Self-Supervised Skeleton Representation Learning Via Actionlet Contrast and Reconstruct

Lilang Lin, *Graduate Student Member, IEEE*, Jiahang Zhang, and Jiaying Liu, *Fellow, IEEE*

***Abstract*—Contrastive learning has shown remarkable success in the domain of skeleton-based action recognition. However, the design of data transformations, which is crucial for effective contrastive learning, remains a challenging aspect in the context of skeleton-based action recognition. The difficulty lies in creating data transformations that capture rich motion patterns while ensuring that the transformed data retains the same semantic information. To tackle this challenge, we introduce an innovative framework called ActCLR+ (Actionlet-Dependent Contrastive Learning), which explicitly distinguishes between static and dynamic regions in a skeleton sequence. We begin by introducing the concept of *actionlet*, connecting self-supervised learning quantitatively with downstream tasks. Actionlets represent regions in the skeleton where features closely align with action prototypes, highlighting dynamic sequences as distinct from static ones. We propose an anchor-based method for unsupervised actionlet discovery, establishing a motion-adaptive data transformation approach based on this discovery. This motion-adaptive data transformation strategy tailors data transformations for actionlet and non-actionlet regions, respectively, introducing more diverse motion patterns while preserving the original motion semantics. Additionally, we incorporate a semantic-aware masked motion modeling technique to enhance the learning of actionlet representations. Our comprehensive experiments on well-established benchmark datasets such as NTU RGB+D and PKUMMD validate the effectiveness of our proposed method.**

***Index Terms*—Skeleton-based action recognition, self-supervised learning, contrastive representation learning.**

I. Introduction

SKELETONS depict human joints through 3D coordinate locations and offer a lightweight and compact means of representing human motion in contrast to RGB videos and depth data. Due to ease of use and improved discriminative capabilities for analysis, skeletons have found extensive application in action recognition tasks [1], [2], [3], [4], [5], [6].

Supervised skeleton-based action recognition methods [7], [8], [9] have demonstrated remarkable performance, but heavily rely on vast labeled training data, of which collection can be costly work. To mitigate dependence on complete supervision, self-supervised learning has been studied for skeleton-based action recognition [6], [10], [11], [12].

When considering pre-training paradigms, most methods fall into two categories: reconstruction-based [6], [13], [14] and contrastive learning-based. Reconstruction-based approaches model the spatial-temporal correlations by forecasting masked skeleton data. In the domain of long-term global motion dynamics, Zheng et al. [10] pioneered the concept of reconstructing masked skeletons. On the other hand, contrastive learning-based methods have recently exhibited remarkable potential, which utilize skeleton transformations to generate positive pairs, and to seek consistency in the embedding space. Rao et al. [15] introduced shearing and cropping as data augmentation techniques. Guo et al. [16] extended these efforts by suggesting additional augmentations, such as rotation, masking, and flipping, to enhance the consistency of contrastive learning.

Although previous works [17], [18], [19], [20], [21] have demonstrated the importance of data transformations, they less comprehensively study the impact of data transformations on feature space. Previous works have typically approached the analysis from the perspectives of mutual information, loss function, or graph optimization, respectively. However, there are relatively few theoretical studies addressing data transformation in contrastive learning. And data transformation is crucial for the performance of contrastive learning. Therefore, in this paper, we show that contrastive learning loss is equivalent to performing spectral clustering in data augmentation graphs. We focus on the role of data transformation in contrastive learning. We show that the strength of the data transformation determines the number of clusters in graph clustering, while the method of data transformation determines the purity of clustering. Our results show that good downstream task performance is achieved by reducing the number of clusters and improving clustering purity. However, we also note that it is difficult to optimize both the quantity and quality of clustering. When the data transformation is strong, the number of clusters is small, but also it is more likely that there are different classes of data clustered into the same clusters. This leads to a decrease in the purity of the clustering.

In response to these challenges, we introduce a novel enhanced actionlet-dependent contrasting and reconstructing learning method (ActCLR+), wherein we treat motion and static regions distinctly. We ensure that the clusters have a good purity by utilizing data transformation that can preserve the semantics, so that data of the same category can be aggregated into the same clusters.

Received 2 April 2024; revised 9 March 2025; accepted 26 July 2025. Date of publication 13 August 2025; date of current version 3 October 2025. This work was supported in part by the Program of Beijing Municipal Science and Technology Commission Foundation under Grant Z241100003524010 and in part by the National Natural Science Foundation of China under Grant 62172020. Recommended for acceptance by L. Wang. *(Corresponding author: Jiaying Liu.)*

The authors are with the Wangxuan Institute of Computer Technology, Peking University, Beijing 100080, China (e-mail: linlilang@pku.edu.cn; zjh2020@pku.edu.cn; liujiaying@pku.edu.cn).

Digital Object Identifier 10.1109/TPAMI.2025.3598138

0162-8828 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

First, we introduce the concept, *actionlet*, which is defined in the work [22] as the highly representative skeleton structure, to the self-supervised learning context. These actionlets contain strong discriminative action patterns for distinguishing the corresponding action from others, and hence can guide the contrastive learning process. Inspired by this, we propose an unsupervised method to mine the actionlets by contrasting action prototypes as positive anchors and mean motions as negative anchors. Specifically, action prototypes represent cluster centers obtained through feature clustering, while the mean motion is calculated based on the average of all sequences in the dataset. Action prototypes can thus be considered a series of actions, while mean motion acts as a stationary anchor devoid of motion. Subsequently, we compare action sequences to the nearest action prototypes and mean motions to identify the region with the most representative unique patterns, which is treated as the *actionlets*.

Based on the concept of actionlets, we further develop a motion-adaptive similarity distillation module (MASD) to by reduce the number of clusters and improve clustering purity. This module involves applying the proposed semantically preserving data transformations to the actionlet region, while utilizing stronger data transformations only for non-actionlet regions. This approach preserves action movement within actionlet while incorporating richer motion patterns, resulting in more compact and informative learned features. This leads to a decrease in the number of clusters in spectral clustering while upholding a high cluster purity. We then apply an intra-sequence and inter-sequence similarity distillation loss to constrain the feature consistency of the transformed data with the original data.

Additionally, we employ a semantic-aware masked motion modeling method (SAM^3). This method focuses on reconstructing data in the actionlets region, emphasizing features related to motion information and enhancing performance in downstream tasks.

We conduct extensive experiments on NTU RGB+D [23], [24] and PKUMMD [25] datasets. Our model attains outstanding results through self-supervised learning in comparison to contemporary sota methods.

Our contributions are summarized as follows:

- We prove that contrastive learning is equivalent to spectral clustering on the data augmentation graph. The data transformation determines the number and purity of clusters. Fewer number of clusters and better purity of clusters lead to good downstream task performance. We propose a semantics-preserving data transformation to obtain optimal clustering embeddings, with a lower number of clusters and higher purity.
- We design an actionlet-based motion adaptive data transform to preserve motion information. After mining the motion regions as actionlets through an unsupervised approach, we enhance the data transformation of the non-actionlet regions to reduce the number of clusters and improve clustering purity. Meanwhile, we employ an intra-data similarity distillation to enhance the consistency along with an inter-data contrasting to provide richer motion patterns.
- Through the analysis we find that overfitting of contrastive learning is caused by that the learned features contain only mutual information between positive samples. Therefore, we propose semantic-aware masked motion modelling to increase task-relevant information. By reconstructing the data in the actionlets region, the features are more concerned with motion information and have better performance in downstream tasks.

This paper is an extension of our earlier publication [26]. Compared to our previous work, we make significant additional contributions in both theory, technological, and experimental parts. (1) This study provides novel theoretical derivations on contrastive learning, demonstrating its equivalence to spectral clustering in data augmentation graphs. This newly theoretical conclusion leads to the comprehensive upgrade of our actionlet-based method. (2) Beyond using the average motion to select actionlets in [26], action prototypes obtained based on cluster characteristics are used to select the motion regions. Additionally, we are further inspired to augment motion-aware data transformation with newly proposed strong data transformations based on adversarial noise and skeleton masking. (3) Furthermore, at the loss end, to model the inter-data relationship and capture fine-grained motion details of skeletons, we propose two loss terms built on mix-based inter-sequence similarity distillation and mask reconstruction. (4) The experiments have been significantly enriched, providing more comprehensive comparison results from diverse aspects. Impressively, we achieve a 7.5% increase in xview and a 5.7% increase on xsub for the NTU 60 dataset with only 1% training samples.

The remainder of this paper is organized as follows. In Section II, we provide an overview of existing research in relevant fields. Then, in Sections III and IV, we delve into the motivation behind our approach and its intricate design details, respectively. Next, in Section V, we present experimental results to showcase the efficacy of our methods. Finally, in Section VI, we conclude the paper with a summary and outline potential future directions.

II. Related Work

A. Skeleton-Based Action Recognition

Sequences of skeletal data provide a detailed representation of the movements of human joints. As a result, skeletons are appropriate modality for recognizing actions. This type of data is obtained by applying pose estimation algorithms to RGB videos and depth maps [27]. Owing to its scale invariance, and resilience to variations in clothing texture and background, the field of skeleton-based human action recognition has garnered considerable interest within the research community [2], [3].

Initial investigations concentrated on deriving manually crafted spatial and temporal domain characteristics of skeletal sequences to recognize human movements [28], [29]. In later work, Tao et al. [28], [30] focused on capturing both positional data and higher-order temporal variations within the human

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

skeleton, while Wang et al. [31] constructed a graphical model that tracked the trajectory of human joints to represent the joint information within the video sequence.

Given the potent expressiveness of graph structures, there has been a burgeoning interest in the recent research in deploying graph-based learning models [31]. Graph Neural Networks (GNNs) are connectivity frameworks that adeptly capture the dependencies within a graph, allowing for the transfer of information between nodes. Si et al. [32] were pioneers in introducing a network that deduces spatial domain inferences and proficiently captures the high-level spatial structure and temporal dynamics present in skeletal data. The introduction of high-level joint semantics for human movement recognition was explored by Zhang et al. [3]. Moreover, the application of attention mechanisms to extract discriminative information and global dependencies was investigated by Si et al. [7]. To alleviate the computational demands of GCN, Song et al. [33] devised a multi-stream GCN model that early integrates various input branches such as joint position, motion velocity, and bone features, employing separable convolutional layers to significantly cut down on trainable parameters. Shi et al. [34] introduced the 3D-Shift GCN, a novel architecture leveraging a spatiotemporal volume framework to model cross-joint dependencies and interactions for global feature extraction. Zhou et al. [35] proposed a novel topological encoding approach that captures the skeletal structure by encoding the relative distances between joint pairs within the skeletal graph for effectively preserving the spatial relationships and hierarchical dependencies inherent in the skeletons.

However, the receptive fields of Graph Convolutional Networks (GCNs) are inherently constrained by joint connectivity, limiting their ability to capture global dependencies. To overcome this limitation, recent advancements [36], [37] have introduced transformer-based methods. Jeonghyeok et al. [37] proposed Skeletal-Temporal Transformer (SkateFormer), that partitions joints and frames based on distinct skeletal-temporal relationships and applies skeletal-temporal self-attention within each partition, enabling efficient and focused modeling of skeletal-temporal dynamics.

B. Self-Supervised Representation Learning

Contrastive learning offers a self-supervised approach to align pairs of positive sample inputs while pushing pairs of negative sample inputs further apart. Typically, these methods aim to minimize the distance of representations between positive samples and maximize that between negative samples. In the realm of images, positive samples are often created through various augmentation such as rotation, cropping, and random adjustments in grayscale and color [38], [39], [40], [41], [42]. Translating these techniques to videos poses challenges, particularly in considering the temporal domain. While some methods straightforwardly extend the same augmentation applied to images to each frame in the video [43], [44], [45], others integrate additional frame alignments based on temporal considerations [46], [47]. Additionally, certain approaches rely on motion and optical flow maps as positive samples [48].

For skeleton data, it is difficult to directly introduce data transformation methods from the image or video domains because of the sparsity in the spatial domain and the redundant low-rank nature of skeleton data in the temporal domain. Many methods have tried to design corresponding data transformations for skeletons [11], [15], [16], [26], [49], [50]. Lin et al. [11] pioneered the integration of contrastive learning into skeleton action recognition, introducing data transformations based on masking and temporal-domain shuffling. iMiGUE [51] specializes in body movement analysis through an encoder-decoder architecture to extract discriminative features from keypoint-based motion sequences in an unsupervised manner. Rao et al. [15] devised more data transformations and demonstrated that strong data transformations produce better recognition performance. Mao et al. [49] subsequently introduced a relational distillation loss to address the pseudo-negative sample problem in contrastive learning. This innovation aimed to improve the consistency and quality of learned representations. Lin et al. [26] further refined the data transformation method for skeleton data by introducing the decomposition of moving and non-moving regions. This decomposition enabled the design of a data transformation that preserves the semantic information of motion. Shah et al. [52] explored the challenge of hallucinating new positives within the latent space. Building on this, Lin et al. [53] introduced equivariant contrastive learning, an extension of invariant contrastive learning, designed to retain critical transformation information while enhancing representational robustness.

