joints that can form a skeleton bone in the human skeleton topology. When a skeleton joint is occluded, the adjacent joints' information will also be cleared, preventing the network from obtaining spatiotemporal information related to this occluded skeleton joint from adjacent skeleton joints. It effectively reduces the redundant information among positive samples after data augmentation. The diverse positive samples with independent and multi-level feature information will significantly improve our model's contrastive learning ability.

TABLE I
THE TEST ACCURACY (%) OF DIFFERENT SPATIOTEMPORAL OCCLUSION MASK RATES ON THE PKU MMD PART II DATASET

Augmentation		PKU MMD Part II (%)
Temporal Occlusion Mask Rate	Spatial Occlusion Mask Rate	
0.2	0.2	15.8
0.4	0.4	22.9
0.6	0.6	30.4
0.5	0.6	29.8
0.8	0.6	23.2

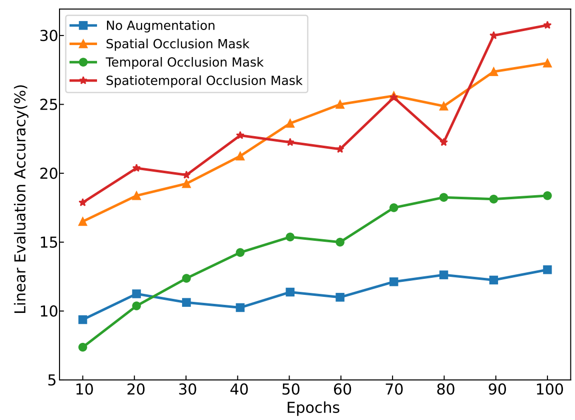


Fig. 5. The accuracy curve of spatiotemporal occlusion mask with linear evaluation. The test accuracy with the spatiotemporal mask is the highest.

Impact of Encoders On CrossMoCo. In this part, we explore the effectiveness of ST-GCN encoder, ST-MLP encoder, uncrossed ST-GCN encoder with ST-MLP encoder whose query-key pairs are multiplied independently, and crossed ST-GCN encoder with ST-MLP encoder whose query-key pairs are cross-multiplied. Results are shown in Table II.

Crossed encoder reaches the highest accuracy and increases the 8.2 % and 4.5 % than the uncrossed encoder on PKU MMD Part II in two experiments, respectively. The crossed encoder

TABLE II
TEST ACCURACY (%) OF CROSSMOCO'S ENCODERS ON THE PKU MMD PART II DATASET IN TERMS OF THE LE AND FE TESTS

Encoder	PKU MMD Part II (%)	
	L E	F E
ST-GCN	30.0	49.8
ST-MLP	26.6	45.7
Uncrossed ST-GCN+ST-MLP	28.1	50.9
Crossed ST-GCN+ST-MLP	30.4	57.3

TABLE III
TEST ACCURACY (%) OF CROSSMOCO'S MEMORY BANKS ON THE PKU MMD PART II DATASET IN TERMS OF THE LE AND FE TESTS

Numbers of memory banks	PKU MMD Part II (%)	
	L E	F E
1	29.6	47.4
2	30.4	57.3
3	27.3	50.1

TABLE IV
TEST ACCURACY (%) OF MLP LAYERS IN ST-MLP ON THE PKU MMD PART II DATASET IN TERMS OF THE LE AND FE TESTS

MLP Layers in ST-MLP	PKU MMD Part II (%)	
	L E	F E
1	30.4	57.3
2	23.8	49.4
3	29.8	49.7

TABLE V
COMPARISON WITH DIFFERENT MODELS ON THE NTU-60 DATASET AND THE NW-UCLA DATASET IN THE TERMS OF LINEAR EVALUATION TEST ACCURACY (%)

Methods	NTU-60 (%)		NW-UCLA (%)
	xview	xsub	
LongT GAN [34]	48.1	39.1	74.3
P&C [28]	76.3	50.7	71.4
AS-CAL [13]	64.8	58.5	75.6
3s-SkeletonCLR [31]	79.8	75.0	70.4
3s-CrosSCLR [31]	83.4	77.8	83.6
3s-AimCLR [37]	83.8	78.9	–
Auto-encoder [38]	70.3	78.3	87.4
VEJP+VPE [39]	54.9	51.4	85.4
4s-MG-AL [40]	64.7	68.0	81.1
CrossMoCo (ours)	84.9	78.4	87.6

