Self-Supervised Representation Learning for Skeleton-Based Group Activity Recognition

Cunling Bian
College of Intelligence and Computing, Tianjin University
China
clbian@tju.edu.cn

Wei Feng*
College of Intelligence and Computing, Tianjin University
China
wfeng@tju.edu.cn

Song Wang*
University of South Carolina
Columbia, USA
songwang@cec.sc.edu

ABSTRACT

Group activity recognition (GAR) is a challenging task for discerning the behavior of a group of actors. This paper aims at learning discriminative representation for GAR in a self-supervised manner based on human skeletons. As modeling relations between actors lie at the center of GAR, we propose a valid self-supervised learning pretext task with a matching framework, where a representation model is driven to identify subgroups in a synthetic group based on actors' skeleton sequences. For backbone networks, while spatial-temporal graph convolution networks have dominated the skeleton-based action recognition, they under-explore the group relevant interactions among actors. To address this issue, we come up with a novel plug-in Actor-Association Graph Convolution Module (AAGCM) based on inductive graph convolution, which can be integrated into many common backbones. It can not only model the interactions at different levels but also adapt to variable group sizes. The effectiveness of our approaches is demonstrated by extensive experiments on three benchmark datasets: Volleyball, Collective Activity, and Mutual NTU. Code is available at https://github.com/xiaochehe/SSL_Skeleton_GAR.

CCS CONCEPTS

• **Computing methodologies** → **Activity recognition and understanding**.

KEYWORDS

self-supervised representation learning, group activity recognition, spatial-temporal graph convolution network, skeleton

ACM Reference Format:
Cunling Bian, Wei Feng, and Song Wang. 2022. Self-Supervised Representation Learning for Skeleton-Based Group Activity Recognition. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3503161.3547822

*Corresponding authors: W. Feng and S. Wang

MM '22, October 10–14, 2022, Lisboa, Portugal

ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3547822

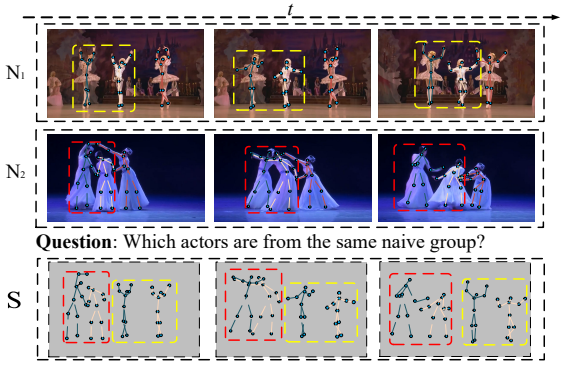

Figure 1: An illustration of the main idea of the proposed self-supervised representation learning task. Given skeleton sequences of two *Naive Groups* N_1 and N_2, subsets of actors from different naive groups, marked with different colored dotted boxes, are randomly sampled to construct a *Synthetic Group* S. The naive groups consist of actors with some kind of relations between each other. The relations can vary from body interaction to spatial distribution. The task is *Subgroup Identification* from the synthetic group based on actors' skeleton sequences.

1 INTRODUCTION

Group activity recognition (GAR), which tries to discern the activities involving a large number of interactive individuals, has a variety of applications in video understanding, such as sports analysis, crowd monitoring, and crime prevention [46, 47]. Unlike conventional action recognition, GAR requires both modeling the spatiotemporal dynamics of individual actors and understanding the group relevant interactions among them.

In the past few years, in order to obtain training samples for GAR, annotators usually need to watch through the lengthy video, manually localize those positive frames, and provide group activity labels and sometimes even individual action labels for each actor, which is extremely laborious and costly [8, 15]. With the booming development of self-supervised learning in artificial intelligence, it is highly desirable to explore alternative, cheap, and yet often noisy and indirect supervision signals, without any human annotations. This motivates us to study non-manual supervision signals for self-supervised representation learning for GAR.

Apart from that, previous methods mainly focus on image visual information, which mostly reasons about the group relation at high levels [25, 46]. Recently researchers try to complement or even replace image visual information with a skeleton representing the human body in the form of a set of coordinate positions for

key joints, which may lead to better performance [11, 14]. Significantly, not only the skeleton is robust to inconsistent appearances and different environments, but also the joint-part-body hierarchical architecture of the skeleton makes it easier to reason about the individuals' interactions in multi-levels. At present skeleton is still underutilized in GAR. Usually, it is only taken as an extra/alternative stream for isolated individual feature extraction. An important reason for this situation is the lack of an effective and generic network to handle skeletons of multi-actors. Existing spatial-temporal graph convolution networks, which have achieved state-of-the-art results on many skeleton-based action analysis tasks, simply average individuals' features before classification in multi-actor scenarios [33, 48]. In the stage of feature extraction, almost no consideration is given to modeling interactions among actors. Therefore, they lack the flexibility in dealing with more complex scenarios of group activities.

In this paper, we propose a self-supervised representation learning algorithm for skeleton-based group activity recognition. Specifically, by constructing a synthetic group with randomly sampled subsets of actors from different naive groups, the self-supervised representation learning pretext task is subgroup identification from the synthetic group based on actors' skeleton sequences, as shown in Figure 1. In this way, the representation model is driven to model the spatiotemporal dynamics of individual actors and understand the group relevant interactions among them. Our benchmark is based on the following principle: a good representation transfers with limited supervision and limited fine-tuning. Thus, the representation model is evaluated through linear evaluation protocol and semi-supervised fine-tuning. Meanwhile, we come up with an Actor-Association Graph Convolution Module (AAGCM) based on inductive graph convolution to effectively model the interactions among multi-actors and adapt to variable group sizes. As a general plug-in module, AAGCM can be added to different layers of spatial-temporal graph convolution networks for modeling group-relevant multi-level interactions among actors. Extensive experiments demonstrate that for both the self-supervised representation learning algorithm and the network architecture for skeleton-based group activity recognition, we outperform the existing state-of-the-art results by a sizable margin. The contributions can be summarized as follow:

- To the best of our knowledge, this is the first work to explore self-supervised representation learning for skeleton-based group activity recognition.
- We propose a self-supervised pretext task and a matching framework to learn representations for group activity recognition. By identifying subgroups in a synthetic group based on actors' skeleton sequences, our framework can learn group activity representation in the spatiotemporal domain.
- Based on inductive graph convolution, we propose the AAGCM, which can model the group relevant interactions among actors at different levels and adapt to variable group sizes.

