Learning Spatial-Preserved Skeleton Representations for Few-Shot Action Recognition

Ning Ma[1,2,3], Hongyi Zhang[1,2,3] *, Xuhui Li[1,2,3], Sheng Zhou[1,2,3] †, Zhen Zhang[4], Jun Wen[5], Haifeng Li[6], Jingjun Gu[1,2], and Jiajun Bu[1,2,3] ‡

[1] Zhejiang Provincial Key Laboratory of Service Robot, College of Computer Science, Zhejiang University, Hangzhou, China
[2] Alibaba-Zhejiang University Joint Institute of Frontier Technologies, China
[3] Ningbo Research Institute, Zhejiang University, Ningbo, China
[4] Department of Computer Science, National University of Singapore, Singapore
[5] Department of Biomedical Informatics, Harvard Medical School, USA
[6] The Children's Hospital Zhejiang University School of Medicine, China
{ma_ning, zhy1998, 12021064, zhousheng_zju,junwen, 6199005, gjj, bjj}@zju.edu.cn, zhen@nus.edu.sg

Abstract. Few-shot action recognition aims to recognize few-labeled novel action classes and attracts growing attentions due to practical significance. Human skeletons provide explainable and data-efficient representation for this problem by explicitly modeling spatial-temporal relations among skeleton joints. However, existing skeleton-based spatial-temporal models tend to deteriorate the positional distinguishability of joints, which leads to fuzzy spatial matching and poor explainability. To address these issues, we propose a novel spatial matching strategy consisting of spatial disentanglement and spatial activation. The motivation behind spatial disentanglement is that we find more spatial information for leaf nodes (e.g., the "hand" joint) is beneficial to increase representation diversity for skeleton matching. To achieve spatial disentanglement, we encourage the skeletons to be represented in a full rank space with rank maximization constraint. Finally, an attention based spatial activation mechanism is introduced to incorporate the disentanglement, by adaptively adjusting the disentangled joints according to matching pairs. Extensive experiments on three skeleton benchmarks demonstrate that the proposed spatial matching strategy can be effectively inserted into existing temporal alignment frameworks, achieving considerable performance improvements as well as inherent explainability.

Keywords: Action Recognition; Few-shot Learning; Explainable AI

*Equal Contribution With the First Author
†Corresponding Author
‡Corresponding Author

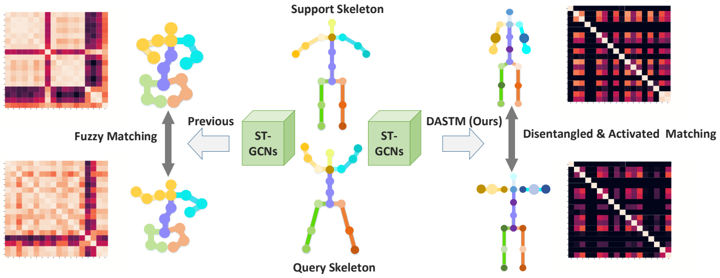


Fig. 1: The Illustration that demonstrates the fuzzy skeleton matching using degenerated spatial representation, and the disentangled skeleton matching by our spatial disentanglement and activation. The brighter grids in heatmaps denote larger similarity among intra-skeleton joints. The size of joints denotes their importance in matching.

1 Introduction

Action recognition has achieved tremendous success with developed deep learning models and abundant action recordings [21,39]. However, in many scenarios like healthcare, collecting and labeling enough medical action videos may spend several years with the efforts of multiple medical experts. To address this data scarcity, few-shot action recognition is proposed and attracts growing attentions [2,7,15,16,20,28].

Given a few labeled demonstrations of novel action classes, i.e., *support* actions, few-shot action recognition usually aims to predict the unlabeled actions, i.e., *query* actions. Existing works can be divided into video-based [2,16,28,29] and skeleton-based methods [15,25,33]. In the video-based methods, the high dimensional redundancy information such as luminance and background is usually unreliable under few-shot scenarios. In contrast, skeleton sequences provide *explainable* and *data-efficient* action representation by explicitly modeling the spatial-temporal relation of body joints. Existing methods usually perform Spatial-Temporal Graph Convolution (ST-GCN, [42]) to capture the spatial-temporal relations among skeletons. However, the over-smoothing of graph convolution tends to make the nodes representations indistinguishable, resulting in the partial loss of joints' positional information after ST-GCN. The left part of Fig. 1 illustrates the over-smoothed similarity heatmaps of intra-skeleton joints and the fuzzy spatial matching caused by the degenerated spatial representation. This fuzzy spatial matching further leads to fragile spatial-temporal matching between the query and the support skeletons in few-shot action recognition.