Recent studies have integrated generative pre-training into unsupervised representation learning, yielding promising advancements. Mao et al. [54] proposed predicting the temporal motion of masked human joints within spatio-temporal skeleton sequences, effectively leveraging generative models for temporal understanding. Lin et al. [55] provided a theoretical demonstration of the equivalence between generative models and maximum entropy coding. They introduced an idempotency constraint that enforces stronger consistency regularization in the feature space. Abdelfattah et al. [56] predicted the latent representations of missing joints within the same sequence to learn the high-level context and depth information.

III. A Graph Learning Perspective

A. Spectral Clustering on Augmentation Graph

In this section, we show that contrastive learning loss is equivalent to performing spectral clustering in data augmentation graphs. From an energy modelling perspective, contrastive learning loss functions constitute an estimate of the distribution.

$$\begin{aligned} p(\mathbf{x}) &= \sum_{\mathbf{x}^+} p(\mathbf{x}, \mathbf{x}^+) = \sum_{\mathbf{x}^+} p(\mathbf{x}|\mathbf{x}^+) p(\mathbf{x}^+) \\ &= \sum_{\mathbf{x}^+} \mathcal{A}(\mathbf{x}|\mathbf{x}^+) p(\mathbf{x}^+) \\ &= \frac{1}{|\mathcal{X}|} \sum_{\mathbf{x}^+ \in \mathcal{X}} \frac{1}{Z} \exp\left(f(\mathbf{x})^T f(\mathbf{x}^+) \right), \end{aligned} \tag{1}$$

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

where Z is the normalization factor. where $\mathbf{x}^+$ is the transformed views of input skeleton data $\mathbf{x}$ with data transformation $\mathcal{T}$. $f(\cdot)$ projects $\mathbf{x}$ into the hypersphere $\mathbb{S}^{d-1}$, where d is the dimension size. We take InfoNCE ($\mathcal{L}_{\text{simclr}}$) [57] as an example, which is commonly used in contrastive learning.

$$\begin{aligned}\mathcal{L}_{\text{simclr}} &= -\sum_{\mathbf{x}} \log p(\mathbf{x}) = -\sum_{\mathbf{x}} \log \frac{1}{Z} \exp\left(f(\mathbf{x})^T f(\mathbf{x}^+)\right) \\ &= -\sum_{\mathbf{x},\mathbf{x}^+} \left[f(\mathbf{x})^T f(\mathbf{x}^+)\right] \\ &\quad + \sum_{\mathbf{x}} \left[\log \sum_{\mathbf{x}^-} \left[\exp\left(f(\mathbf{x})^T f(\mathbf{x}^-)\right)\right]\right] \\ &= -\sum_{\mathbf{x},\mathbf{x}'\in\mathcal{X}} \omega_{\mathbf{x},\mathbf{x}'} f(\mathbf{x})^T f(\mathbf{x}') \\ &\quad + \sum_{\mathbf{x}\in\mathcal{X}} \omega_{\mathbf{x}} \log \sum_{\mathbf{x}'\in\mathcal{X}} \omega_{\mathbf{x}'} \exp\left(f(\mathbf{x})^T f(\mathbf{x}')\right) \\ &= -\text{Tr}(\mathbf{F}^T\mathbf{A}\mathbf{F}) + \mathbf{1}\boldsymbol{\omega} \log\left(\exp(\boldsymbol{\omega}^{-\frac{1}{2}}\mathbf{F}\mathbf{F}^T\boldsymbol{\omega}^{-\frac{1}{2}})\boldsymbol{\omega}^T\mathbf{1}^T\right). \\ &= \mathcal{L}_{\text{align}} + \mathcal{L}_{\text{unif}}, \end{aligned} \tag{2}$$

$(\mathbf{x}, \mathbf{x}') \sim \omega_{\mathbf{x},\mathbf{x}'}$. $\mathbf{x}$ and $\mathbf{x}'$ are sampled from data distribution $\mathcal{X}$. $\omega_{\mathbf{x},\mathbf{x}'}$ is the joint probability of $\mathbf{x}$ and $\mathbf{x}'$. Here, we propose the concept of a transformation flow, which is a series of transformations connecting two samples:

$$\mathbf{x} \to \mathbf{x}_1 \to \cdots \to \mathbf{x}_n \to \mathbf{x}', \tag{3}$$

$$\omega_{\mathbf{x},\mathbf{x}'} = \mathbb{E}_{(\mathbf{x}_1,\ldots,\mathbf{x}_n)\in\mathcal{X}^n}\left[\mathcal{A}(\mathbf{x}_1|\mathbf{x})\mathcal{A}(\mathbf{x}_2|\mathbf{x}_1)\cdots\mathcal{A}(\mathbf{x}'|\mathbf{x}_n)\right], \tag{4}$$

where $\mathcal{A}(\cdot|\mathbf{x})$ is distribution of augmented data given $\mathbf{x}$. And $\omega_{\mathbf{x}} = \sum_{\mathbf{x}'\in\mathcal{X}} \omega_{\mathbf{x},\mathbf{x}'}$ is the marginal distribution of $\mathbf{x}$. $\boldsymbol{\omega} \in \mathbb{R}^{N\times N}$ is a diagonal matrix with $\boldsymbol{\omega}_{\mathbf{x},\mathbf{x}} = \omega_{\mathbf{x}}$. $\mathbf{Z}$ is the feature matrix with $f(\mathbf{x})^T$ as the $i-$th row. $\mathbf{F} = \sqrt{\boldsymbol{\omega}}\mathbf{Z}$. $\mathbf{A} \in \mathbb{R}^{N\times N}$ is the adjacency matrix defined by the data transformations. The weights $\mathbf{A}_{\mathbf{x},\mathbf{x}'} = \frac{\omega_{\mathbf{x},\mathbf{x}'}}{\sqrt{\omega_{\mathbf{x}}\omega_{\mathbf{x}'}}}$. $\mathbf{L}$ is the Laplacian matrix:

$$\mathbf{L} = \mathbf{I} - \mathbf{A}.$$

$$\text{Tr}(\mathbf{F}^T\mathbf{L}\mathbf{F}) = -\text{Tr}(\mathbf{F}^T\mathbf{A}\mathbf{F}) + \text{const}, \tag{5}$$

const means a constant. This illustrates that contrastive learning is equivalent to optimizing spectral clustering with regularization constraints. Thus the dynamics of contrastive learning can be modelled as follows.

B. Gradient Dynamics

We derive $\mathcal{L}_{\text{simclr}}$ to obtain the update process for the features. For alignment, We take the derivative of $\mathcal{L}_{\text{align}}$ as:

$$\dot{\mathbf{F}}_{\text{align}} = -\eta\nabla_{\mathbf{F}}\mathcal{L}_{\text{align}} = \eta\mathbf{A}\mathbf{F}, \tag{6}$$

where η is the learning rate. We observe that the alignment loss in the context of contrastive learning plays a pivotal role in the feature update process. Specifically, it updates the feature of $\mathbf{x}$ by considering a weighted summation that aggregates features from the neighborhood $\mathcal{N}_{\mathbf{x}}$.

For uniformity, the derivative of $\mathcal{L}_{\text{unif}}$ is as follows:

$$\dot{\mathbf{F}}_{\text{unif}} = -\eta\nabla_{\mathbf{F}}\mathcal{L}_{\text{unif}} = -\eta(\mathbf{D'}^{-1}\boldsymbol{\omega}^{\frac{1}{2}}\mathbf{A}'\boldsymbol{\omega}^{\frac{1}{2}})\mathbf{F}, \tag{7}$$

where $\mathbf{A}' = \exp(\boldsymbol{\omega}^{-\frac{1}{2}}\mathbf{F}\mathbf{F}^T\boldsymbol{\omega}^{-\frac{1}{2}})$ and $\mathbf{D}' = \deg(\mathbf{A}'\boldsymbol{\omega})$.

We present our conclusive update rule for contrastive learning combining the alignment update and uniformity update:

$$\dot{\mathbf{F}} = \dot{\mathbf{F}}_{\text{align}} + \dot{\mathbf{F}}_{\text{unif}} = \eta(\mathbf{A} - \mathbf{D'}^{-1}\boldsymbol{\omega}^{\frac{1}{2}}\mathbf{A}'\boldsymbol{\omega}^{\frac{1}{2}})\mathbf{F}. \tag{8}$$

Hence, the training procedure aims to instruct the network in aligning the graph of feature similarities $\mathbf{A}'$ with the data-augmentation graph created from data transformations $\mathbf{A}$.

Equilibrium State: After reaching equilibrium, the difference between the two graphs converges to 0. That is $\mathbf{A} - \mathbf{D'}^{-1}\boldsymbol{\omega}^{\frac{1}{2}}\mathbf{A}'\boldsymbol{\omega}^{\frac{1}{2}} = 0$. In this way, we employ the similarity between features to estimate the adjacency matrix defined by the data transformation:

$$\frac{\omega_{\mathbf{x},\mathbf{x}'}}{\omega_{\mathbf{x}}\omega_{\mathbf{x}'}} = \frac{\exp(f(\mathbf{x})^T f(\mathbf{x}'))}{\sum_{\mathbf{x}'\in\mathcal{X}} \omega_{\mathbf{x}'} \exp\left(f(\mathbf{x})^T f(\mathbf{x}')\right)} = p(\mathbf{x}). \tag{9}$$

This converges to fixed points in the feature space:

$$\mathbf{A}\mathbf{F}^* = \mathbf{D'}^{-1}\boldsymbol{\omega}^{\frac{1}{2}}\mathbf{A}'\boldsymbol{\omega}^{\frac{1}{2}}\mathbf{F}^*. \tag{10}$$

Thus, at equilibrium, the data on the transformation flow converges exponentially to the centre point $\mathbf{p}$:

$$\mathbf{p} = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i). \tag{11}$$

C. Exploring Augmentation Complexity

From this we note that contrastive learning is essentially the alignment of transformation space and metric space. In the transformation space, we use various data transformations to connect data samples of the same category. And in the metric space, we prove that contrastive learning uses the distance between features to estimate the probability of the transformation flow between two samples. We use the number of clusters to represent data transformation strength. Longer transform flows lead to fewer number of clusters. Besides, we use the category diameter as a quantitative measure of cluster purity and show that longer transformation flows lead to smaller diameters.

Spectral Clustering Number: For the Laplace matrix, the geometric multiplicity of zero eigenvalue (or the number of eigenvalues less than a threshold for approximation) represents the tightly connected parts of the graph. Therefore, we estimate the number of clusters to represent the data transformation strength using the geometric multiplicity of zero eigenvalue.

Specifically, we analyze why the geometric multiplicity of the zero eigenvalue represent the tightly connected parts of the graphs. Our analysis is formalized using the Cheeger Inequality from graph theory. We first give the definition of Cheeger Constant:

Definition 1 (Cheeger Constant): Consider a subset of nodes, denoted as $\mathbf{S}$, within graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$. The set $\partial\mathbf{S}$ captures the edges originating from nodes in $\mathbf{S}$ and terminating in nodes outside $\mathbf{S}$ as $\partial\mathbf{S} = \{(\mathbf{u}, \mathbf{v}) \in \mathbf{E} : \mathbf{u} \in \mathbf{S}, \mathbf{v} \in \mathbf{V} \setminus \mathbf{S}\}$. In this context, the Cheeger Constant, represented as $h_{\mathbf{G}}$, characterizes the

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

connectivity of graph $\mathbf{G}$ and is defined as $h_{\mathbf{G}} = \min_{\mathbf{S}} \Phi_{\mathbf{G}}(\mathbf{S})$, where

$$\Phi_G(\mathbf{S}) = \frac{\partial \mathbf{S}}{\min(\text{vol}(\mathbf{S}), \text{vol}(\mathbf{V}) - \text{vol}(\mathbf{S}))}, \tag{12}$$

which is the conductance, with $\text{vol}(\mathbf{S})$ denoting the sum of degrees of nodes within set $\mathbf{S}$.

In a more intuitive sense, the Cheeger Constant is small when there is a bottleneck within graph, meaning that there are two sets of nodes with only a limited number of edges connecting them. Conversely, we can deduce that $h_{\mathbf{G}}$ is greater than zero if and only if graph is a connected graph. This notion aligns with the idea that the Cheeger Constant quantifies the connectivity and bottleneck properties of the graph.

We extend Cheeger Constant to higher orders to describe the results of k-clustering:

$$h_{\mathbf{G}}(k) = \min_{\text{partition } \mathbf{C}_1,\ldots,\mathbf{C}_k} \max_i \Phi_G(\mathbf{C}_i), \tag{13}$$

where $\mathbf{C}_i$ is a subgraph of $\mathbf{G}$. The metric $h_{\mathbf{G}}(k)$ becomes minimal specifically when the graph $\mathbf{G}$ can be divided into k clusters, each characterized by low conductance. The correlation between the k-way expansion of graph $\mathbf{G}$ and the eigenvalues of its Laplacian matrix is elucidated by the higher-order Cheeger inequality.