2 RELATED WORK

2.1 Group activity recognition

Group activity recognition aims to automatically identify an activity performed by at least two persons. Therefore, compared with traditional action recognition, it also requires exploring the spatial structures between persons. In the early years, these structures are typically learned based on a combination of handcrafted features with graphical models [1, 7, 9, 22, 23] or AND-OR grammar models [2, 35]. Lan *et al.* [22] jointly captured the group-person and person-person interaction information via a graphical inference framework. Shu *et al.* [35] proposed to jointly group people and recognize events and human roles in aerial videos by an AND-OR spatial-temporal graph formalism.

With the emergence of deep learning, significant performance improvements have been made to group activity recognition. Many deep-learning approaches utilized CNN-RNN type networks, where CNN is used to extract the person-level static features, and RNN can not only capture the temporal dynamics of each individual but also model the interaction context in neighborhoods [3, 36, 37, 44]. For instance, Wang *et al.* [44] developed a hierarchical recurrent framework to handle high order contexts, including single-person dynamics, and intra-group and inter-group interactions. GCN can better deal with graph-structured data, therefore is more suitable to address group activity recognition by treating each person as a node [14]. Wu *et al.* [46] built an actor relation graph using a CNN and a GCN to capture both the appearance and position relations of actors. All previous works only reason about the group relations at high levels. Here we intend to model group relevant multi-level interactions among actors based on the skeleton.

2.2 Skeleton-based action recognition

With excellent performance, spatial-temporal graph convolution networks are widely adopted in skeleton-based action recognition, which models human skeleton sequences as spatial-temporal graphs. Inspired by the booming graph-based methods, Yan *et al.* [48] firstly introduce GCN into the skeleton-based action recognition task. It is a well-known baseline for GCN-based approaches, which models the spatial configurations and temporal dynamics of skeletons synchronously. Upon the baseline, Shi *et al.* [34] propose to either uniformly or individually learned the topology of the graph by the BP algorithm in an end-to-end manner, which significantly improves the generality and accuracy of action recognition. To decline the computational complexity, the study in [6] provides shift graph operations and lightweight point-wise convolutions, where the shift graph operations provide flexible receptive fields for both spatial graph and temporal graph. Despite the great success of spatial-temporal graph convolution network in skeleton-based action recognition, prior works do not address the issue of modeling interactions between actors, which becomes an obstacle to applying them in multi-person scenarios. Therefore, how restructuring spatial-temporal graph convolution networks to satisfy the requirements of group activity recognition is a challenging problem.

2.3 Self-supervised representation learning

Self-supervised representation learning formulates a pretext learning task and leverages large-scale unlabeled data to learn useful representations for various problems. It is first proposed and investigated in the image domain [10]. Various novel pretext tasks have been proposed to learn image representation. Some leverage concept information in images and then construct constraints like

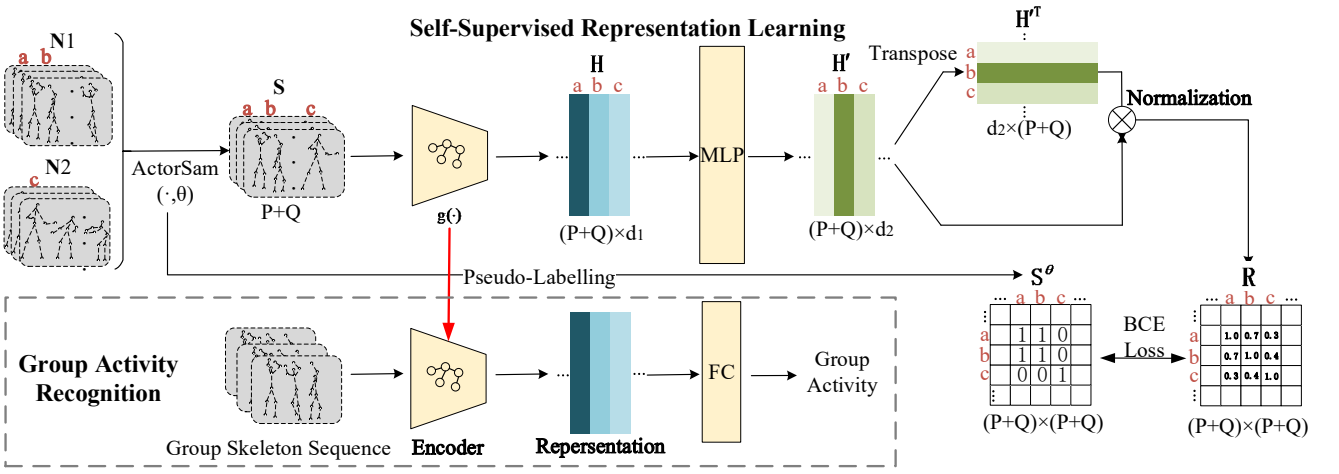

Figure 2: The framework of self-supervised representation learning for group activity recognition. Subsets of actors, e.g., actors a, b, and c, sampled from naive groups N_1 and N_2 construct a synthetic group S. P and Q are the numbers of actors sampled from N_1 and N_2, respectively. The corresponding pseudo-labeling S^θ is a relation matrix associated with the source of actors. $g(\cdot)$ is the target encoder model for skeleton-based group representation learning. It can be any network that can handle skeleton information. In this paper, $g(\cdot)$ is constructed from an improved spatial-temporal graph convolution network based on AAGCM. d_1 and d_2 are the feature dimensions. Combined with a projection head (MLP), the model is trained to predict relations between any two actors in the synthetic group. Binary Cross Entropy (BCE) loss is used for the model optimization.

re-ordering perturbed image patches [29], counting virtual primitives [30], and classifying image rotations [19]. Others reconstruct part of the image, such as image completion [31], colorization [49] and channel prediction [50]. Recently, contrastive learning has attracted much attention in the self-supervised representation learning research [4, 32]. It minimizes the distances of positive pairs and simultaneously maximizes the distances of negative pairs in the latent space, in order to conduct instance discrimination. Inspired by the success of self-supervised image representation learning, many works have emerged to learn transferable representation leveraging rich spatiotemporal context information. Some learn visual features through video generation like VideoGAN [42], colorization [43] and prediction [38]. The inherent temporal information within videos can also be used as a supervision signal, such as temporal order verification [45] and temporal order recognition [17]. Others learn video features from the correspondence of multiple data streams, such as RGB-flow [28], visual-audio [20] and Ego-motion [16]. For self-supervised representation learning for skeleton-based action recognition, the most commonly used non-manual supervision signals are movement forecasting and temporal ordering, which do not well reflect the relations between actors in the group [26, 39]. Thus, we intend to develop a group-relevant interaction-centered self-supervised pretext task for skeleton-based group activity recognition.