combines two encoders' advantages by crosswise learning the query-key pairs' features generated by ST-GCN and ST-MLP. ST-MLP avoids network dependence on long frames via global encoding. ST-GCN captures fine-grained action information features via local encoding. The feature representations generated by ST-GCN and ST-MLP in the uncrossed encoder are only simple addition, needing more information fusion. The feature representations learned by the uncrossed encoder may be inconsistent, and the positive sample features generated by another encoder will be regarded as negative sample features when the pre-training model is frozen in the LE (Linear Evaluation) test, leading to confusion in classification and low performance. However, in the FE (Fine-tuned Evaluation) test, the fine-tuning of the pre-training model under the guidance of the label makes the feature representations unified, thus improving the accuracy. For the crossed encoder, the feature representations generated by ST-GCN and ST-MLP interact to realize the fusion of local and global information. Even if the pre-training model is not fine-tuned, the feature representations extracted from the whole model are quite uniform. Therefore, the crossed encoder has a good performance in the LE test. The experiment shows that the simple combination of multiple encoders cannot make the model perform better than the single encoder. Cross-learning feature representations can integrate the advantages of different base encoders to make the network perform well on public datasets. It provides a solution for the future representations fusion of various encoders in the field of contrastive learning.

In general, the accuracy's gains in the LE test are lower than those in the FE test. In the LE test, only the full connection layer is updated according to the labels' information. The pre-training model is frozen and cannot be updated in the downstream task, limiting the gains in the downstream task. However, the whole model is fine-tuned to complete the downstream task in the FE test. The fine-tuning of the pre-training model makes features extracted by the whole model more precise and accurate than those of the frozen pre-training model in the downstream task so that the accuracy's gains are greater than those in the LE test.

Impact of the Memory Banks' Numbers. In this section, we explore the effectiveness of two memory banks of the CrossMoCo model. In this experiment, the encoders are set to crossed ST-GCN encoder with ST-MLP encoder. Results are shown in Table III.

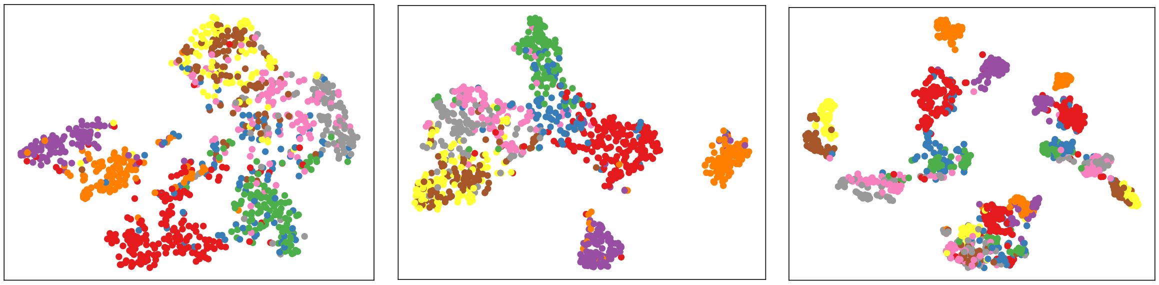
The accuracy is the highest when we adopt two independent negative memory banks. However, when there are more memory banks, the number of negative sample pairs will be much larger than that of positive sample pairs, which is easy for the model to take a shortcut in representation learning. If there is only one memory bank, the negative samples' feature updating in the memory bank cannot be well consistent with the positive key representations embedded by the two encoders. It will reduce the difficulty for our network to discriminate the positive and negative representations so that the ability of the network's contrastive learning cannot be well improved.

Impact of MLP with Different Layers. In this section, we explore the influence of the number of MLP layers of ST-MLP on CrossMoCo. The experimental results are shown in Table IV. It can be seen that when the MLP layer is one, the test accuracy is the highest on PKU MMD Part II. It shows that one layer of MLP is sufficient for data fitting. If the MLP layer's number is more than the appropriate number of fitting layers, it will lead to the difficulty of training and increase the higher training error, which will make the network fall into the local optimization and lead to network degradation.

D. Comparison With the State-Of-The-Art

In this section, we compare CrossMoCo with the state-of-the-art contrastive learning models with different evaluation protocols.

Linear Evaluation Results on NTU-60 and NW-UCLA Datasets. Table V shows that our CrossMoCo performs best on the NTU-60 xview dataset and the NW-UCLA dataset.