Theorem 1: Let $\mathbf{L}$ be the graph Laplacian matrix of $\mathbf{G}$, and let $v_1 \leq v_2 \leq \cdots \leq v_n$ be the eigenvalues of the normalised Laplacian matrix. Therefore, we obtain:

$$\frac{v_k}{2} \leq h_{\mathbf{G}}(k) \leq O(k^3)\sqrt{v_k}. \tag{14}$$

This lemma reveals that establishing an upper bound on $h_{\mathbf{G}}(k)$ and a lower bound on v_{k+1} are adequate conditions to ensure the possibility of partitioning $\mathbf{G}$ into k clusters with low conductance while preventing its partition into k + 1 clusters. Such conditions are frequently employed in the analysis of graph clustering.

• *Number of Clusters:* Therefore, to quantitatively assess the strength of the data transformation, we use the number of zero eigenvalues of the Laplace matrix $\mathbf{L}$ to characterise the strength of the data transformation. This can be written:

$$\kappa = \sum_{i=1}^{n} \mathbb{1}[h_{\mathbf{G}}(k) < \frac{\epsilon}{2}] \approx \sum_{i=1}^{n} \mathbb{1}[v_i < \epsilon], \tag{15}$$

where $\epsilon > 0$ is a threshold. As shown in Fig. 1, the maximum number of singular values below the threshold is used as the data transformation complexity κ. $v_\kappa < \epsilon$ and $v_{\kappa+1} > \epsilon$. Strong data transformation leads to fewer number of clusters, so the parameter κ is small. On the contrary, a large κ indicates that there are more clusters and the data transformation is weak.

• *Relation to Downstream Tasks:* To quantify the validity of our proposed metrics with the accuracy of downstream tasks, we use three data transformations to model different intensities of data transformation on the NTU RGB+D Dataset 60 [23], followed by linear evaluation with a classifier $\phi(\cdot)$ to obtain the action recognition accuracy.

$$\mathbf{x} = \mathbf{M} \odot (\mathbf{xR}) + \mathbf{N}, \tag{16}$$

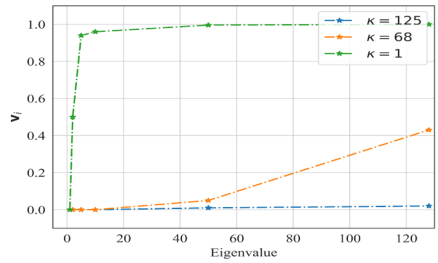


Fig. 1. Curve of singular values with the singular value index of the Laplacian matrix $\mathbf{L}$ of $\mathbf{A}$. The number of small singular values corresponds to the number of clusters. Different colors correspond to different transformation strengths.

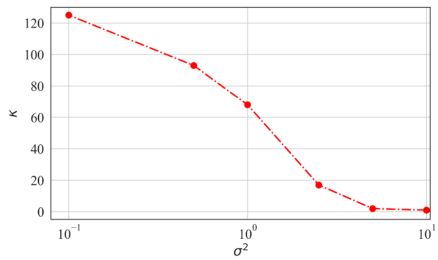


Fig. 2. Curve of Gaussian noise σ^2 with κ. The strong data transformation leads to small κ, representing a small number of clusters.

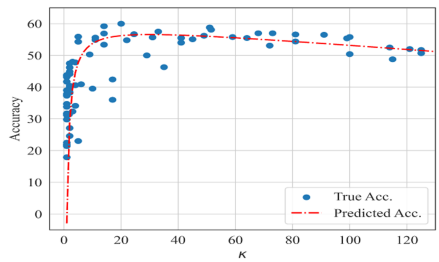


Fig. 3. Curves of κ with accuracy. The red line is the accuracy y_{acc} of our prediction using κ. As the data transformation is enhanced, the accuracy first rises and then falls, representing the optimal data transformation.

where $\mathbf{M}$ is the mask matrix, $\mathbf{R}$ is the shear matrix and $\mathbf{N}$ is the Gaussian noise matrix. We can control the intensity of the data transformation by adjusting the masking ratio of the masking matrix, the intensity of the shear matrix and the variance of the noise matrix. Fig. 2 shows the relationship between adjusting the variance σ^2 of the Gaussian noise and our metrics κ, and it can be seen that a larger variance leads to a stronger data transformation and thus a smaller metric κ. The Fig. 3 shows the relationship between the accuracy and the parameter κ, and it can be seen that the accuracy first increases with increasing κ and then decreases. More specifically, this function relationship between accuracy and κ can be quantified as:

$$y_{\text{acc}} = 60.97 - 0.069\kappa - \frac{63.64}{\kappa}, \tag{17}$$

as shown by the red line in Fig. 3. There exists a maximum value for this function that corresponds to the optimal data transformation intensity. We can prove this relation by the following theorem [58].

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

Theorem 2: If $v_1 \leq v_2 \leq \cdots \leq v_n$ are the eigenvalues of the normalized Laplacian matrix $\mathbf{L}$ of $\mathbf{A}$, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ are the eigenvalues of $\mathbf{A}$, $\lambda_i = 1 - v_i$ and if the clustering purity is $1-\alpha$. $\alpha = P[y_{\mathbf{x}} \neq y_{\mathbf{x}'}] \leq h_{\mathbf{G}}(k) \leq \frac{\epsilon}{2}$. $y_{\text{acc}} = P[\phi(\mathbf{x}) = y_{\mathbf{x}}]$. we obtain:

$$\begin{aligned} 1 - y_{\text{acc}} &\leq c_1 \sum_{i=d+1}^{n} \lambda_i^2 + c_2\alpha \\ &= c_1 \sum_{i=d+1}^{\kappa} (1 - v_i)^2 + c_2\alpha \\ &\leq c_1(\kappa - d) + c_2\alpha \\ &= c_1\kappa + \frac{c_2}{\kappa} + c_3, \end{aligned} \tag{18}$$

where $\alpha \approx O(\frac{1}{\kappa})$ under the current data transformation. c_1, c_2, c_3 are some constants.

This theorem illustrates the constraints on accuracy imposed by the purity $1-\alpha$ and number κ of clusters. A large purity and a small number of clusters result in a low error rate. When only a random data transformation is applied, it fails to retain the motion information. Consequently, when the data transformation is significant, although the number of clusters may decrease, various categories may be clustered together, leading to a decline in purity.

Measuring Category Diameter: Further, we investigate the effect of the intensity of data transformation in the metric space. This section provides a quantitative analysis of cluster purity by category diameter and intra-class distance. The maximum distance between converged features (category diameter) decreases as the length of the transform flow increases.

Theorem 3: Let $\mathbf{z}_1, \ldots, \mathbf{z}_N \sim \mathcal{N}(\mathbf{p}, \sigma^2)$ be samples of a same category. We define the category diameter as:

$$D = \sup_{\mathbf{z}_i, \mathbf{z}_j} \|\mathbf{z}_i - \mathbf{z}_j\|, \tag{19}$$

Then when the feature reaches equilibrium, we obtain:

$$\mathbf{p}_i = \frac{1}{n} \sum_{j=1}^{n} \mathbf{z}_{ij}. \tag{20}$$

$\mathbf{z}_{ij}$ is in the transformation flow of $\mathbf{z}_i$. Then the expectation of the diameter decreases with the growth of the transformation flow:

$$E[D] \sim 2\sqrt{\frac{2\sigma^2 \log(N)}{n}}, \tag{21}$$

where N is the number of samples.

• *Relation to Linear Separability:* Through the analysis of previous work [59], we find that category diameter is directly related to linear separability performance.

Theorem 4: Define the distance between categories $\mathbf{p}$ and $\mathbf{q}$ as:

$$L = \inf_{\mathbf{z}_i \sim \mathcal{N}(\mathbf{p}, \sigma^2),\ \mathbf{z}_j \sim \mathcal{N}(\mathbf{q}, \sigma^2)} \|\mathbf{z}_i - \mathbf{z}_j\|, \tag{22}$$

When the distance-diameter ratio is sufficiently large:

$$\frac{L}{D} \geq \frac{\kappa(\kappa-1)\sqrt{\pi}d}{4}, \tag{23}$$

the feature space can be separated by $\kappa - 1$ hyperplanes, where d is the dimension size.

As stronger data transformations are applied, the category diameters decrease and thus the clustering becomes tighter. This leads to an improvement in the linear divisibility of the feature space.

D. Towards Better Transformation Flow

To learn a robust representation space, the construction of the transformation flow is essential. A well-designed transformation flow allows for a more effective separation of classes in the feature space.

Based on our analysis above, the optimal data transformation flow requires a reduction in the number of clusters while maintaining the purity of the clusters:

$$\mathcal{T} = \arg\min \kappa, \text{ with } \alpha \leq \frac{\epsilon}{2}, \tag{24}$$

where $\mathcal{T}$ is the transformation flow. And we have found, that clustering purity is strongly correlated with interclass distances and intraclass diameters. This is therefore equivalent to:

$$\mathcal{T} = \arg\min \kappa, \text{ with } \frac{L}{D} \geq \frac{\kappa(\kappa-1)\sqrt{\pi}d}{4}, \tag{25}$$

We consider these two parts of the constraint separately. Specifically, a longer transformation flow tends to reduce the number of clusters. According to the definition of κ:

$$\begin{aligned} \mathcal{T} &= \arg\min \kappa = \arg\min \sum_{i=1}^{n} \mathbb{1}[v_i < \epsilon] = \arg\max \sum_{i=1}^{n} v_i \\ &= \arg\max \text{Tr}(\mathbf{L}) = \arg\max \text{Tr}(\mathbf{F}^T\mathbf{L}\mathbf{F}) = \arg\max \mathcal{L}_{\text{align}}, \end{aligned} \tag{26}$$

under the condition that the features are orthogonal to each other $\mathbf{F}\mathbf{F}^T = \mathbf{I}$. We then consider interclass distances and intraclass diameters to maintain clustering purity. Without any class-supervised information introduced, the best data flow is distance-preserving mapping, *i.e.*, the inter-data distances are not changed before and after the data transformation to avoid introducing noisy data to destroy the data structure:

$$D = \sup_{\mathbf{z}_i, \mathbf{z}_j} \|\mathbf{z}_i - \mathbf{z}_j\| \approx 2\|\mathbf{z} - \mathbf{p}\| = 4(1 - \mathbf{z}^T\mathbf{p}), \tag{27}$$

where $\mathbf{z}$ is the feature point furthest from its clustering center $\mathbf{p}$.

$$L = \inf_{\mathbf{z}_i,\ \mathbf{z}_j} \|\mathbf{z}_i - \mathbf{z}_j\| \leq \frac{1}{N} \sum_{\mathbf{z}_j} \|\mathbf{z} - \mathbf{z}_j\| = 2(1 - \mathbf{z}^T\mathbf{b}), \tag{28}$$

where the minimum distance is less than or equal to the average distance. $\mathbf{b} = \sum \mathbf{z}_j / N$ is the average motion feature. Maintaining feature distance means:

$$\frac{\partial D}{\partial \mathbf{x}} \leq 0, \quad \frac{\partial L}{\partial \mathbf{x}} \geq 0. \tag{29}$$

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

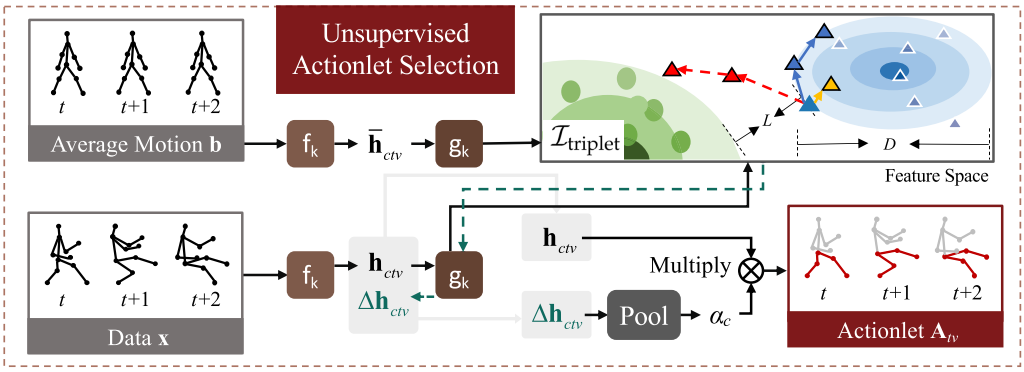


Fig. 4. The region of motion is identified in an unsupervised manner by comparing the data with static sequences and action prototypes, and locating the actionlet region where the motion occurs via gradient backpropagation. Here shows interclass distances and intraclass diameters in feature space. The optimal data transformation should maintain this distance metric to learn the data structure. Therefore, three different data transformation flows are shown here. The red transformation flow causes the interclass distance to decrease because of too strong a data transformation, leading to an increase in the error rate. Whereas the yellow transformation flow is weak, which leads to difficulty in learning rich motion patterns. Our approach uses the blue transform flow to provide more motion information to connect more samples while maintaining the structure of the data.