3 METHODOLOGY

To learn representation for skeleton-based group activity recognition in a self-supervised manner, a self-supervised pretext task, a matching framework, and an Actor-Association Graph Convolution Module are proposed. Following the work of [48], we also utilize the spatial-temporal graph to construct hierarchical representations of the group skeleton sequences in this paper. Specifically, group skeleton sequence is organized as an undirected spatial-temporal graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{\mathbf{v}_{ti}^m \mid t = 1, ..., T; m = 1, ..., M; i = 1, ..., N\}$ denotes a set of vertices, corresponding to T frames, M actors per frame and N body joints per actor. $\mathcal{E}$ is the set of edges, indicating the connections between nodes. It consists of three parts: temporal connections $\mathcal{E}_{\text{tem}}$, spatial intra-actor connections $\mathcal{E}_{\text{spaintra}}$ and inter-actor connections $\mathcal{E}_{\text{spainter}}$. Specifically, $\mathcal{E}_{\text{tem}}$ links each joint with its counterpart across the neighboring frames. $\mathcal{E}_{\text{spaintra}}$ includes both direct and indirect kinematic dependencies among joints of an actor while $\mathcal{E}_{\text{spainter}}$ relates its joints with all joints of other actors in the same frame.

3.1 Self-supervised subgroup identification

In this paper, we define a *naive group* which consists of actors with some kind of relations between each other. The relations can vary from body interaction to spatial distribution. In this way, naive groups can take many forms. Taking sports events as an example, a naive group can be very small, consisting of only athletes, or very large, consisting of all the athletes, judges, and spectators. In order to meet an unsupervised setting, we take all of the actors, who appear in a sample, as a naive group. Although we do not exactly know the direct and indirect relations between group members, we at least know that there is spatial distribution relation among them. Then, we select subsets of actors in different naive groups and make up a *synthetic group*, where the actors from the same naive group form one subgroup in the synthetic one. The proposed pretext task is subgroup identification in this synthetic group based on actors' skeleton sequences. Note that in order to ensure that there is sufficient differentiation between subgroups, the naive groups are sampled at different time points or scenes. Based on skeleton sequences of the synthetic group, the subgroup identification task can drive the representation model to learn both the spatiotemporal dynamics of individual actors and the relations between them.

Assuming that a synthetic group $\mathbf{S}$ is composed of subsets of actors drawn from two naive groups $\mathbf{N_1}$ and $\mathbf{N_2}$, we can randomly sample P and Q actors from them respectively by a function ActorSam$(\cdot, \theta)$ which selects actors randomly in accordance with a predefined sampling probability θ. In this way, we represent the synthetic group as $\mathbf{S} = \{\mathbf{I}_1, ..., \mathbf{I}_P, ..., \mathbf{I}_{P+Q}\}$, where $\mathbf{I}_i$ is the skeleton sequence of an actor from $\mathbf{N_1}$ or $\mathbf{N_2}$. As for the automatically generated pseudo label $\mathbf{S}^\theta$, in order to make it clearly indicate the group internal structure and be adapted to varied number of subgroups, it is expressed as a relation matrix between actors. In this matrix, the relation between actors belonging to the same subgroup is defined as one state and belonging to the different subgroups is defined as another state. Specifically, the relation matrix is represented as $\mathbf{S}^\theta = [J_{i,j}]$, where $i, j = 1, ..., P+Q$, and $J_{i,j} = 1$ if $\mathbf{I}_i$ and $\mathbf{I}_j$ belong to one same subgroup and else $J_{i,j} = 0$. In this way, the subgroup identification task is ultimately converted to a binary classification problem of relations between actors.

The implementation of the proposed self-learning framework is shown in Figure 2. Multiple skeleton sequences of the synthetic group are organized as a sptial-temporal graph and fed into a graph neural network-based encoder $g(\cdot)$ to extract features. $g(\cdot)$ is the target encoder we learned in this self-supervised setting for the downstream group activity recognition task. It can be any network that can handle skeleton information to obtain group representation $\mathbf{H}$ here. In this paper, $g(\cdot)$ is constructed from a spatial-temporal graph convolution network. Then, we use an MLP with one hidden layer as a neural network projection head to map $\mathbf{H}$ to the space where the loss function is applied. The projected representation is named as $\mathbf{H}'$. Subsequently, we obtain the predicted group actor relation matrix $\mathbf{R}$ via calculating the correlation between any two actors in $\mathbf{H}'$. The model is trained to predict the relation between any two actors for the synthetic group. That is, we try to minimize the following objective function:

$$\begin{aligned}\mathcal{L} &= \mathrm{BCE}(\mathbf{R}, \mathbf{S}^\theta) \\ &= \frac{\sum_{i,j=1}^{P+Q} \mathbf{S}^\theta_{i,j} \log \mathbf{R}_{i,j} + (1 - \mathbf{S}^\theta_{i,j}) \log(1 - \mathbf{R}_{j,j})}{(P+Q)^2},\end{aligned} \tag{1}$$

where BCE$(\cdot)$ creates a criterion that measures the Binary Cross Entropy between the predicted relation matrix $\mathbf{R}$ and the real relation matrix $\mathbf{S}^\theta$. The learned representation can then be used as input to skeleton-based group activity modeling tasks, such as skeleton-based GAR.

3.2 Actor-association graph convolution module

Existing spatial-temporal graph convolution networks have been successfully applied to skeleton-based action recognition, which usually consists of two parts: spatial graph convolution and temporal graph convolution [6, 33, 34, 48]. In this section, given the drawbacks of modeling the interactions between multi-actors in the existing networks, we propose a spatial intra-inter actor graph convolution to replace the original spatial graph convolution, with two information aggregation modes. As shown in Figure 3, the intra-actor graph convolution aggregates feature from neighboring kinematic related joints to model individual actors' dynamics

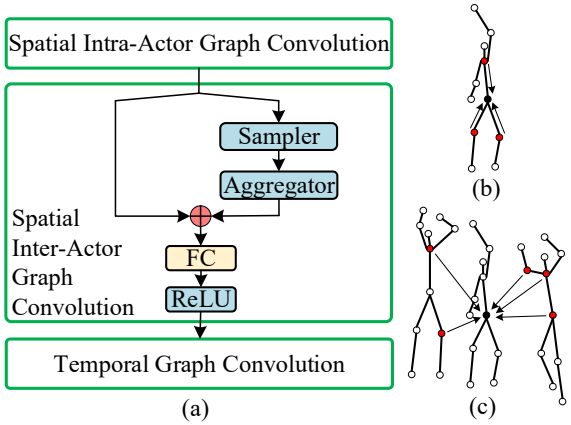

Figure 3: (a) Actor-association graph convolution module. (b) and (c) are information aggregation modes for spatial intra-actor graph convolution and spatial inter-actor graph convolution, respectively. The black dot is a root joint. Red dots are sampled neighbors. Arrows represent directions of information flow.

while the inter-actor graph convolution concentrates on assembling joints' feature from other actors to model interactions among them. Based on it, we construct a new spatial-temporal graph convolution network block, named AAGCM, to model spatiotemporal information of skeleton-based group activity. The detailed structure of AAGCM is shown in Figure 3 (a).