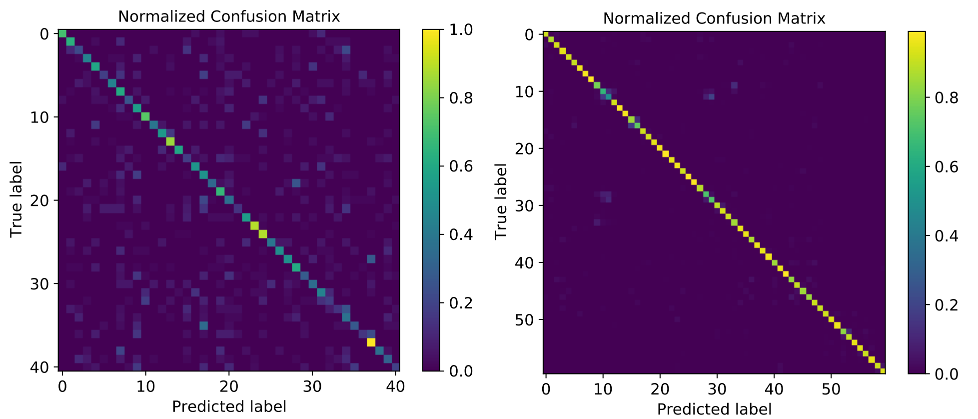


(a) (b) (c)

Fig. 6. The t-SNE visualization of feature embeddings' distributions from three algorithms on the NW-UCLA dataset. Different colors represent different action categories. The same classes are expected to be grouped together and different classes are expected to be far way. (a) 3s-SkeletonCLR. (b) 3s-CrosSCLR. (c) CrossMoCo (ours).

TABLE VI
SEMI-SUPERVISED EVALUATION ACCURACY (%) ON THE PKU MMD PART II DATASET , THE NTU-60 XVIEW DATASET AND THE NW-UCLA DATASET

Methods	Label Fraction	Accuracy (%)		
		PKU MMD Part II	NTU-60 xview	NW-UCLA
LongT GAN [34]	1%	12.4	–	18.2
MS^{2}L [10]	1%	–	–	21.9
ISC[55]	1%	–	38.1	–
3s-CrosSCLR [31]	1%	10.2	50.0	29.9
CrossMoCo (ours)	1%	16.4	21.7	31.4
LongT GAN [34]	10%	25.8	–	59.9
MS^{2}L [10]	10%	26.1	–	60.5
ISC[55]	10%	–	72.5	–
3s-CrosSCLR [31]	10%	21.1	77.8	66.5
CrossMoCo (ours)	10%	26.7	68.7	69.7



(a) (b)

Fig. 7. The confusion matrix of proposed CrossMoCo on two large public datasets. (a) PKU MMD Part II dataset. (b) NTU-60 xiew dataset.

Specially, it exceeds the baseline algorithm 3s-CrosSCLR by 1.8 % and 4.8 % on NTU-60 xivew and NW-UCLA datasets, respectively. In addition, t-SNE [41] is used to show the distributions of feature embeddings from 3s-SkeletonCLR, 3s-CrosSCLR and CrossMoCo (ours) algorithms on the NW-UCLA dataset shown in Fig.6. Different colors represent action categories. The same action classes are expected to converge, and different action classes are expected to separate. It can be seen intuitively that CrossMoCo clusters better than that of 3s-CrosSCLR. The same categories are more closed and different categories are farther apart.

Semi-supervised Evaluation Results on Three Datasets. We compare our model with other excellent algorithms under semi-supervised evaluation with a small number of labels on the PKU MMD Part II dataset, the NTU-60 xview dataset, and the NW-UCLA dataset, shown in Table VI. We test the effect of different algorithms on the linear evaluation of three datasets with 1 % and 10 % labels. When the label fractions are respectively 1 % and 10 %, the test results on PKU MMD Part II are 60.8 % and 26.5 % higher than that of 3s-CrosSCLR. Specially, CrossMoCo has achieved the best results both on the PKU MMD Part II and the NW-UCLA datasets.

Fine-tuned Evaluation Results. We compare the fine-tuned evaluation of different algorithms on the PKU MMD Part II dataset and the NTU-60 dataset, as shown in Table VII. Our algorithm has outperformed 9.1 % of the baseline 3s-

TABLE VII
FINE-TUNED EVALUATION ACCURACY (%) WITH DIFFERENT ALGORITHMS ON THE PKU MMD PART II DATASET AND THE NTU-60 DATASET

Methods	PKU MMD Part II (%)	NTU-60 (%)	
		xview	xsub
3s-ST-GCN [6]	26.1	91.4	85.2
SkeletonCLR [31]	37.7	88.9	82.2
3s-CrosSCLR [31]	52.5	92.5	86.2
3s-AimCLR [37]	–	92.8	86.9
Auto-encoder [38]	–	86.5	76.5
CrossMoCo (ours)	57.3	93.1	87.2

CrosSCLR on the PKU MMD Part II dataset and is superior to other algorithms on the NTU-60 dataset. Fig.7 shows the confusion matrix of the proposed CrossMoCo on PKU MMD Part II and the NTU-60 xiew. As shown in the confusion matrix, most of the actions are predicted by our model.