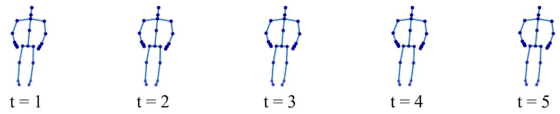


Fig. 5. Visualization of the average motion sequence. Average motion represents the average of a sequence of motions and is used to estimate the average distance between data in (28) and (36). The sequence exhibits no clear action and is considered a static anchor.

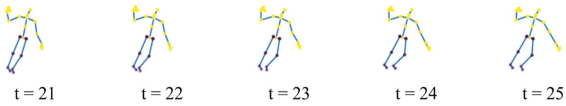


Fig. 6. In unsupervised actionlet selection, the action prototypes and static sequences are utilized as anchors to acquire the region of motion. The yellow joints are the actionlet. Note that hand movements are mainly selected, indicating that the actionlet is reasonable.

We need to keep the diameter within classes from increasing and the distance between classes from decreasing. The following section discusses specifically how to get the optimal transform flow.

IV. Actionlet-Guided Contrastive Learning

Based on the above analysis, we conclude that the optimal data transformation is to reduce the number of clusters κ while keeping the purity $1-\alpha$ large. Therefore, to maintain the purity of the clusters while reducing the number of clusters, we design a data transformation that preserves motion information. To comprehensively and quantitatively capture the motion information, we introduce the concept of actionlet, which serves as a quantitative measure to assess the motion information extracted by model as shown in Fig. 6.

Actionlets are essentially atomic elements of motion, carefully designed to represent the most granular and task-relevant aspects of movement. By decomposing motion into these discrete actionlets, we gain a finer level of understanding over the underlying dynamics. This not only enables us to precisely quantify the extracted motion information but also facilitates a more meticulous analysis of how the network learns and represents this critical semantics of skeleton data. Specifically, we can use actionet to decompose action sequences and decouple them into mutually independent feature spaces at the feature level. Based on the actionlet-decomposition, we propose an actionlet-based motion-adaptive data transformation method to effectively preserve motion information. Through an unsupervised approach, we identify motion regions as actionlets and enhance the data transformation in non-actionlet regions, reducing the number of clusters and improving clustering purity. Additionally, we introduce intra-data similarity distillation to strengthen feature consistency and inter-data contrastive learning to uncover richer and more diverse motion patterns.

Compared to our previous work, action prototypes derived from clustering are leveraged to identify and select motion regions effectively. Building on this, we introduce an enhanced motion-aware data transformation that incorporates newly proposed strong transformations, including adversarial noise and skeleton masking, to enrich the diversity of data representations. Furthermore, at the loss level, we design two novel loss terms to capture fine-grained motion details and inter-data relationships. The first term employs mix-based inter-sequence similarity distillation to model relational dependencies, while the second term uses mask reconstruction to ensure robust feature learning and preserve critical motion semantics.

A. Unsupervised Actionlet Selection

To fulfil the equations above, we first analyze the relationship between global and local features.

$$\begin{aligned}\mathbf{z} &= \mathrm{GAP}(g_k(\mathbf{h}^l_{ctv})) \\ &= \frac{1}{TV}\sum_{t=1}^{T}\sum_{v=1}^{V}\mathbf{P}\mathbf{h}^l_{ctv},\end{aligned} \tag{30}$$

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

where $\mathbf{P}$ is the weights of $g_k(\cdot)$. And $\mathbf{h}_{ctv}^l = f_k(\mathbf{x})$ is the l-th output. For both GCN and transformer architectures, each layer of the network can be conceptualized as comprising two key components: feature fusion between joints and feature transformation module. Additionally, residual connections are established between these components across all layers, facilitating information flow and gradient propagation:

$$\begin{aligned}\tilde{\mathbf{h}}_{ctv}^l &= \text{Fusion}(\mathbf{h}_{ctv}^{l-1}) + \mathbf{h}_{ctv}^{l-1},\\ \mathbf{h}_{ctv}^l &= \text{Trans}(\tilde{\mathbf{h}}_{ctv}^l) + \tilde{\mathbf{h}}_{ctv}^l,\end{aligned} \tag{31}$$

global features can be rewritten as

$$\begin{aligned}\mathbf{z} &= \frac{1}{TV}\sum_{t=1}^{T}\sum_{v=1}^{V}\mathbf{P}\sum_{l=1}^{L}\left(\text{Fusion}(\mathbf{h}_{ctv}^{l-1}) + \text{Trans}(\tilde{\mathbf{h}}_{ctv}^l)\right)\\ &= \frac{1}{TV}\sum_{t=1}^{T}\sum_{v=1}^{V}\sum_{l=1}^{L}\mathbf{c}_{ctv}^l,\\ \mathbf{c}_{ctv}^l &= \mathbf{P}\left(\mathbf{W}^l\mathbf{h}_{ctv}^{l-1}\mathbf{G}_{vv} + \tilde{\mathbf{W}}^l\tilde{\mathbf{h}}_{ctv}^l\right),\end{aligned} \tag{32}$$

where $\mathbf{W}^l$ and $\tilde{\mathbf{W}}^l$ are the weights of the l-th feature fusion module and l-th feature transformation module, respectively. $\mathbf{G}_{vv}$ is the adjacency matrix of skeleton data. By understanding the relationship between global and local features, we can begin to extract motion regions from the local features, effectively identifying them as actionlets.

To efficiently extract actionlets from skeleton data, we employ a gradient-based approach. Traditional methods for mining actionlets rely heavily on action labels to delineate motion regions. However, this approach becomes impractical in unsupervised learning scenarios. Taking inspiration from the principles of contrastive learning, we introduce an innovative unsupervised spatio-temporal actionlet selection technique in Fig. 4.

Action Prototypes as Anchors: We employ action prototypes as positive anchors and static sequences as negative anchors, effectively extracting motion regions through the utilization of triplet mining loss.

We define action equilibrium points $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_N\}$ as N clustering centers of the data as positive prototypes. These prototypes, denoted as $\mathcal{P}$, inherently possess the capability to represent diverse classes. Given that features on the unit hypersphere $\mathbb{S}^{d-1}$, we perform k-Means clustering on the hypersphere manifold with Riemannian metric.

$$\mathcal{P} = \text{SPHERE\_CLUSTER}(\mathbf{z}), \tag{33}$$

where SPHERE_CLUSTER$(\cdot)$ is the hypersphere manifold k-Means clustering and $f(\cdot)$ is an encoder. The clustering center corresponding to the feature $\mathbf{z}$ is $\mathbf{p}$. So the similarity of features and clustering centers can be:

$$\begin{aligned}\text{sim}(\mathbf{z},\mathbf{p}) &= \mathbf{p}^T\mathbf{z}\\ &= \frac{1}{TV}\sum_{t=1}^{T}\sum_{v=1}^{V}\sum_{l=1}^{L}\mathbf{p}^T\mathbf{c}_{ctv}^l.\end{aligned} \tag{34}$$

So the derivation for each joint gives us this equation:

$$\frac{\partial\text{sim}(\mathbf{z},\mathbf{p})}{\partial\mathbf{x}_i} = \frac{1}{TV}\sum_{t=1}^{T}\sum_{v=1}^{V}\sum_{l=1}^{L}\mathbf{p}^T\frac{\partial\mathbf{c}_{ctv}^l}{\partial\mathbf{x}_i}, \tag{35}$$

where $\mathbf{x}_i$ is the i-th joint of $\mathbf{x}$. Hence, the gradient of a joint on its feature within the linear space of the action prototype identifies it as part of the actionlet for that particular action prototype.

Gradient Activation Mapping for Actionlet Localization: To delineate the motion region, we compute the triplet loss of two features, wherein the static motion and the nearest clustering center serve as the negative and positive anchors, respectively. This approach involves contrasting the features of the static motion with those of the nearest clustering center to facilitate the identification of the motion region within the dataset, formalized as:

$$\begin{aligned}\mathcal{I}_{\text{triplet}} &= \left[-\text{sim}(\mathbf{z},\mathbf{b}) + \text{sim}(\mathbf{z},\mathbf{p}) + \gamma\right]_+,\\ \mathbf{p} &= \arg\max_{\mathbf{p}\in\mathcal{P}}\text{sim}(\mathbf{z},\mathbf{p}),\end{aligned} \tag{36}$$

where $[\cdot]_+ = \max(\cdot, 0)$. γ is the margin between positive and negative pairs. $\mathbf{b} = \text{GAP}(g_k(f_k(\sum_{\mathbf{x}\in\mathcal{X}}\omega_{\mathbf{x}}\mathbf{x})))$ is the average motion as a statics anchor as shown in Fig. 5.

To determine the motion region, we perform backpropagation to compute the gradient of the triplet loss with respect to the dense feature $\mathbf{h}_{ctv}$. This process allows us to identify the gradient flow and assess how changes in the dense feature affect the triplet loss, thereby guiding us in locating the motion region within the data. The calculated gradients are then pooled over the joint and temporal dimensions to get the neuron importance weights α_c^i:

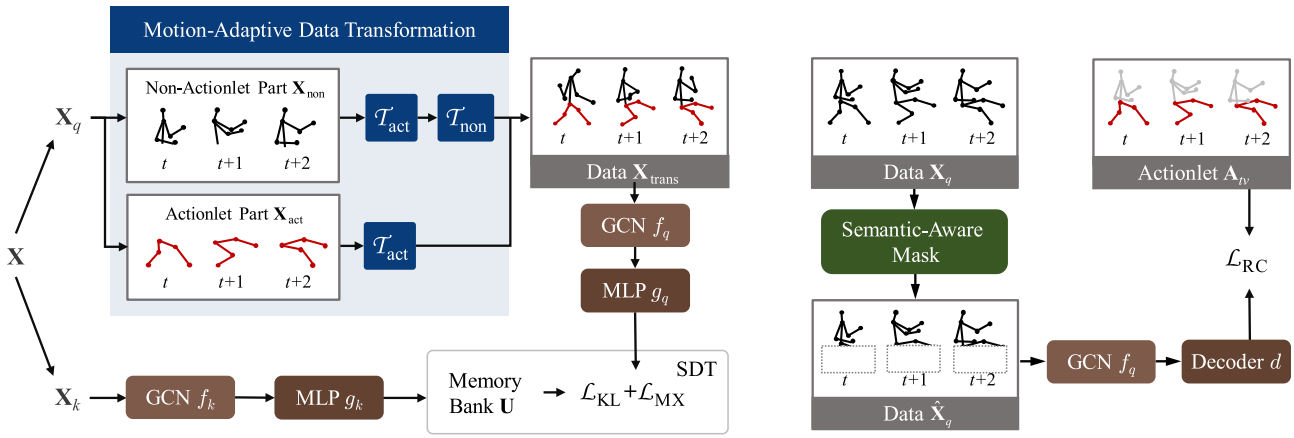
$$\begin{aligned}\Delta\mathbf{h}_{ctv} &= \frac{\partial\mathcal{I}_{\text{triplet}}}{\partial\mathbf{h}_{ctv}},\\ \alpha_c &= \frac{1}{TV}\sum_{t=1}^{T}\sum_{v=1}^{V}[\Delta\mathbf{h}_{ctv}]_+.\end{aligned} \tag{37}$$

We approximate the neural network with a linear subspace in the feature neighborhood by computing the gradient. With the activation function, we filter the features that contribute negatively to the triplet loss, and these are the static regions that are similar to the average motion. α_c reflects the separability of each channel; some channels are always activated or always inactive and therefore not separable. Some channels capture specific patterns and hence have a high degree of separability and are thus noticed. Subsequently, we conduct a weighted combination of the forward activation maps and the neuron importance weights:

$$\mathbf{A}_{tv} = \left[\sum_{c=1}^{C}\alpha_c\mathbf{h}_{ctv}\right]_+\mathbf{G}_{vv}. \tag{38}$$

The adjacency matrix of skeleton data $\mathbf{G}_{vv}$ is utilized for importance smoothing. This is because the forward activation maps show patterns of different regions extracted by the network. Large activation values mean that a specific motion pattern is extracted. And large neuron importance weights represent positive contribution of this pattern to the triplet loss. Thus the two are multiplied to indicate the regions in the sequence with

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.



(a) Motion-Adaptive Similarity Distillation (b) Semantic-Aware Masked Motion Modeling

Fig. 7. Our algorithm comprises two key modules: (a) Motion-Adaptive Similarity Distillation (MASD) and (b) Semantic-Aware Masked Motion Modeling (SAM^3). The MASD module incorporates two streams: an online stream and an offline stream. The online stream, located at the top, is updated via gradient descent, whereas the offline stream, positioned at the bottom, is updated using momentum. The process commences with the input data $\mathbf{x}_q$ in the online stream. Motion-Adaptive Data Transformation (MATS) is then applied to $\mathbf{x}_q$ using actionlets derived from the offline stream, yielding the augmented data $\mathbf{x}_{\text{trans}}$. Subsequently, via Similarity Distillation (SDT), the pipeline endeavors to maximize similarity between positive samples while minimizing similarity between negative samples. To enhance the accuracy of motion information extraction, we introduce a Semantic-Aware Masked Motion Modeling approach. This involves masking out motion regions based on actionlets within the Semantic-Aware Mask Module. A decoder is then trained to predict the masked skeleton data, enabling the generation of a fine-grained motion representation. Through this process, our algorithm aims to capture nuanced motion patterns while leveraging semantic information to improve the modeling of motion dynamics.

positive contribution to the separation of data point from other clustering centers and static sequences.