Spatial intra-actor graph convolution: Group skeleton graph is firstly divided into multiple independent actor skeleton graphs to model individual dynamics separately. As an actor skeleton graph contains a definite number of joints as nodes, being consistent with previous works, spatial intra-actor graph convolution is based on a graph with fixed size and directly optimizes the node embeddings using matrix-factorization based objectives. Specifically, the intra-actor neighbor set of nodes is defined as an adjacent matrix $\mathbf{A} \in \{0, 1\}^{N \times N}$ based on $\mathcal{E}_{\text{spaintra}}$, which is partitioned into 3 partitions: the centripetal partition containing neighboring nodes that are closer to the skeleton center, the node itself and otherwise the centrifugal partition. Let $\mathbf{F} \in \mathbb{R}^{N \times C}$ and $\mathbf{F}' \in \mathbb{R}^{N \times C'}$ denote the input and output representations of all joints for an actor in one frame, where C and C' are the input and output feature dimensions. The intra-actor spatial graph convolution is computed as:

$$\mathbf{F}' = \sum_{p \in \mathcal{P}} \bar{\mathbf{A}}_p \mathbf{F} \mathbf{W}_p, \tag{2}$$

where $\mathcal{P} = \{\text{root}, \text{centripetal}, \text{centrifugal}\}$ denotes the spatial partitions, $\bar{\mathbf{A}}_p = \mathbf{\Lambda}_p^{-\frac{1}{2}} \mathbf{A}_p \mathbf{\Lambda}_p^{-\frac{1}{2}} \in \mathbb{R}^{N \times N}$ is the normalized adjacent matrix and $\mathbf{\Lambda}_p^{ij} = \sum_j (\mathbf{A}_p^{ij}) + \alpha$, α is set to 0.001 to avoid empty rows. $\mathbf{W}_p \in \mathbb{R}^{1 \times 1 \times C \times C'}$ is the weight of the 1×1 convolution for each partition.

It is important to notice that $\bar{\mathbf{A}}_p$ makes graph convolution approaches inherently transductive and do not naturally generalize to unseen nodes. Specifically, adding new nodes into the graph will change $\bar{\mathbf{A}}_p$ and deviate embedding of the existing nodes. However, a group skeleton graph usually consists of variable numbers of

actors. Thus, it is unreasonable to directly handle the group skeleton graph with transductive graph convolution when modeling the spatial inter-actor interactions. We design an inductive graph convolution-based module to deal with the above issues here.

Spatial inter-actor graph convolution: Inspired by GraphSAGE [13], the strategy that samples neighbors of each node in graph and aggregates information of the neighbors is adopted here to model the spatial inter-actor interactions. We explain it in the case where the group skeleton spatial-temporal graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, and representations for all joints $\mathbf{x}_{\mathbf{v}_{ti}^m}$, $\mathbf{v}_{ti}^m \in \mathcal{V}$, are provided as input.

Initially, we randomly sample part joints from direct neighbors determined by $\mathcal{E}_{\text{spainter}}$. A set of sampled neighbor joints, $\mathcal{U}_{ti}^m$, is generated as follows:

$$\mathcal{U}_{ti}^m = \text{sampler}(\mathcal{B}_{ti}^m, \rho), \tag{3}$$

where $\mathcal{B}_{ti}^m$ is a set of neighbor joints of $\mathbf{v}_{ti}^m$, $\mathcal{B}_{ti}^m \subseteq \mathcal{V}$ and $\mathcal{B}_{ti}^m = \{\mathbf{v}_{tj}^n \mid n = 1, ..., M; n \neq m; j = 1, ..., N\}$. These are exactly the first-order neighbors of it. sampler$(\cdot, \rho)$ is a sampler function which selects the node in accordance with the sampling probability ρ.

$$\rho_{\mathbf{v}_{tj}^n} = \frac{\exp(\text{LeakyReLU}(\mathbf{W}_{g2}[\mathbf{W}_{g1}\mathbf{x}_{\mathbf{v}_{tj}^n} || \mathbf{W}_{g1}\mathbf{x}_{\mathbf{v}_{ti}^m}]))}{\sum_{k \in \mathcal{B}_{ti}^m} \exp(\text{LeakyReLU}(\mathbf{W}_{g2}[\mathbf{W}_{g1}\mathbf{x}_k || \mathbf{W}_{g1}\mathbf{x}_{\mathbf{v}_{ti}^m}]))}, \tag{4}$$

where $\rho_{\mathbf{v}_{tj}^n}$ is the normalized sampling probability of joint $\mathbf{v}_{tj}^n$. $\mathbf{W}_{g1}$ and $\mathbf{W}_{g2}$ are two learnable weight matrixes. $||$ is the concatenation operation. The sampling operation provides the benefit that the computational and memory complexity is constant with respect to the size of a graph.

Then, we aggregate the embeddings of joints in the sampled set $\mathcal{U}_{ti}^m$ to form a corresponding neighborhood vector $\bar{\mathbf{x}}_{\mathbf{v}_{ti}^m}$ for $\mathbf{v}_{ti}^m$. This operation is expressed as

$$\bar{\mathbf{x}}_{\mathbf{v}_{ti}^m} = \text{aggregator}(\mathbf{x}_v, \forall v \in \mathcal{U}_{ti}^m), \tag{5}$$

where aggregator$(\cdot)$ represents an aggregator function. As there is no obvious sequential feature among the sampled joints when aggregating them, aggregator$(\cdot)$ must have symmetry property to be insensitive to the input order. Here, we explore a simple mean aggregator function, where $\bar{\mathbf{x}}_{\mathbf{v}_{ti}^m}$ is formulated as:

$$\bar{\mathbf{x}}_{\mathbf{v}_{ti}^m} = \sum_{v \in \mathcal{U}_{ti}^m} \frac{\mathbf{x}_v}{|\mathcal{U}_{ti}^m|}. \tag{6}$$

Finally, we let the representations of each node bring together information from intra-actors and inter-actors. To be specific, the joint's intra-actor representations $\mathbf{x}_{\mathbf{v}_{ti}^m}$ is added to its aggregated neighborhood vector $\bar{\mathbf{x}}_{\mathbf{v}_{ti}^m}$, which is later fed into a fully connected layer with a nonlinear activation function σ. This process is formulated as:

$$\mathbf{x}'_{\mathbf{v}_{ti}^m} = \sigma((\mathbf{x}_{\mathbf{v}_{ti}^m} + \bar{\mathbf{x}}_{\mathbf{v}_{ti}^m})\mathbf{W}_{\text{fc}} + \mathbf{b}_{\text{fc}}), \tag{7}$$

where $\mathbf{x}'_{\mathbf{v}_{ti}^m}$ is the representation that integrates intra-actor information and intra-actor information. $\mathbf{W}_{\text{fc}}$ and $\mathbf{b}_{\text{fc}}$ are weight and bias of the fully connected layer.