V. CONCLUSION

In our work, we propose a new contrastive learning framework CrossMoCo for self-supervised 3D human skeleton action recognition. It encodes positive input data via two encoders, the base encoder ST-GCN and the ST-MLP encoder by adding an MLP project head to the top of the ST-GCN. Two kinds of semantic query-key positive feature representations embedded by the encoders are cross-multiplied to learn local and global semantic representations, improving representation learning ability. Inspired by MoCo, we establish two independent negative memory banks to provide high-quality negative samples that have consistent representations with the positive embeddings from the two encoders. The similarity of positive and negative representations increases the difficulty of discrimination, promoting the model's contrastive learning. Besides, we invent the spatiotemporal occlusion mask data augmentation method to generate positive samples without redundant information. Experiments on the PKU-MMD Part II dataset, the NTU RGB+D 60 dataset, and the NW-UCLA dataset show that our CrossMoCo has achieved a comparable result.

In the future, more downstream tasks, such as 3D action retrieval and prediction, will be completed. Moreover, the labels' language semantic information will also be used as the input to guide our model to learn representations and realize zero-shot action recognition.

REFERENCES

[1] M. Seo, D. Cho, Lee *et al.*, "A self-supervised sampler for efficient action recognition: Real-world applications in surveillance systems," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1752–1759, 2021.

[2] H. D. Mehr and H. Polat, "Human activity recognition in smart home with deep learning approach," in *2019 7th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)*. IEEE, 2019, pp. 149–153.

[3] D. Nalci and Y. S. Akgul, "Human action recognition with raw millimeter wave radar data," in *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. IEEE, 2022, pp. 1–5.

[4] X. Li, C. Li, Wei *et al.*, "Manifold guided graph neural networks for skeleton-based action recognition in human computer interaction videos," in *2021 International Conference on Signal Processing and Machine Learning (CONF-SPML)*. IEEE, 2021, pp. 239–244.

[5] Y. Ji, Y. Yang, Shen *et al.*, "A survey of human action analysis in hri applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 7, pp. 2114–2128, 2019.

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

[7] M. Li, S. Chen, and a. o. Chen, "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.

[8] L. Shi, Y. Zhang, Cheng *et al.*, "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.

[9] Z. Liu, H. Zhang, Chen *et al.*, "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.

[10] L. Lin, S. Song, Yang *et al.*, "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.

[11] T. Chen, S. Kornblith, and a. o. Norouzi, "A simple framework for contrastive learning of visual representations," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 1597–1607.

[12] K. He, H. Fan, Wu *et al.*, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.

[13] H. Rao, S. Xu, Hu *et al.*, "Augmented skeleton based contrastive action learning with momentum lstm for unsupervised action recognition," *Information Sciences*, vol. 569, pp. 90–109, 2021.

[14] J. Liu, S. Song, Liu *et al.*, "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.

[15] A. Shahroudy, J. Liu, Ng *et al.*, "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.

[16] J. Wang, X. Nie, Xia *et al.*, "Cross-view action modeling, learning, and recognition," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2649–2656.

[17] M. E. Hussein, M. Torki, Gowayyed *et al.*, "Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations," in *Twenty-third international joint conference on artificial intelligence*, 2013.

[18] L. Xia, C.-C. Chen, Aggarwal *et al.*, "View invariant human action recognition using histograms of 3d joints," in *2012 IEEE computer society conference on computer vision and pattern recognition workshops*. IEEE, 2012, pp. 20–27.

[19] E. Ohn-Bar and M. Trivedi, "Joint angles similarities and hog2 for action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2013, pp. 465–470.

[20] Y. Lu, Y. Shi, Jia *et al.*, "A new method for semantic consistency verification of aviation radiotelephony communication based on lstm-rnn," in *2016 IEEE International Conference on Digital Signal Processing (DSP)*. IEEE, 2016, pp. 422–426.

[21] H. Wang, B. Yu, Li *et al.*, "Multi-stream interaction networks for human action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 3050–3060, 2022.

[22] P. Zhang, C. Lan, Xing *et al.*, "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.

[23] C.-F. R. Chen, R. Panda, Ramakrishnan *et al.*, "Deep analysis of cnn-based spatio-temporal representations for action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6165–6175.

[24] H. Wu, X. Ma, and a. o. Li, "Spatiotemporal multimodal learning with 3d cnns for video action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1250–1261, 2021.