To keep the semantics of the skeleton sequence unchanged, we just need to preserve the skeleton of the actionlets region to be identifiable. This is because

$$\mathcal{I}_{\text{triplet}} = \left[\frac{L}{2} - \frac{D}{4} + \gamma\right]_+. \tag{39}$$

Because a change in the region of actionlets region greatly affects $\mathcal{I}_{\text{triplet}}$, it may change the distance between the data, resulting in an increase in the diameter within classes or a decrease in the distance between classes. Therefore, we maintain the motion semantics of this region to maintain the data structure.

B. Motion-Adaptive Similarity Distillation

Building upon the actionlet regions, we introduce an actionlet-dependent contrastive learning approach to maximize the advantages offered by actionlets, as depicted in Fig. 7. In this approach, we apply data transformations to different regions within the Motion-Adaptive Similarity Distillation (MASD) module to reduce the number of clusters κ and improve clustering purity $1-\alpha$. Additionally, we incorporate inter-data relationship modeling through skeleton mix contrastive learning, facilitating the exploration of interactions between different data samples.

Motion-Adaptive Transformation Strategy (MATS): The choice of data transformation $\mathcal{T}$ is critical for in extracting semantic information and enhancing generalization capabilities. Designing data transformations while retaining motion-relevant information remains a challenge. Better transformation leads to a decrease in the number of clusters κ in spectral clustering while maintain a high cluster purity $1-\alpha$.

To tackle the challenge of limited diversity in simple transformations and the risk of information loss in overly complex transformations, we propose motion-adaptive data transformations tailored for skeleton data, leveraging the concept of actionlets.

To address the specific characteristics of actionlet and non-actionlet regions, we propose two distinct transformations: the actionlet transformation and the non-actionlet transformation.

• *Actionlet Transformation $\mathcal{T}_{act}$:* Within the actionlet regions, we perform data transformations to augment the diversity of patterns, drawing inspiration from prior research [16]. Specifically, we employ the following spatial transformations: {Shear, Spatial Flip, Rotate, and Axis Mask}. Additionally, two temporal transformations, namely {Crop and Temporal Flip}, are applied to introduce temporal variations. To introduce spatio-temporal variations, we utilize {Gaussian Noise and Gaussian Blur} as two additional transformations. Collectively, these transformations enrich the augmented data with diverse spatio-temporal patterns, enhancing the learning process and enabling the model to capture a broader range of motion dynamics and semantic information within the actionlet regions.

• *Non-Actionlet Transformation $\mathcal{T}_{non}$:* To enhance the model's generalization capability, we introduce several additional data transformations specifically targeted at the non-actionlet regions. These transformations include techniques such as dropout, rescale, and others. Besides, we implement an intra-sequence data transformation {Adversarial Noise} and an inter-sequence data transformation {Skeleton Mask}.

The problem of *ActCLR* is that, the randomly selected data transformation $\mathcal{T}$ may be weak. Therefore, we propose adversarial training as an enhancement strategy for data transformation to improve feature representations.

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

To update the parameters of the data transformation with gradients, we add Adversarial Noise $\mathbf{N}$ to data $\mathbf{x}$:

$$\mathcal{T}_{\mathbf{N}}(\mathbf{x}) = \mathbf{x}_{\text{noise}} = \mathbf{x} + \mathbf{N}, \tag{40}$$

where the jittering matrix $\mathbf{N}$ is initialized to $\mathbf{0}$. Based on previous analyses, we obtain:

$$\mathcal{T}_{\mathbf{N}} = \arg\max_{\mathbf{N}} \mathcal{L}_{\text{align}}. \tag{41}$$

We update the parameters of these three matrices by adversarial attack. Following the Fast Gradient Sign Method (FGSM) [60], we hope to make data transformations attack the triplet loss $\mathcal{I}_{\text{triplet}}$ to approximate $\mathcal{L}_{\text{align}}$ to fool the encoder $f(\cdot)$. Therefore, the parameters of the three matrices are updated as follows:

$$\mathbf{N} = \mathbf{N} - \epsilon \cdot \text{sgn}(\nabla_{\mathbf{N}} \mathcal{I}_{\text{triplet}}), \tag{42}$$

where ϵ is the learning rate. Then, we add Gaussian noise to strengthen $\mathbf{N}$ as well.

Skeleton Mask refers to scenarios where parts of the body are hidden. We mask a specific area such as the hands or legs:

$$\mathbf{x}_{\text{mask}} = \mathbf{x} \odot \mathbf{M} + \mathbf{d} \odot (1 - \mathbf{M}), \tag{43}$$

where $\mathbf{M}$ represents the mask. Because the partial padding of 0 after masking may cause the generated data to be outside the distribution resulting in anomalous output features, we experiment with different padding values $\mathbf{d}$ including 0, Gaussian noise, and sequences $\mathbf{x}'$ sampled randomly within the sample distribution.

• *Actionlet-Dependent Combination:* We apply data augmentation of different intensity to actionlet and non-actionlet regions, and combine them to obtain the final augmented data. It is formalized as:

$$\mathbf{x}_{\text{trans}} = \mathbf{A}_{tv} \odot \mathbf{x}_{\text{act}} + (1 - \mathbf{A}_{tv}) \odot \mathbf{x}_{\text{non}}, \tag{44}$$

where $\mathbf{x}_{\text{trans}}$ is the transformed skeleton sequence. First, we employ actionlet transformations $\mathcal{T}_{\text{act}}$ to obtain $\mathbf{x}_{\text{act}}$ and $\mathbf{x}_{\text{non}}$. Then, we utilize non-actionlet transformations $\mathcal{T}_{\text{non}}$ for $\mathbf{x}_{\text{non}}$. $\mathbf{A}_{tv}$ represents the actionlet. We reduce κ through enhanced data transformation $\mathcal{T}_{\text{non}}$ while using this actionlet-dependent combination module to maintain large purity $1 - \alpha$.

Similarity Distillation Training (SDT): In our approach, we employ both intra-sequence and inter-sequence similarity distillation learning to guide the model towards learning minimum sufficient representations by maximizing the mutual information between positive samples.

We initiate the pretraining process with two encoders: an online encoder denoted as $f_q(\cdot)$ and an offline encoder referred to as $f_k(\cdot)$. The online one is updated by back-propagating gradients, while the offline one is maintained as a momentum-updated version of the online one.

For the offline stream, we feed the original data $\mathbf{x}$ into the model $f_k(\cdot)$. Utilizing the unsupervised actionlet selection module, we obtain actionlet regions denoted as $\mathbf{A}_{tv}$. This process is described in detail in Section IV-A. The actionlet regions serve as key features for guiding the subsequent learning process, facilitating the extraction of discriminative motion patterns within the data.

After generating the actionlet regions $\mathbf{A}_{tv}$ with the offline stream, we subject the data to a data transformation $\mathcal{T}$ to derive two distinct views, namely, $\mathbf{x}_q$ and $\mathbf{x}_k$. Additionally, we employ the motion-adaptive transformation strategy (MATS) to improve the diversity of $\mathbf{x}_q$. This ensures that both views capture a wide range of motion patterns and semantic information, facilitating robust learning.

Subsequently, we optimize the contrastive learning process to promote stronger coherence and more diverse movement patterns. This is achieved through intra-sequence distillation and inter-sequence distillation, respectively.

• *Intra-Sequence Feature Extraction (Intra-SD):* We utilize two encoders, namely, an online encoder $f_q(\cdot)$ and an offline encoder $f_k(\cdot)$, to extract features from the input sequences. This yields feature representations $\mathbf{z}_q = g_q(f_q(\mathbf{x}_q))$ and $\mathbf{z}_k = g_k(f_k(\mathbf{x}_k))$, where $g_q(\cdot)$ and $g_k(\cdot)$ serve as online and offline projectors, respectively.

The parameters of the offline networks $f_k(\cdot)$ and $g_k(\cdot)$ are updated by leveraging the momentum of their online counterparts $f_q(\cdot)$ and $g_q(\cdot)$. Specifically, the offline networks are updated using the momentum update rule, $\hat{f} \leftarrow \alpha \hat{f} + (1 - \alpha) f$, where α represents a momentum coefficient.

To facilitate contrastive learning, we introduce a memory bank denoted as $\mathbf{U} = \{\mathbf{u}^i\}_{i=1}^M$ to store offline features. In each training batch, the features extracted from the offline data are stored in the memory bank. We maintain the memory bank using a first-in, first-out (FIFO) strategy to ensure continuous updates, thereby enabling efficient retrieval and utilization of historical feature representations during the contrastive learning process.

Following recent works [49], [50], we apply similarity distillation loss to optimize:

$$\begin{aligned} \mathcal{L}_{\text{KL}} &= -\mathbf{P}_k \log \mathbf{P}_q, \\ \mathbf{P}_q &= \text{SoftMax}(\text{sim}(\mathbf{z}_q, \mathbf{U})/\tau_q), \\ \mathbf{P}_k &= \text{SoftMax}(\text{sim}(\mathbf{z}_k, \mathbf{U})/\tau_k), \end{aligned} \tag{45}$$

where $\text{sim}(\mathbf{z}_q, \mathbf{U}) = [\text{sim}(\mathbf{z}_q, \mathbf{u}^j)]_{j=1}^M$, which indicates the similarity distribution between the representation $\mathbf{z}_q$ and other elements from $\mathbf{U}$.

• *Inter-Sequence Feature Extraction (Inter-SD):* In inter-sequence feature modelling, we use the mix method to obtain richer motion patterns. Specifically, *skeleton mixing* encompasses three distinct mixing methods tailored for skeleton data: cut mix [61], resize mix [62], and mix up [63].

In both cut mix and resize mix methods, the skeletal joints are initially categorized into multiple subsets based on different human body parts. We randomly select another skeleton sequence $\mathbf{x}'_q$ and blend it with the original data $\mathbf{x}_q$ at the level of a specific body part to generate the mixed data $\tilde{\mathbf{x}}_q$. The mixing mask $\tilde{\mathbf{M}}$ is defined as the mask indicating the replaced joints in $\mathbf{x}_q$. In the case of the mix-up method, we straightforwardly blend all the joints of $\mathbf{x}_q$ and $\mathbf{x}'_q$ using a mask $\tilde{\mathbf{M}}$ to produce the mixed data $\tilde{\mathbf{x}}_q$:

$$\tilde{\mathbf{x}}_q = \tilde{\mathbf{M}} \odot \mathbf{x}_q + (1 - \tilde{\mathbf{M}}) \odot \mathbf{x}'_q. \tag{46}$$

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

In our implementation, we randomly choose one of the three mixing methods mentioned above and apply it to the skeleton data. This random selection allows for flexibility, as constraining the model to learn invariance under the transformation might be unreasonable, given that semantic preservation is not guaranteed when applying it.

In light of this, drawing inspiration from the use of mixed labels in prior works [61], [62], [63], we manually construct $\tilde{\mathbf{x}}_k$ as the target feature to be learned. This construction is based on the mixing mask and actionlets:

$$a = \frac{\sum_{t=1}^{T}\sum_{v=1}^{V}\tilde{\mathbf{M}} \odot \mathbf{A}_{tv}}{\sum_{t=1}^{T}\sum_{v=1}^{V}\mathbf{A}_{tv}},$$

$$b = \frac{\sum_{t=1}^{T}\sum_{v=1}^{V}(1-\tilde{\mathbf{M}}) \odot \mathbf{A}'_{tv}}{\sum_{t=1}^{T}\sum_{v=1}^{V}\mathbf{A}'_{tv}},$$

$$\tilde{\mathbf{z}}_k = \frac{a}{a+b} f_k(\mathbf{x}_k) + \frac{b}{a+b} f_k(\mathbf{x}'_k). \tag{47}$$

Similarly, we use similarity distillation loss to align the positive and negative samples.

$$\mathcal{L}_{\text{MX}} = -\tilde{\mathbf{P}}_k \log \tilde{\mathbf{P}}_q,$$

$$\tilde{\mathbf{P}}_q = \text{SoftMax}(\text{sim}(\tilde{\mathbf{z}}_q, \mathbf{U})/\tau_q),$$

$$\tilde{\mathbf{P}}_k = \text{SoftMax}(\text{sim}(\tilde{\mathbf{z}}_k, \mathbf{U})/\tau_k). \tag{48}$$

C. Semantic-Aware Masked Motion Modeling (SAM^3)

To achieve more accurate extraction of motion information, we propose a semantic-aware masked motion modeling method. Our theoretical analysis reveals that contrastive learning may lead to overfitting to shared information between views. To mitigate this issue, we aim to extract more non-shared task-relevant information from $\mathbf{x}_q$, thereby augmenting the term $I(\mathbf{z}_q, \mathbf{y}|\mathbf{z}_k)$. However, it's important to note that we cannot leverage any downstream task information during training, precluding direct enhancement of $I(\mathbf{z}_q, \mathbf{y}|\mathbf{z}_k)$. Instead, we propose increasing $I(\mathbf{z}_q, \mathbf{A}_{tv})$ as an alternative strategy. This choice is motivated by the notion that actionlets encapsulate information crucial for downstream classification tasks, thus enhancing $I(\mathbf{z}_q, \mathbf{A}_{tv})$ indirectly enhances the model's capacity to extract task-relevant representations.