Temporal graph convolution: Similar to most existing spatial-temporal graph convolution networks [6, 34], we construct a temporal graph by connecting identical joints in consecutive frames and use regular 1D convolution along the temporal dimension as the temporal graph convolution to model the temporal dynamic information.

4 EXPERIMENTS

4.1 Setup

Dataset: We evaluate our approach on the following three datasets: *Volleyball* [15], *Collective Activity* [8], and *Mutual NTU* [27] datasets. Volleyball dataset is a large-scale latest and largest benchmark for group activity recognition. It contains 55 volleyball games with 4,830 labeled frames and multiple surrounding unannotated frames, where each player is annotated with the bounding box and one of the 9 individual actions, and the whole group is assigned with one of the 8 group activity labels. In evaluation, we use 39 videos for training and 16 videos for testing following [15]. Collective Activity dataset is composed of 44 video clips collected by a low-resolution handheld camera. A total of 2,481 activity clips are annotated with one of 5 group activities, including crossing, queueing, walking, talking, and waiting. Following [14], we merge the classes walking and crossing as moving. NTU-120 RGB+D [27] is currently the largest dataset with 3D joints annotations for human action recognition. Mutual NTU contains 24,828 action samples in 26 mutual action classes sampled from NTU-120 RGB+D. We follow the two recommended benchmarks: (1) cross-subject (X-sub) benchmark: the 106 subjects are split into training and testing groups. Each group contains 53 subjects. (2) cross-setup (X-set) benchmark: training data comes from samples with even setup IDs, and testing data comes from samples with odd setup IDs

Metrics: The performance of representation learning algorithms is evaluated by the group activity recognition task. In the experiments, Multi-Class Classification Accuracy (MCA), the percentage of correct predictions, is utilized as a performance metric. To validate the effectiveness of the proposed self-supervised representation learning method, we follow the commonly used linear evaluation protocol [18, 26], where a linear classifier is trained on top of the frozen representation model with the annotated samples, and test accuracy is used as a proxy for representation quality. Besides, semi-supervised fine-tuning is also adopted [26], where the representation model is finetuned with a certain fraction of labeled samples on the training set.

Implementation details: The public available *OpenPifPaf* [21] toolbox is applied to estimate and track the skeletons of actors on the Volleyball and Collective Activity datasets. In this paper, the default representation model is constructed based on Shift-GCN [6], where part spatial-temporal graph convolution network blocks are replaced by our AAGCMs in different layers. During the self-supervised representation learning, the encoder model is trained on the training set without ground-truth labels. When conducting supervised learning, the representation model and linear classifier are randomly initialized and jointly trained on the training set with ground-truth labels.

4.2 Comparison experiments

To quantitatively evaluate the performance, Table **??** and Figure 4 list the results of proposed approaches and several other self-supervised representation learning algorithms. The method which only trains

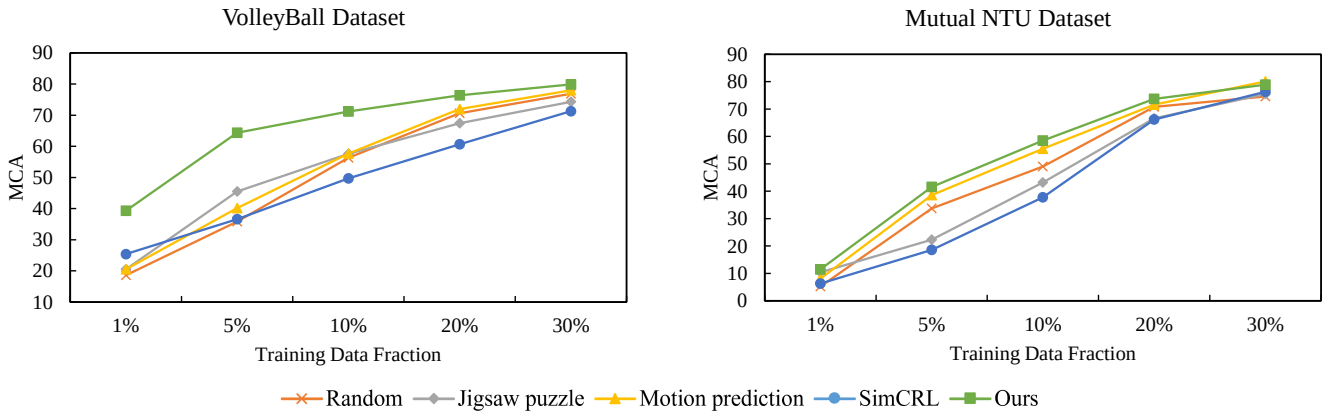

Figure 4: Data efficiency for semi-supervised training. Training data fraction vs. classifier accuracy on the Volleyball and Mutual NTU (X-sub) datasets. The method involves self-supervised pretraining, followed by semi-supervised fine-tuning.

Table 1: Linear classification accuracies from different self-supervised learning strategies.

Method	Volleyball	Collective Activity	Mutual NTU X-sub	Mutual NTU X-set
Random	30.8%	50.6%	11.5%	11.8%
Jigsaw puzzle [26]	28.6%	65.8%	9.5%	12.9%
Motion prediction [26]	46.4%	69.9%	12.2%	19.1%
SimCRL [4]	47.0%	55.0%	11.0%	12.8%
AimCLR [12]	46.7%	52.3%	**17.8%**	**20.1%**
Ours	**58.4%**	**88.9%**	17.3%	19.6%

the linear classifier and freezes the randomly initialized representation model is denoted as Random. We regard it as one of our baselines. Since there are few prior works on self-supervised representation learning for skeleton-based group activity recognition, we enumerate several alternative self-supervision tasks to provide meaningful references. Jigsaw puzzle and motion perdition are implemented based on the work of Lin et al. [26]. SimCRL is a popular framework for contrastive learning of visual representations [4]. We obtain the representation model by maximizing cosine similarity among original data, spatial and temporal mask based on SimCRL. Meanwhile, in order to further verify the effectiveness and generality of AAGCM, we apply it to several state-of-the-art spatial-temporal graph convolution networks for skeleton-based group activity recognition, as shown in Table 2.