[25] K. Zhu, R. Wang, and a. o. Zhao, "A cuboid cnn model with an attention mechanism for skeleton-based action recognition," *IEEE Transactions on Multimedia*, vol. 22, no. 11, pp. 2977–2989, 2020.

[26] N. Heidari and A. Iosifidis, "Temporal attention-augmented graph convolutional network for efficient skeleton-based human action recognition,"

in *2020 25th international conference on pattern recognition (ICPR)*. IEEE, 2021, pp. 7907–7914.

[27] L. Shi, Y. Zhang, Cheng *et al.*, “Skeleton-based action recognition with multi-stream adaptive graph convolutional networks,” *IEEE Transactions on Image Processing*, vol. 29, pp. 9532–9545, 2020.

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

[29] X. Gao, Y. Yang, Zhang *et al.*, “Efficient spatio-temporal contrastive learning for skeleton-based 3d action recognition,” *IEEE Transactions on Multimedia*, pp. 1–1, 2021.

[30] T. Chen, S. Kornblith, Norouzi *et al.*, “A simple framework for contrastive learning of visual representations,” in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[31] L. Li, M. Wang, Ni *et al.*, “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.

[32] F. M. Thoker, H. Doughty, Snoek *et al.*, “Skeleton-contrastive 3d action representation learning,” in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 1655–1663.

[33] T. Pan, Y. Song, Yang *et al.*, “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.

[34] N. Zheng, J. Wen, Liu *et al.*, “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.

[35] C. Shorten and T. M. Khoshgoftaar, “A survey on image data augmentation for deep learning,” *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[36] C. Wei, L. Xie, Ren *et al.*, “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.

[37] T. Guo, H. Liu, Chen *et al.*, “Contrastive learning from extremely augmented skeleton sequences for self-supervised action recognition,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, 2022, pp. 762–770.

[38] J. Zhou and T. Komuro, “An asymmetrical-structure auto-encoder for unsupervised representation learning of skeleton sequences,” *Computer Vision and Image Understanding*, vol. 222, p. 103491, 2022.

[39] W. You and X. Wang, “View enhanced jigsaw puzzle for self-supervised feature learning in 3d human action recognition,” *IEEE Access*, vol. 10, pp. 36 385–36 396, 2022.

[40] 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*, vol. 32, no. 12, pp. 8623–8634, 2022.

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

Qinyang Zeng received her M.S. degree in mechanical engineering from the School of Mechatronics Engineering, Harbin Institute of Technology, Harbin, in 2018, and B.S. degree in Vehicle Application Engineering, Harbin Institute of Technology, Weihai, in 2016. She is a Ph.D. candidate in College of Electronics and Information Engineering, Tongji University, Shanghai. Her current research interests include action recognition, graph neural network and deep learning.

Chengju Liu received the Ph.D. degree in control theory and control engineering from Tongji University, Shanghai, China, in 2011. From October 2011 to July 2012, she was with the BEACON Center, Michigan State University, East Lansing, MI, USA, as a Research Associate. From March 2011 to June 2013, she was a Postdoctoral Researcher with Tongji University, where she is currently a Professor with the Department of Control Science and Engineering, College of Electronics and Information Engineering, and a Chair Professor of Tongji Artificial Intelligence (Suzhou) Research Institute. She is also a Team Leader with the TJArk Robot Team, Tongji University. Her research interests include intelligent control, motion control of legged robots, and evolutionary computation.

Ming Liu (Senior Member, IEEE) received the B.A. degree in automation from Tongji University, Shanghai, China, in 2005, and the Ph.D. degree from the Department of Mechanical Engineering and Process Engineering, ETH Zürich, Zürich, Switzerland, in 2013. He was a Visiting Scholar with the University of Erlangen-Nuremberg, Erlangen, Germany, and the Fraunhofer Institute of Integrated Systems and Device Technology (IISB), Erlangen. He is currently an Assistant Professor with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong. His current research interests include autonomous mapping, visual navigation, topological mapping, and environment modeling.

Qijun Chen (Senior Member, IEEE) received the B.S. degree in automatic control from Huazhong University of Science and Technology, Wuhan, China, in 1987, the M.S. degree in control engineering from Xi’an Jiaotong University, Xi’an, China, in 1990, and the Ph.D. degree in control theory and control engineering from Tongji University, Shanghai, China, in 1999. He was a Visiting Professor with the University of California at Berkeley, Berkeley, CA, USA, in 2008. He is currently a Professor with the Department of Control Science and Engineering, College of Electronics and Information Engineering, Tongji University. His current research interests include network-based control systems and robotics.