Semantic-Aware Masking: To intensify the complexity of the reconstruction task, we incorporate actionlets to mask regions within the sequence exhibiting motion. Our approach involves employing Gumbel sampling to select actionlets, which are then applied as masks for the purpose of semantic-aware masking. This strategy contributes to the difficulty of the reconstruction task and ultimately enhances the learning process.

It is formalized as:

$$\hat{\mathbf{x}}_q = \mathbf{x}_q \odot \text{Gumbel}(\mathbf{A}_{tv}), \tag{49}$$

where $\text{Gumbel}(\cdot)$ is the Gumbel-Max.

Masked Motion Modeling: Subsequently, we carry out reconstruction tasks for both the skeleton data of actionlets and the actionlet regions. This approach emphasizes the extraction of information from motion regions, contributing to the effectiveness of our method.

$$\mathcal{L}_{\text{RC}} = \|\mathbf{A}_{tv} - d(f_q(\hat{\mathbf{x}}_q))\|_2^2, \tag{50}$$

where $d(\cdot)$ is reconstruction network.

V. Experiment Results

A. Datasets and Settings

• *NTU RGB+D Dataset 60 (NTU 60) [23]:* This dataset consists of 56,578 videos capturing 60 different action labels. Each body is represented with the positions of 25 joints in the videos, covering both interactions involving pairs and individual actions.

• *NTU RGB+D Dataset 120 (NTU 120) [24]:* As an expansion of NTU 60, NTU 120 is the most extensive dataset for human action understanding. It comprises 114,480 videos with 120 distinct action categories. The dataset includes actions performed by 106 subjects across various settings, utilizing 32 different recording setups.

• *PKU Multi-Modality Dataset (PKUMMD) [25]:* This dataset focuses on multi-modality 3D understanding of human actions. It contains 52 action categories with almost 20,000 instances. Each sample includes 25 joints. The dataset is divided into two parts, with Part II presenting more challenging data due to increased view variation, leading to additional skeleton noise.

To optimize our network, we utilize the Adam optimizer [72]. The training process is performed on a single NVIDIA TitanX GPU with a batch size set to 128, and the network is trained for 300 epochs. During training, all skeleton sequences are temporally down-sampled to 50 frames to facilitate network training. The encoder $f(\cdot)$ is constructed based on ST-GCN [73], with hidden channels set to a size of 16, which is a quarter of the original model's size. The projection heads for both contrastive learning and auxiliary tasks consist of multilayer perceptrons, projecting features from 256 dimensions to 128 dimensions. We set the temperature parameter τ_q to 0.1 and τ_k to 0.04 to control the sensitivity of the contrastive loss function. For evaluation purposes, we utilize a fully connected layer $\phi(\cdot)$, enabling effective assessment of the model's performance on downstream tasks.

B. Evaluation and Comparison

For a thorough evaluation, we compare our method with other approaches in diverse settings.

1) Linear Evaluation: In the linear evaluation protocol, we employ a fixed encoder $f(\cdot)$ to process the extracted features, and a linear classifier $\phi(\cdot)$ is used for action classification. The evaluation metric employed is the accuracy of action recognition. Notably, the encoder $f(\cdot)$ remains constant throughout the evaluation.

Compared to other approaches outlined in Tables I, III, and IV, our method demonstrates superiority on these datasets. It is worth noting that the transformation strategies utilized by 3s-CrosSCLR [70] and 3s-AimCLR [16] for the contrastive training are uniform across spatial-temporal regions, leading

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

TABLE I
COMPARISON OF ACTION RECOGNITION RESULTS ACHIEVED WITH UNSUPERVISED LEARNING APPROACHES ON THE NTU DATASET

Models	Stream	NTU 60 xview	NTU 60 xsub	NTU 120 xset	NTU 120 xsub
AimCLR [16]	joint	79.7	74.3	63.4	63.4
ActCLR [26]	joint	86.7	80.9	70.5	69.0
ActCLR+	joint	**88.2**	**82.3**	**73.2**	**70.9**
AimCLR [16]	motion	70.6	66.8	54.4	57.3
ActCLR [26]	motion	84.4	78.6	67.8	68.3
ActCLR+	motion	**85.8**	**79.9**	**71.2**	**69.4**
AimCLR [16]	bone	77.0	73.2	63.4	62.9
ActCLR [26]	bone	85.0	80.1	68.2	67.8
ActCLR+	bone	**87.2**	**82.3**	**73.1**	**72.4**
3s-AimCLR [16]	joint+motion+bone	83.8	78.9	68.8	68.2
3s-ActCLR [26]	joint+motion+bone	88.8	84.3	75.7	74.3
3s-ActCLR+	joint+motion+bone	**90.7**	**85.7**	**78.7**	**76.7**

TABLE II
COMPARATIVE ANALYSIS OF ACTION RECOGNITION PERFORMANCE BETWEEN THE PROPOSED METHOD AND OTHER SUPERVISED LEARNING APPROACHES ON THE NTU DATASET

Models	Params	NTU 60 xview	NTU 60 xsub	NTU 120 xset	NTU 120 xsub
Single-stream:					
CPM [49]	0.84M	91.1	84.8	78.9	78.4
RVTCLR+ [63]	0.84M	91.3	84.4	78.4	77.2
ActCLR [26]	0.84M	91.2	85.8	80.3	79.4
ActCLR+	0.84M	**92.0**	**86.1**	**80.5**	**80.0**
Three-stream:					
3s-SkeleMixCLR [64]	2.55M	93.9	87.8	81.2	81.6
3s-RVTCLR+ [63]	2.55M	93.9	87.5	83.4	82.0
3s-ActCLR [26]	2.52M	93.9	88.2	84.6	82.1
3s-ActCLR+	2.52M	**94.2**	**89.0**	**84.8**	**82.2**

to interference with motion information. Conversely, our approach adopts a motion-adaptive data transformation strategy. Consequently, our method extracts features that retain more robust action-related information, making them better suited for downstream tasks.

2) Supervised Finetuning: In our approach, we adopt a two-step process: initially, the encoder $f(\cdot)$ is pretrained following the self-supervised learning framework, and subsequently, the entire network is fine-tuned. Both the encoder $f(\cdot)$ and classifier $\phi(\cdot)$ are trained with the complete training set.

Table II showcases the performance of action recognition on the NTU datasets, demonstrating that our method extracts the requisite information for action understanding, leading to enhanced action recognition accuracy. Particularly noteworthy is our model's superior performance in comparison to state-of-the-art supervised learning methods.

3) Transfer Learning: In the transfer learning scenario, we explore the generalization ability of our model by employing self-supervised task pretraining on the source data. Subsequently, we assess the model's performance on the target dataset using the linear evaluation mechanism, where the parameters of the encoder $f(\cdot)$ remain fixed.

Our method exhibits superior performance, as illustrated in Table V. By leveraging Motion-Adaptive Transformation Strategies (MATS) to eliminate irrelevant information and preserve downstream task-relevant data, our encoder $f(\cdot)$ demonstrates stronger generalization capabilities. The use of MATS contributes to the enhanced performance observed in the transfer learning evaluation.

4) KNN Evaluation: In the K-Nearest Neighbors (KNN) evaluation setup, where the fixed encoder $f_q(\cdot)$ extracts features without any trainable parameters, our model showcases superiority in the accuracy of action recognition on the presented datasets. Table VI highlights the effectiveness of our mothod compared to others in this evaluation.

5) Semi-Supervised Learning: Table VII showcases the accuracy of action recognition results on the NTU datasets. Remarkably, our method surpasses state-of-the-art supervised learning approaches, affirming its efficacy in enhancing action recognition by extracting essential information for downstream tasks. Particularly noteworthy is the substantial improvement over SkeleMixCLR, with an impressive increase of 16.9% on xview and 14.6% on xsub for the NTU 60 dataset, achieved with merely 1% of training samples. This signifies a remarkable advancement in the semi-supervised setting.

6) FLOPS and Params Results: We have conducted an estimation of the space and computational complexities of the proposed model, as detailed in Table VIII. Notably, the reported results pertain to the pretraining stage with a batch size of 128. Recent work [54], [74] has increasingly adopted architectures

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

TABLE III
COMPARING ACTION RECOGNITION PERFORMANCE WITH VARIOUS UNSUPERVISED LEARNING APPROACHES ON THE NTU 60 DATASET

Models	Architecture	xview	xsub
Single-stream:			
GL-Transformer [14]	Transformer	83.8	76.3
CPM [49]	GCN	84.9	78.7
RVTCLR+ [63]	GCN	79.1	74.7
Colorization [65]	DGCNN	82.6	73.2
CMD [48]	GRU	86.9	79.4
HaLP [51]	GRU	86.8	79.7
DMMG [66]	GCN	87.1	82.1
ActCLR [26]	GCN	86.7	80.9
ActCLR+	GCN	**88.2**	**82.3**
Three-stream:			
3s-Colorization [65]	DGCNN	87.2	79.1
3s-SkeleMixCLR [64]	GCN	87.1	82.7
3s-CPM [49]	GCN	87.0	83.2
3s-RVTCLR+ [63]	GCN	84.6	79.7
SkeAttnCLR [67]	GCN	86.5	82.0
PSTL [68]	GCN	83.8	79.4
2s-DMMG [66]	GCN	89.3	84.2
3s-ActCLR [26]	GCN	88.8	84.3
3s-ActCLR+	GCN	**90.7**	**85.7**

TABLE IV
COMPARING ACTION RECOGNITION PERFORMANCE WITH VARIOUS UNSUPERVISED LEARNING APPROACHES ON NTU 120 DATASET

Models	Architecture	xset	xsub
Single-stream:			
CMD† [48]	GRU	66.0	65.4
GL-Transformer [14]	Transformer	68.7	66.0
CPM [49]	GCN	69.6	68.7
DMMG [66]	GCN	70.1	69.6
ActCLR [26]	GCN	70.5	69.0
ActCLR+	GCN	**73.2**	**70.9**
Three-stream:			
3s-CrosSCLR [69]	GCN	66.7	67.9
3s-AimCLR [16]	GCN	68.8	68.2
3s-CMD† [48]	GRU	69.6	69.1
3s-SkeleMixCLR [64]	GCN	70.7	70.5
3s-CPM [49]	GCN	74.0	73.0
3s-RVTCLR+ [63]	GCN	68.9	68.0
2s-DMMG [66]	GCN	72.4	72.7
3s-Colorization [65]	DGCNN	70.8	69.2
3s-ActCLR [26]	GCN	75.7	74.3
3s-ActCLR+	GCN	**78.7**	**76.7**

† indicates that results reproduced on our settings of feature dimension size.

like Transformers, which have demonstrated strong performance across various tasks due to their powerful self-attention mechanisms and ability to model long-range dependencies effectively. However, this comes at the cost of significantly higher computational demands and an increased number of parameters. Such resource-intensive models can pose limitations in practical applications.

In contrast, our approach is designed to be much more lightweight. Moreover, we emphasize that our work is primarily focused on theoretical analysis, which can be applied to various network structures beyond our specific implementation.