Comparison of self-supervised representation learning approaches: To assess the quality of the learned representations, we first follow the widely used linear evaluation protocol. We train the representation model in different self-supervised learning strategies employing the unlabelled training set, and then a linear classifier corresponding to the number of classes is trained on top of the learned features. The results are reported in Table **??**. We can notice that the proposed approach shows a dramatic improvement of 27.6%, 38.3%, 5.8%, and 7.8% from the Random baseline on the Volleyball, Collective Activity, Mutual NTU X-sub, and X-set datasets. Also, we can see that ours gets significantly better MCA than the other three self-supervised representation tasks on the Volleyball and the Collective Activity datasets. This improvement verifies that our methods can force the encoder to extract more effective representations. It is worth noting that the correlation between supervision signals from our approach and relations among actors is more significant than that from others. Therefore, we achieve better performance with much more suitable feature representations compared to others.

The significance of self-supervised representation learning is that it allows us to learn good representations without using large annotated databases, which can be used to learn new tasks in which data is scarce. Therefore, we also evaluate the data efficiency of our representation for skeleton-based GAR in semi-supervised settings, where the learned representation model is finetuned with a certain fraction of labeled samples on the training set. Figure 4 summarizes the results. Our representation significantly improves classification performance over the random baseline. When the training data fraction declines from 30% to 1%, we can observe that the less annotated data, the more significant superiority we have. This illustrates the value of our approach. Apart from that, we found that Jigsaw puzzle and SimCRL have adverse side effects sometimes. Maybe it's because they lack an effective mechanism to drive the representation model to learn interactions between actors.

Combination AAGCM with state-of-the-art networks: To verify the effectiveness and generality of AAGCM, we apply it to several state-of-the-art spatial-temporal graph convolution networks. The results in supervised settings are shown in Table 2. From two large-scale datasets, the Volleyball and the Mutual NTU, it is found that AAGCM produces good quality results. Combined with AAGCM, these state-of-the-art networks successfully improve their performance in skeleton-based GAR by a sizable margin. This verifies both the effectiveness and the generality of the proposed AAGCM. The prime reason why it works is that AAGCM makes up a fatal defect of the existed networks that lack effective mechanisms to model interactions among actors. Inserting multiply AAGCMs into different network depths makes the model can represent multi-level group relevant interactions between actors. One thing to note here is that AAGCM does not yield the expected same improvements in the Collective Activity dataset. This however may be explained by the fact that there is a very limited number of samples in this dataset.

Table 2: Evaluation of AAGCM based on various spatial-temporal graph convolution networks in a supervised manner. We report the 95% confidence intervals over four times repetitions.

Method	Volleyball	Collective Activity	Mutual NTU	
			X-sub	X-set
ST-GCN [48]	[88.42%, 88.88%]	[83.89%, 84.18%]	[81.22%, 81.82%]	[77.63%,78.12%]
ST-GCN-AAGCM	[89.85%, 91.47%]	[83.91%, 84.28%]	[84.39%, 85.23%]	[81.01%, 81.84%]
AGCN [34]	[85.25%, 85.53%]	[81.58%, 81.84%]	[85.16%, 86.38%]	[86.19%, 86.96%]
AGCN-AAGCM	[88.87%, 89.27%]	[77.89%, 78.65%]	**[87.78%, 88.43%]**	**[88.02%, 89.28%]**
Shift-GCN [6]	[86.30%, 87.08%]	[76.29%, 77.19%]	[83.22%, 84.07%]	[84.01%, 85.18%]
Shift-GCN-AAGCM	[88.93%, 89.25%]	[79.24%, 79.98%]	[87.25%, 88.24%]	[86.28%, 87.32%]
CTR-GCN [5]	[90.13%, 90.89%]	**[87.01%, 88.28%]**	[85.79%, 86.27%]	[85.01%,86.24%]
CTR-GCN-AAGCM	**[92.09%, 92.97%]**	[84.59%,85.03%]	[86.73%, 87.64%]	[86.52%, 87.21%]

4.3 Ablation experiments

Self-supervised representation learning task: We start with the exploration of the parameters of our self-supervised representation learning task by varying the synthetic group size and the subgroup number. The results following the linear evaluation protocol are presented in Table 3 and Table 4. Because AAGCM supports variable group size, we can carry the self-supervised representation learning with one group size and deploy the representation model with another group size. The results in Table 3 show that the performance of our representation learning task varies at different synthetic group sizes, but the representation models can always provide powerful features that facilitate group activity classification. In addition, we experiment with the subgroup number. Twelve actors in the synthetic group are uniformly sampled from different naive groups here. Results illustrated in Table 4 show that increasing the subgroup number leads to performance degradation, 58.4% to 52.1%, from which it can be concluded that enhancing the complexity of the pretext task, subgroup identification, cannot help the model to learn better representations for group activity recognition.

Actor-association graph convolution module: In Table 5, we compare the performance of our approach when different insertion depth for AAGCM is used. The best performance is obtained as AAGCMs are inserted into block1, 5, and 8. This is as expected since the multi-level interactions between actors can be modeled in this way. We can discover that inserting AAGCM into block1 is better than block5 and block8 in two datasets, which shows that modeling low-level interactions between actors is more important than high-level. Besides, we investigate the performance of different sampling aggregator functions. Apart from *Mean*, the remaining aggregator functions are: *MaxPool* where an elementwise max-pooling operation is applied to aggregate information across the neighbor set [13] and *Graph Attention* where different importances are assigned to nodes based on self-attention within a neighborhood to aggregate information [41]. The results are reported in Tabel 6. From these results, we can see that although the mean aggregator function is simple, it always performs better than the other two functions with different sampling probabilities. So the mean aggregator function is the best choice here.

Multimodal Fusion: In the last ablation, we compare RGB and skeleton for GAR and also combine them using late fusion. The results are presented in Table 7. ARG is an excellent RGB-based GAR approach, where a flexible and efficient Actor Relation Graph is built to simultaneously capture the appearance and position relation between actors [46]. It is satisfactory to find that our skeleton-based method 93.0% outperforms RGB-based ARG 92.6%. Meanwhile, fusing multiple modalities bring a performance improvement compared to the method of using a single modality, since they contain complementary and discriminative appearance and geometrical information, which is useful for activity analysis.

Table 3: Ablation study of synthetic group size on the Volleyball dataset. Each synthetic group contains two subgroups.

# Synthetic Group Size	10	11	12	13	14
Volleyball	56.4%	56.2%	58.4%	**58.6%**	53.2 %

Table 4: Ablation study of subgroup number on the Volleyball dataset.

# Subgroup Number	2	3	4	5
Volleyball	**58.4%**	57.1%	55.4%	52.1%

Table 5: Ablation study of insertion depth for AAGCM on the Volleyball and Mutual NTU(X-sub) in a supervised manner.