TABLE V
COMPARISON OF TRANSFER LEARNING PERFORMANCE WHERE MODELS ARE PRETRAINED ON THE NTU 60 DATASET AND THEN EVALUATED ON THE PKUMMD DATASET USING A LINEAR EVALUATION PROTOCOL

Models	PKU I xview	PKU II xview
3s-AimCLR [16]	85.3	42.4
3s-ActCLR [26]	91.6	44.5
3s-ActCLR+	**93.1**	**51.5**
Models	**PKU I xsub**	**PKU II xsub**
LongT GAN [10]	-	44.8
MS^{2}L [11]	-	45.8
ISC [12]	-	51.1
Hi-TRS [70]	-	55.0
3s-CrosSCLR [69]	-	51.3
3s-AimCLR [16]	85.6	51.6
3s-ActCLR [26]	90.0	55.9
3s-ActCLR+	**91.6**	**62.1**

TABLE VI
COMPARISON OF ACTION RECOGNITION PERFORMANCE USING KNN EVALUATION, WHERE ONLY THE JOINT STREAM IS UTILIZED

Models	NTU 60 xview	NTU 60 xsub
AimCLR [16]	71.0	63.7
SkeleMixCLR [64]	72.3	65.5
ActCLR [26]	78.0	66.6
ActCLR+	**81.6**	**75.9**
Models	**NTU 120 xset**	**NTU 120 xsub**
AimCLR [16]	48.9	47.3
SkeleMixCLR [64]	49.3	48.3
ActCLR [26]	52.6	49.0
ActCLR+	**62.3**	**58.9**

TABLE VII
COMPARISON OF ACTION RECOGNITION RESULTS USING SEMI-SUPERVISED LEARNING APPROACHES ON THE NTU 60 DATASET

Models	xview	xsub
1%:		
3s-CrosSCLR [69]	50.0	51.1
3s-AimCLR [16]	54.3	54.8
3s-SkeleMixCLR [64]	56.2	55.9
3s-C-F Masked Colorization [65]	53.1	52.3
3s-ActCLR [26]	65.6	64.8
3s-ActCLR+	**73.1**	**70.5**
10%:		
3s-CrosSCLR [69]	77.8	74.4
3s-AimCLR [16]	81.6	78.2
3s-SkeleMixCLR [64]	84.7	81.3
3s-C-F Masked Colorization [65]	81.3	76.5
3s-ActCLR [26]	85.8	81.7
3s-ActCLR+	**86.0**	**82.2**

C. Analysis of Actionlet

Quantification of Actionlet: To provide a more explicit understanding, we delve into the specific quantitative metrics to evaluate the quality of actionlets. These metrics include:

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

TABLE VIII
FLOPS AND PARAMS RESULTS OF DIFFERENT MODELS

Models	Params ↓	FLOPs ↓	xview	xsub
GL-Transformer [14]	214M	59.35G	83.8	76.3
3s-CMD [48]	99M	17.32G	90.9	84.1
MAMP [53]	8.7M	5.46G	89.1	84.9
3s-UmUR [73]	64M	5.22G	**91.4**	84.4
S-JEPA [55]	15M	14.92G	89.8	85.3
Ours	**2.5M**	**3.08G**	90.7	**85.7**

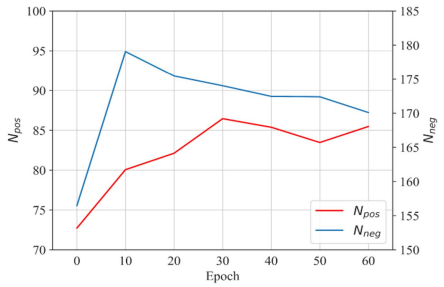


Fig. 8. Curve of the number of correct actionets and the number of incorrect actionlets with epoch.

• *Correct Actionlets Count:* This metric quantifies the number of actionlets that are accurately identified.

$$N_{\text{pos}} = \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{v=1}^{V} \mathbf{A}_{tv}^{i} \cap \hat{\mathbf{A}}_{tv}^{i}, \tag{51}$$

where $\hat{\mathbf{A}}_{tv}^{i}$ is the actionlets obtained through the supervisor method. We utilize it as the ground truth reference.

• *Incorrect Actionlets Count:* This metric measures the number of actionlets that are wrongly identified.

$$N_{\text{neg}} = \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{v=1}^{V} \mathbf{A}_{tv}^{i} \cap \sim \hat{\mathbf{A}}_{tv}^{i}, \tag{52}$$

where $\sim$ represents the complement of a set.

• *Percentage of Correct Actionlets:* This metric expresses the proportion of correctly identified actionlets relative to the total number of actionlets.

$$R = \frac{N_{\text{pos}}}{N_{\text{pos}} + N_{\text{neg}}}. \tag{53}$$

With these specific metrics, we gain a comprehensive and precise insight into the contrastive learning process.

1) Dynamic Processes of Actionlet: By closely examining the alterations in actionlets throughout the training process and across various data transformation intensities, we discern several notable findings.

Throughout the training process of contrastive learning, the network goes through distinct phases, as shown in Fig. 8. Initially, it extensively explores regions that display potential motion. As training progresses, it consistently removes areas that prove to be irrelevant to the task. By constructing information bottlenecks through data transformations, the network filters out task-irrelevant incorrect actionlets, and obtains the maximum

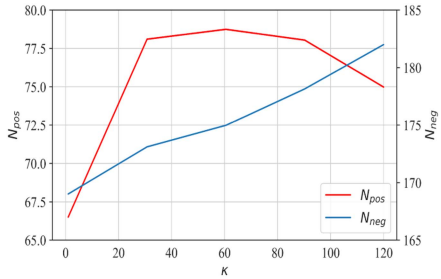


Fig. 9. Number of correct actionlets and number of incorrect actionlets with data transformation.

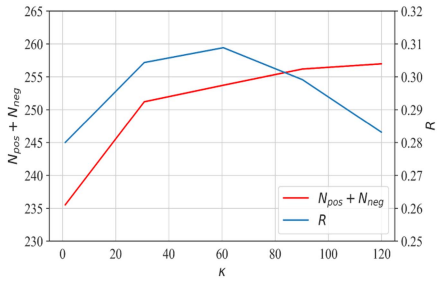


Fig. 10. The number of actionlets and the proportion of correct actionlets vary with data transformation.

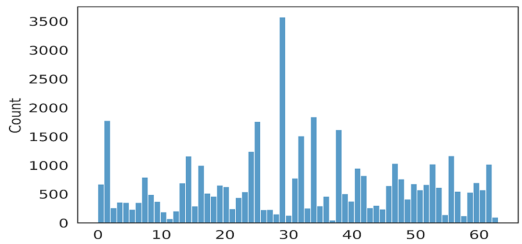


Fig. 11. The distribution of local features. Some of these features are heavily clustered in some of the clustering centres.

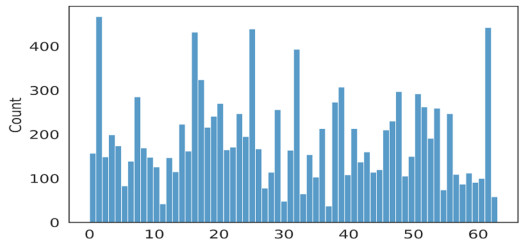


Fig. 12. The distribution of local features in the actionlet region. The features are more uniformly distributed across the clustering centres.

proportion of correct actionlets with minimal sufficient transformation, as shown in Figs. 9 and 10.

2) Analysis of Actionlet and Non-Actionlet Semantic Decoupling: In Fig. 13, we evaluate our model's performance by extracting information exclusively from the actionlet region or non-actionlet region and reporting action recognition performance. The accuracy in the actionlet region is always greater than in the non-actionlet region. This indicates that the actionlet

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

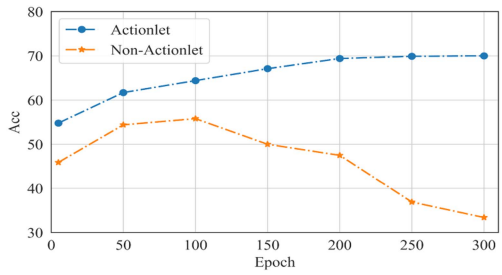


Fig. 13. Action recognition accuracy of actionlet regions and non-actionlet regions.

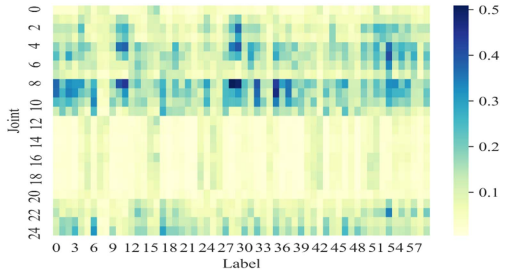


Fig. 14. Heat map of skeleton joints and action labels. $[20, 3, 2, 1, 0]$ are trunk indexes, $[8, 9, 10, 11, 23, 24]$ are left hand indexes, $[4, 5, 6, 7, 21, 22]$ are right hand indexes, $[16, 17, 18, 19]$ are left leg indexes and $[12, 13, 14, 15]$ are right leg indexes.

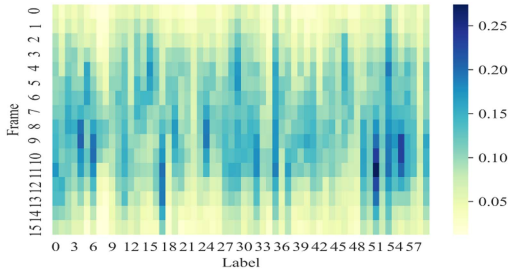


Fig. 15. Heat map of skeleton frames and action labels. We perform 4-fold downsampling in the temporal domain.

regions extracted by our method contain more recognition-related information. Whereas the accuracy of non-actionlet regions first rises and then falls, we believe this is because actionlet mining keeps getting more accurate as the training progresses.

3) Analysis of Frequency of Actionlet and Local Feature: We counted the frequency of local features and found an unbalanced distribution in Fig. 11. Most of the features are frequently occurring features that are closer to the average feature and less divisible because of their extensive interconnections. The local features of actionlet are those that occur relatively infrequently, while filtering out frequently occurring features that cannot be helpful for classification, as shown in Fig. 12.

4) Analysis of Actionlet and Action Label: To analyze the relationship between actionlets and action labels, we conducted an investigation into the actionlet regions selected for each category, separately in the spatial and temporal dimensions, as illustrated in Figs. 14 and 15. In the spatial dimension, it becomes evident that the importance of different joints varies. Notably, hand movements are frequently selected, suggesting that a majority of actions involve the participation of the hands. In the temporal dimension, actions tend to occur predominantly in the middle of the sequence. This suggests that multiple actionlet regions may exist for the same action label, each representing different facets or modes of that action.

TABLE IX
ANALYSIS OF MODULE COMBINATIONS ON NTU 60 XVIEW DATASET WITH THE JOINT STREAM

Module			KNN	Linear
Intra-SD	Inter-SD	SAM^3		
✓			78.5	87.0
✓	✓		81.4	87.7
✓		✓	80.3	87.1
✓	✓	✓	**81.8**	**88.2**

"SD" means similarity distillation.

TABLE X
ANALYSIS OF MOTION-ADAPTIVE DATA TRANSFORMATION ON NTU 60 XVIEW DATASET WITH THE JOINT STREAM

Transformation	Region	KNN	Linear
Adversarial Noise	*w/* Actionlet	**81.8**	**88.2**
	w/o Actionlet	74.2	83.0
Skeleton Mask	*w/* Actionlet	**81.8**	**88.2**
	w/o Actionlet	73.2	82.0
Masking Strategy	*w/* MAM	79.8	87.3
	w/ SAM	**81.8**	**88.2**

D. Ablation Study

Next, we provide more detailed analyses of our proposed approach by conducting extensive ablation experiments.

1) Analysis of Module Combination: We explore the performance of different combinations of modules and observe that each module contributes to a certain degree of improvement. Optimal performance is achieved when all three modules are combined. As shown in Table IX, each module improves performance. Intra-SD (intra-data similarity distillation) enhances feature learning within data, Inter-SD (inter-data similarity distillation) enriches knowledge exchange across data, and SAM^3 (fine-grained reconstruction from a joint perspective) further refines feature reconstruction.

2) Analysis of Motion-Adaptive Data Transformation: Data transformation plays a crucial role in contrastive learning, and thus we conducted experiments to evaluate the impact of motion-adaptive data transformations on action recognition accuracy under various scenarios.

As shown in Table X, motion-adaptive transformations consistently outperform full-region transformations (involving the entire skeleton data) across different noise settings. This observation emphasizes the robustness of our design to variations in data transformations. Compared with motion-aware masking (MAM) strategy [54], SAM strategy slightly outperforms MAM in terms of accuracy, particularly in actions involving complex

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

TABLE XI
ANALYSIS OF DATA TRANSFORMATION COMBINATIONS ON NTU 60 XVIEW DATASET

Module				KNN	Linear
$\mathcal{T}_{act}$	$\mathcal{T}_{non}$	Adversarial Noise	Skeleton Mask		
				67.5	79.9
✓				73.6	83.2
✓	✓			79.6	86.4
✓	✓	✓		80.1	86.9
✓	✓		✓	80.8	87.7
✓	✓	✓	✓	**81.8**	**88.2**

$\mathcal{T}_{act}$ is actionlet transformations. $\mathcal{T}_{non}$ is non-actionlet transformations, excluding Adversarial Noise and Skeleton Mask.

semantic relationships between joints. This suggests that SAM is more effective in understanding the semantic context.

To further understand the impact of different data transformation combinations on the effect of contrastive learning, we assessed the accuracy of action recognition under various scenarios, as presented in Table XI. The results highlight that enhancing the consistency of the feature space through multiple data transformations leads to improved performance in downstream tasks.

VI. CONCLUSION

In this research, we propose an innovative actionlet-dependent contrastive learning method. Utilizing actionlets, we devise motion-adaptive data transformations to efficiently segregate action and non-action regions. The proposed modules enable focused attention on motion information within the sequence while reducing disturbances from static regions for represntation extraction. This approach preserves action movement within actionlet while incorporating richer motion patterns, resulting in more compact and informative learned features. Additionally, the similarity mining loss further regulates the representation space.

REFERENCES

[1] X. Zhang, C. Xu, and D. Tao, "Context aware graph convolution for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14321–14330.

[2] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang, "Disentangling and unifying graph convolutions for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 140–149.

[3] P. Zhang, C. Lan, W. Zeng, J. Xing, J. Xue, and N. Zheng, "Semantics-guided neural networks for efficient skeleton-based human action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1109–1118.