Insertion Depth	block1	block5	block8	block1,5,8
Volleyball	89.0%	88.3%	88.0%	**89.2%**
Mutual NTU	86.5%	85.6%	85.8%	**87.5%**

Table 6: Ablation study of AAGCM with different aggregator functions on the Volleyball and Mutual NTU(X-sub) in a supervised manner, where the baseline indicates the model without AAGCM.

Aggregator	Baseline	Mean	MaxPool	Attention
Volleyball	87.0%	**89.2%**	88.6%	88.0%
Mutual NTU	83.2%	**87.5%**	86.2%	85.4%

4.4 Analysis

To analyze the benefits of AAGCM, Figure 6 (b) shows the neural response magnitude of each actor in the last layer of our encoder model trained with annotated data. In order to identify *left spike* in a

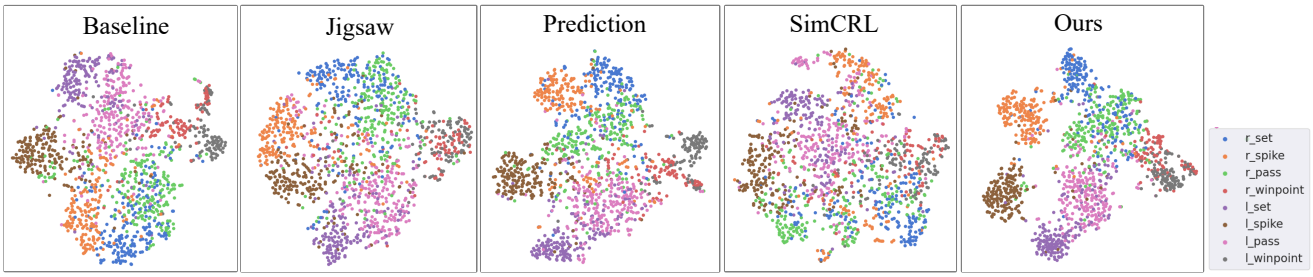

Figure 5: t-SNE [40] visualization of representation on the Volleyball dataset learned by self-supervised learning tasks and finetuned with 20% labeled samples. Each sample is visualized as one point and colors denote different group activities.

Table 7: Comparison of different modalities on the Volleyball dataset in supervised manner.

Modality	RGB	Skeleton	RGB+Skeleton
Method	ARG [46]	CTR-GCN-AAGCM	Later Fusion
Volleyball	92.6%	93.0%	**94.2%**

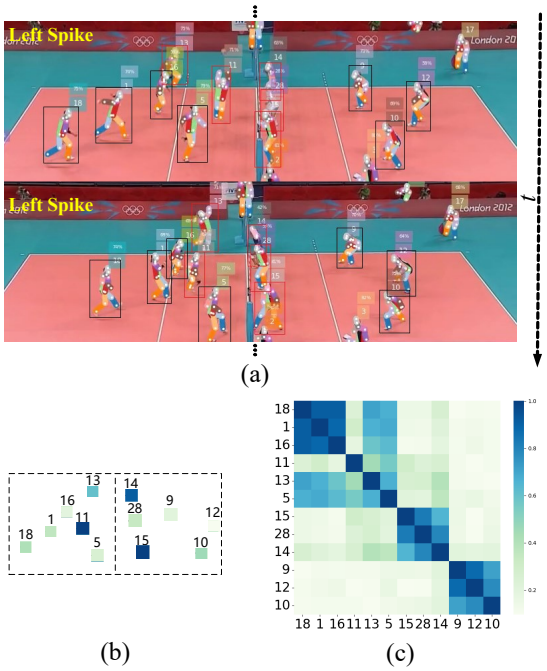

Figure 6: Example of learned representations obtained by AAGCM-based encoder in a supervised manner. (a) Actors of the group, labeled by bounding boxes, and group activity label; (b) Normalized neural response magnitude for each actor; (c) Correlation matrix of actors' representations. When the corresponding value is larger in (b) and (c), the color is deeper.

volleyball game, the states of attaquant principal, setter, and blocker are the most crucial points. In the visualization results, we can see that the response magnitude of attaquant principal(13), setter(11), and blockers (14, 15) are significantly higher than others. Therefore, AAGCM automatically learns the key actors to determine the group activity in the scene. Meanwhile, we also visualize the relation information between actors in Figure 6 (c), where normalized cosine similarities for the learned representations are calculated and plotted in a correlation matrix. AAGCM has a high capability in modeling interactions among actors for group activity representation learning. As expected, there are higher correlations among actors who have stronger cooperative relationships with each other. For instance, actors (14, 15, 28) cooperate closely to make a combination block, which is reflected in the correlation matrix. The collaborative relationships between the back row players or front row players are also clearly manifested in the matrix.

Figure 5 shows the t-SNE visualization for embedding the group activity representation learned by different self-supervised learning tasks. The baseline is the model without pre-training. Specifically, we project the representations of samples on the test set of the Volleyball dataset into 2-dimensional space using t-SNE. We can observe that the representations learned by ours are more concentrated for the same kind of samples and more separated for different. These visualization results indicate the proposed approach is more effective for group activity recognition.

However, the group skeleton sequence in this paper is assumed to keep high continuity in the temporal domain for each actor, which is challenging for current multi-person pose estimation and tracking algorithms. Meanwhile, our information aggregation models hinder the direct flow across spacetime for capturing complex regional spatial-temporal relations between actors. These will be our further research problems in the future.

5 CONCLUSION

This paper has presented an efficient self-supervised representation learning task and a plug-in graph convolution module for skeleton-based group activity recognition. First, we define a pretext task of subgroup identification in a synthetic group to learn the spatiotemporal dynamics of individual actors and group relevant interactions between them. Second, we propose an Actor-Association Graph Convolution Module for spatial-temporal graph convolution networks to effectively model the multi-level relations between actors and adapt to variable group sizes. With experiments on the Volleyball, Collective Activity, and Mutual NTU datasets, we confirmed both the self-supervised representation learning algorithm and the network architecture for skeleton-based group activity recognition outperform the existing state-of-the-art results by a sizable margin.

ACKNOWLEDGMENTS

This work was supported in part by the NSFC under Grants U1803264, 62072334.

REFERENCES

[1] Mohamed Rabie Amer, Peng Lei, and Sinisa Todorovic. 2014. Hirf: Hierarchical random field for collective activity recognition in videos. In *ECCV*.

[2] Mohamed R Amer, Dan Xie, Mingtian Zhao, Sinisa Todorovic, and Song-Chun Zhu. 2012. Cost-sensitive top-down/bottom-up inference for multiscale activity recognition. In *ECCV*.