[4] Y. Song, Z. Zhang, C. Shan, and L. Wang, "Stronger, faster and more explainable: A graph convolutional baseline for skeleton-based action recognition," in *Proc. ACM Int. Conf. Multimedia*, 2020, pp. 1625–1633 .

[5] W. Peng, X. Hong, H. Chen, and G. Zhao, "Learning graph convolutional network for skeleton-based human action recognition by neural searching," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 2669–2676.

[6] K. Su, X. Liu, and E. Shlizerman, "PREDICT & cluster: Unsupervised skeleton based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9631–9640.

[7] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional LSTM network for skeleton-based action recognition," in *Proc. Int. Conf. Signal Process. Mach. Learn.*, 2019, pp. 1227–1236.

[8] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12026–12035.

[9] 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 *Proc. Int. Conf. Comput. Vis.*, 2021, pp. 13359–13368.

[10] 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 *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 2644–2651.

[11] L. Lin, S. Song, W. Yang, and J. Liu, "MS2L: Multi-task self-supervised learning for skeleton based action recognition," in *Proc. ACM Int. Conf. Multimedia*, 2020, pp. 2490–2498.

[12] F. M. Thoker, H. Doughty, and C. G. Snoek, "Skeleton-contrastive 3D action representation learning," in *Proc. ACM Int. Conf. Multimedia*, 2021, pp. 1655–1663.

[13] S. Yang, J. Liu, S. Lu, M. H. Er, and A. C. Kot, "Skeleton cloud colorization for unsupervised 3D action representation learning," in *Proc. Int. Conf. Comput. Vis.*, 2021, pp. 13423–13433.

[14] B. Kim, H. J. Chang, J. Kim, and J. Y. Choi, "Global-local motion transformer for unsupervised skeleton-based action learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 209–225.

[15] H. Rao, S. Xu, X. Hu, J. Cheng, and B. Hu, "Augmented skeleton based contrastive action learning with momentum LSTM for unsupervised action recognition," *Inf. Sci.*, vol. 569, pp. 90–109, Aug. 2021.

[16] 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," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 762–770.

[17] Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, and P. Isola, "What makes for good views for contrastive learning?," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 6827–6839.

[18] J. Robinson, L. Sun, K. Yu, K. Batmanghelich, S. Jegelka, and S. Sra, "Can contrastive learning avoid shortcut solutions?," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 4974–4986.

[19] T. Wang and P. Isola, "Understanding contrastive representation learning through alignment and uniformity on the hypersphere," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 9929–9939.

[20] Y. Wang, Q. Zhang, Y. Wang, J. Yang, and Z. Lin, "Chaos is a ladder: A new theoretical understanding of contrastive learning via augmentation overlap," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–15.

[21] Y. Tian, X. Chen, and S. Ganguli, "Understanding self-supervised learning dynamics without contrastive pairs," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 10268–10278.

[22] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1290–1297.

[23] A. Shahroudy, J. Liu, T. T. Ng, and G. Wang, "NTU RGB D: A large scale dataset for 3D human activity analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1010–1019.

[24] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2684–2701, Oct. 2020.

[25] J. Liu, S. Song, C. Liu, Y. Li, and Y. Hu, "A benchmark dataset and comparison study for multi-modal human action analytics," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 16, no. 2, pp. 41:1–41:24, 2020.

[26] L. Lin, J. Zhang, and J. Liu, "Actionlet-dependent contrastive learning for unsupervised skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 2363–2372.

[27] J. Shotton et al., "Real-time human pose recognition in parts from single depth images," *Commun. ACM*, vol. 56, no. 1, pp. 116–124, 2013.

[28] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 588–595.

[29] Y. Goutsu, W. Takano, and Y. Nakamura, "Motion recognition employing multiple kernel learning of fisher vectors using local skeleton features," in *Proc. Int. Conf. Mach. Learn. Workshops*, 2015, pp. 79–86.

[30] L. Tao and R. Vidal, "Moving poselets: A discriminative and interpretable skeletal motion representation for action recognition," in *Proc. Int. Conf. Mach. Learn. Workshops*, 2015, pp. 61–69.

[31] P. Wang, C. Yuan, W. Hu, B. Li, and Y. Zhang, "Graph based skeleton motion representation and similarity measurement for action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 370–385.

[32] C. Si, Y. Jing, W. Wang, L. Wang, and T. Tan, "Skeleton-based action recognition with spatial reasoning and temporal stack learning," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 103–118.

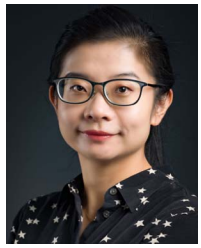
Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

[33] Y.-F. Song, Z. Zhang, C. Shan, and L. Wang, "Constructing stronger and faster baselines for skeleton-based action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 1474–1488, Feb. 2023.
[34] H. Shi, W. Peng, H. Chen, X. Liu, and G. Zhao, "Multiscale 3D-shift graph convolution network for emotion recognition from human actions," *IEEE Intell. Syst.*, vol. 37, no. 4, pp. 103–110, Jul./Aug. 2022.
[35] Y. Zhou, X. Yan, Z.-Q. Cheng, Y. Yan, Q. Dai, and X.-S. Hua, "BlockGCN: Redefine topology awareness for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 2049–2058.
[36] W. Wu, C. Zheng, Z. Yang, C. Chen, S. Das, and A. Lu, "Frequency guidance matters: Skeletal action recognition by frequency-aware mixed transformer," in *Proc. ACM Int. Conf. Multimedia*, 2024, pp. 4660–4669.
[37] J. Do and M. Kim, "Skateformer: Skeletal-temporal transformer for human action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2025, pp. 401–420.
[38] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," 2019, *arXiv: 1906.05849*.
[39] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–15.
[40] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2536–2544.
[41] M. Noroozi, A. Vinjimoor, P. Favaro, and H. Pirsiavash, "Boosting self-supervised learning via knowledge transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9359–9367.
[42] S. Purushwalkam and A. Gupta, "Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 3407–3418.
[43] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 776–794.
[44] Y. Yao, C. Liu, D. Luo, Y. Zhou, and Q. Ye, "Video playback rate perception for self-supervised spatio-temporal representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6548–6557.
[45] S. Jenni, G. Meishvili, and P. Favaro, "Video representation learning by recognizing temporal transformations," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 425–442.
[46] G. Lorre, J. Rabarisoa, A. Orcesi, S. Ainouz, and S. Canu, "Temporal contrastive pretraining for video action recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2020, pp. 662–670.
[47] D. Kim, D. Cho, and I. S. Kweon, "Self-supervised video representation learning with space-time cubic puzzles," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 8545–8552.
[48] N. Rai, E. Adeli, K.-H. Lee, A. Gaidon, and J. C. Niebles, "CoCon: Cooperative-contrastive learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 3384–3393.
[49] Y. Mao, W. Zhou, Z. Lu, J. Deng, and H. Li, "CMD: Self-supervised 3D action representation learning with cross-modal mutual distillation," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 734–752.
[50] H. Zhang, Y. Hou, W. Zhang, and W. Li, "Contrastive positive mining for unsupervised 3D action representation learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 36–51.
[51] X. Liu, H. Shi, H. Chen, Z. Yu, X. Li, and G. Zhao, "Imigue: An identity-free video dataset for micro-gesture understanding and emotion analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10631–10642.
[52] A. Shah et al., "Halp: Hallucinating latent positives for skeleton-based self-supervised learning of actions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 18846–18856.
[53] L. Lin, J. Zhang, and J. Liu, "Mutual information driven equivariant contrastive learning for 3D action representation learning," *IEEE Trans. Image Process.*, vol. 33, pp. 1883–1897, 2024.
[54] Y. Mao, J. Deng, W. Zhou, Y. Fang, W. Ouyang, and H. Li, "Masked motion predictors are strong 3D action representation learners," in *Proc. Int. Conf. Comput. Vis.*, 2023, pp. 10181–10191.
[55] L. Lin, L. Wu, J. Zhang, and J. Liu, "Idempotent unsupervised representation learning for skeleton-based action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2025, pp. 75–92.
[56] M. Abdelfattah and A. Alahi, "S-JEPA: A joint embedding predictive architecture for skeletal action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2025, pp. 367–384.
[57] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
[58] J. Z. HaoChen, C. Wei, A. Gaidon, and T. Ma, "Provable guarantees for self-supervised deep learning with spectral contrastive loss," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 5000–5011.
[59] T. Wang, Z. Dou, C. Bao, and Z. Shi, "Diffusion mechanism in residual neural network: Theory and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 2, pp. 667–680, Feb. 2024.
[60] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2014, *arXiv:1412.6572*.
[61] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "CutMix: Regularization strategy to train strong classifiers with localizable features," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 6022–6031.
[62] S. Ren et al., "A simple data mixing prior for improving self-supervised learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 14575–14584.
[63] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," 2017, *arXiv: 1710.09412*.
[64] Y. Zhu, H. Han, Z. Yu, and G. Liu, "Modeling the relative visual tempo for self-supervised skeleton-based action recognition," in *Proc. Int'l Conf. Comput. Vis.*, 2023, pp. 13913–13922.
[65] Z. Chen, H. Liu, T. Guo, Z. Chen, P. Song, and H. Tang, "Contrastive learning from spatio-temporal mixed skeleton sequences for self-supervised skeleton-based action recognition," 2022, *arXiv:2207.03065*.
[66] S. Yang, J. Liu, S. Lu, E. M. Hwa, Y. Hu, and A. C. Kot, "Self-supervised 3D action representation learning with skeleton cloud colorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 1, pp. 509–524, Jan. 2024.
[67] S. Guan, X. Yu, W. Huang, G. Fang, and H. Lu, "DMMG: Dual min-max games for self-supervised skeleton-based action recognition," *IEEE Trans. Image Process.*, vol. 33, pp. 395–407, 2024.
[68] Y. Hua et al., "Part aware contrastive learning for self-supervised action recognition," in *Proc. Int. Joint Conf. Artif. Intell.*, 2023, Art. no. 95.
[69] Y. Zhou, H. Duan, A. Rao, B. Su, and J. Wang, "Self-supervised action representation learning from partial spatio-temporal skeleton sequences," 2023, *arXiv:2302.09018*.
[70] L. Li, M. Wang, B. Ni, H. Wang, J. Yang, and W. Zhang, "3D human action representation learning via cross-view consistency pursuit," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4741–4750.
[71] Y. Chen et al., "Hierarchically self-supervised transformer for human skeleton representation learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 185–202.
[72] W. K. Newey, "Adaptive estimation of regression models via moment restrictions," *J. Econometrics*, vol. 38, no. 3, pp. 301–339, 1988.
[73] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2018. pp. 205–217.
[74] S. Sun et al., "Unified multi-modal unsupervised representation learning for skeleton-based action understanding," in *Proc. ACM Int. Conf. Multimedia*, 2023, pp. 2973–2984.

Lilang Lin (Graduate Student Member, IEEE) received the BS degree in data science, in 2021, from Peking University, Beijing, China, where he is currently working toward the PhD degree with the Wangxuan Institute of Computer Technology. His current research interests include action recognition, self-supervised learning, and unsupervised learning.

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.

Jiahang Zhang received the BS degree in computer science, in 2023, from Peking University, Beijing, China, where he is currently working toward the PhD degree with the Wangxuan Institute of Computer Technology. His current research interests include action recognition and self-supervised learning.



Jiaying Liu (Fellow, IEEE) received the PhD degree (Hons.) in computer science from Peking University, Beijing, China, in 2010. She is currently a professor with the Wangxuan Institute of Computer Technology, Peking University, China. She has authored more than 100 technical articles in refereed journals and proceedings, and holds 70 granted patents. Her current research interests include multimedia signal processing, compression, and computer vision. She is a senior member of IEEE/CSIG, and a distinguished member of CCF. She was a visiting scholar with the University of Southern California, Los Angeles, California, from 2007 to 2008. She was a visiting researcher with Microsoft Research Asia, in 2015 supported by the Star Track Young Faculties Award. She has served as a member of Multimedia Systems and Applications Technical Committee (MSA TC), and Visual Signal Processing and Communications Technical Committee (VSPC TC) in IEEE Circuits and Systems Society. She received the IEEE ICME 2020 Best Paper Award and IEEE MMSP 2015 Top10% Paper Award. She has also served as the associate editor of the *IEEE Transactions on Image Processing*, the *IEEE Transactions on Circuits Systems for Video Technology* and *Journal of Visual Communication and Image Representation*, the Technical Program Chair of ACM MM Asia-2023/IEEE ICME-2021/ACM ICMR-2021/IEEE VCIP-2019, the Area Chair of CVPR-2021/ECCV-2020/ICCV-2019, ACM ICMR Steering Committee member and the CAS Representative at the ICME Steering Committee. She was the APSIPA Distinguished Lecturer (2016-2017).

Authorized licensed use limited to: Peking University. Downloaded on October 18,2025 at 09:50:16 UTC from IEEE Xplore. Restrictions apply.