[3] Timur Bagautdinov, Alexandre Alahi, François Fleuret, Pascal Fua, and Silvio Savarese. 2017. Social scene understanding: End-to-end multi-person action localization and collective activity recognition. In *CVPR*.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *ICML*.

[5] Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. 2021. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *ICCV*.

[6] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. 2020. Skeleton-based action recognition with shift graph convolutional network. In *CVPR*.

[7] Wongun Choi and Silvio Savarese. 2013. Understanding collective activitiesof people from videos. *PAMI* 36, 6 (2013), 1242–1257.

[8] Wongun Choi, Khuram Shahid, and Silvio Savarese. 2009. What are they doing?: Collective activity classification using spatio-temporal relationship among people. In *ICCV workshop*.

[9] Wongun Choi, Khuram Shahid, and Silvio Savarese. 2011. Learning context for collective activity recognition. In *CVPR*.

[10] Carl Doersch, Abhinav Gupta, and Alexei A Efros. 2015. Unsupervised visual representation learning by context prediction. In *ICCV*.

[11] Kirill Gavrilyuk, Ryan Sanford, Mehrsan Javan, and Cees GM Snoek. 2020. Actor-transformers for group activity recognition. In *CVPR*.

[12] Tianyu Guo, Hong Liu, Zhan Chen, Mengyuan Liu, Tao Wang, and Runwei Ding. 2022. Contrastive Learning from Extremely Augmented Skeleton Sequences for Self-supervised Action Recognition. In *AAAI*.

[13] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *NIPS*.

[14] Guyue Hu, Bo Cui, Yuan He, and Shan Yu. 2020. Progressive relation learning for group activity recognition. In *CVPR*.

[15] Mostafa S Ibrahim, Srikanth Muralidharan, Zhiwei Deng, Arash Vahdat, and Greg Mori. 2016. A hierarchical deep temporal model for group activity recognition. In *CVPR*.

[16] Dinesh Jayaraman and Kristen Grauman. 2015. Learning image representations tied to ego-motion. In *ICCV*.

[17] Dahun Kim, Donghyeon Cho, and In So Kweon. 2019. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*.

[18] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. 2019. Revisiting self-supervised visual representation learning. *CVPR*.

[19] Nikos Komodakis and Spyros Gidaris. 2018. Unsupervised representation learning by predicting image rotations. In *ICLR*.

[20] Bruno Korbar, Du Tran, and Lorenzo Torresani. 2018. Cooperative Learning of Audio and Video Models from Self-Supervised Synchronization. In *NIPS*.

[21] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. 2021. OpenPifPaf: Composite Fields for Semantic Keypoint Detection and Spatio-Temporal Association. *arXiv preprint arXiv:2103.02440* (2021).

[22] Tian Lan, Leonid Sigal, and Greg Mori. 2012. Social roles in hierarchical models for human activity recognition. In *CVPR*.

[23] Tian Lan, Yang Wang, Weilong Yang, Stephen N Robinovitch, and Greg Mori. 2011. Discriminative latent models for recognizing contextual group activities. *PAMI* 34, 8 (2011), 1549–1562.

[24] Linguo Li, Minsi Wang, Bingbing Ni, Hang Wang, Jiancheng Yang, and Wenjun Zhang. 2021. 3d human action representation learning via cross-view consistency pursuit. In *CVPR*.

[25] Shuaicheng Li, Qianggang Cao, Lingbo Liu, Kunlin Yang, Shinan Liu, Jun Hou, and Shuai Yi. 2021. GroupFormer: Group Activity Recognition with Clustered Spatial-Temporal Transformer. In *ICCV*.

[26] Lilang Lin, Sijie Song, Wenhan Yang, and Jiaying Liu. 2020. MS2L: Multi-Task Self-Supervised Learning for Skeleton Based Action Recognition. In *ACM MM*.

[27] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C Kot. 2019. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE transactions on pattern analysis and machine intelligence* 42, 10 (2019), 2684–2701.

[28] Aravindh Mahendran, James Thewlis, and Andrea Vedaldi. 2018. Cross pixel optical-flow similarity for self-supervised learning. In *ACCV*.

[29] Mehdi Noroozi and Paolo Favaro. 2016. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*.

[30] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. 2017. Representation learning by learning to count. In *ICCV*.

[31] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. 2016. Context encoders: Feature learning by inpainting. In *CVPR*.

[32] Senthil Purushwalkam Shiva Prakash and Abhinav Gupta. 2020. Demystifying Contrastive Self-Supervised Learning: Invariances, Augmentations and Dataset Biases. In *NIPS*.

[33] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. 2019. Skeleton-based action recognition with directed graph neural networks. In *CVPR*.

[34] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. 2019. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *CVPR*.

[35] Tianmin Shu, Dan Xie, Brandon Rothrock, Sinisa Todorovic, and Song Chun Zhu. 2015. Joint inference of groups, events and human roles in aerial videos. In *CVPR*.

[36] Xiangbo Shu, Jinhui Tang, Guojun Qi, Wei Liu, and Jian Yang. 2019. Hierarchical long short-term concurrent memory for human interaction recognition. *PAMI* (2019), 1110 – 1118.

[37] Xiangbo Shu, Liyan Zhang, Yunlian Sun, and Jinhui Tang. 2020. Host-parasite: Graph LSTM-in-LSTM for group activity recognition. *PAMI* (2020), 663 – 674.

[38] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. 2015. Unsupervised learning of video representations using lstms. In *ICML*.

[39] Kun Su, Xiulong Liu, and Eli Shlizerman. 2020. Predict & cluster: Unsupervised skeleton based action recognition. In *CVPR*.

[40] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[41] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[42] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. 2016. Generating videos with scene dynamics. In *NIPS*.

[43] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. 2018. Tracking emerges by colorizing videos. In *ECCV*.

[44] Minsi Wang, Bingbing Ni, and Xiaokang Yang. 2017. Recurrent modeling of interaction context for collective activity recognition. In *CVPR*.

[45] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. 2018. Learning and using the arrow of time. In *CVPR*.

[46] Jianchao Wu, Limin Wang, Li Wang, Jie Guo, and Gangshan Wu. 2019. Learning actor relation graphs for group activity recognition. In *CVPR*.

[47] Rui Yan, Lingxi Xie, Jinhui Tang, Xiangbo Shu, and Qi Tian. 2020. HiGCIN: Hierarchical Graph-based Cross Inference Network for Group Activity Recognition. *PAMI* (2020), 1–1.

[48] Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*.

[49] Richard Zhang, Phillip Isola, and Alexei A Efros. 2016. Colorful image colorization. In *ECCV*.

[50] Richard Zhang, Phillip Isola, and Alexei A Efros. 2017. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *CVPR*.