To address the distinguishability caused by over-smoothing, typical graph learning methods try to drop graph edges (DropEdge, [32]) or push away non-adjacent nodes (PairNorm [45]). However, in small size skeleton graphs, these methods may destroy the skeleton structure, or smoothing joints' representations as long as they are adjacent, e.g., PairNorm is prone to produce distinguishable representations between elbow joints and hand joints. Instead, the disentangled joint representations naturally produce distinguishability for spatial matching (see the right of Fig. 1). For example, the skeleton's leaf nodes (joints) like "hand" usually contain essential positional information. Hence disentangling these joints from the spatial convolution process can preserve spatial structure for skeleton matching. To achieve this disentanglement, one strategy is to encourage the joint representations to have less linear dependence with rank maximization on skeleton representation matrices.

Although this disentanglement encourages important joints to have independent representations, it does not filter out unimportant joints in the matching process. In other words, when matching a query skeleton and a support skeleton, the query skeleton should know whether one joint is significant for the support one and vice versa. Motivated by this, we design two independent cross-attention modules for query and support pairs to adaptively activate their spatial information.

Finally, this spatial matching strategy can be orthometric with popular temporal matching methods like Dynamic Time Warping (DTW, [34]), which determines the optimal temporal matching strategy for two skeleton sequences. By seamlessly inserting the proposed spatial matching into temporal matching, we propose a holistic spatial-temporal measurement for skeleton sequences. Our ablation experiments on NTU RGB+D 120 [22] and Kinetics [17] demonstrate its effectiveness on few-shot action classification tasks. Our method can be summarized as **D**isentangled and **A**daptive **S**patial-**T**emporal **M**atching (**DASTM**) for few-shot action recognition. The contributions are enumerated as follows:

- We systematically investigate skeleton-based few-shot action recognition and find the degeneration of spatial information existing in mainstream methods under data scarcity scenarios.
- To alleviate the degenerated spatial representations, we propose a novel spatial matching strategy through adaptively disentangling and activating the representations of skeleton joints.
- Extensive few-shot experiments on public action datasets demonstrate the effectiveness of our holistic spatial-temporal matching.
- Our heatmap visualizations demonstrate which joints are vital in recognition tasks, providing explainable predictions for trustworthy action recognition.

2 Related Work

2.1 Few-Shot Action Recognition

Few-shot action recognition aims to recognize novel action classes given a few labeled action examples. Due to practical significance, this recognition paradigm recently attracts enormous attentions [2,4,5,7,14,16,19,20,26,28–30,38,40,41,43,44,46,47].

[15] demonstrates that low dimensional skeleton data may be better for capturing spatial-temporal information. For skeleton-based representation, [33]

uses a temporal convolutional network [1] and computes the cosine distance between the query and support actions. [25] construct positive and negative pairs to learn the appropriate distance of different action classes.

2.2 Graph Representation and Matching

A human skeleton can be naturally represented as a graph whose joints and bones are denoted by vertexes and edges respectively. Based on graph representation, ST-GCN [42] and its variants [8,23,35] perform graph and temporal convolution to capture the spatial and temporal features. Although the ST-GCNs have been mainstream backbones for skeleton-based action classification, it is still challenging to match the action representations after these backbones. For example, the joints' positional information may be lost or inaccurate due to over-smoothed graph convolution in ST-GCNs. This over-smoothing is an intrinsic characteristic brought by message passing mechanism [45] and is further magnified without abundant training data.

2.3 Temporal Alignment

The start time and motion speed of two actions are usually mismatched in spite of the same action labels. This temporal mismatch has drawn elastic temporal alignment methods such as Dynamic Time Warping (DTW, [10,34]). DTW calculates an optimal match between two given sequences using dynamic programming. Recently, DTW and its variants have been used to boost the alignment of temporal features in low-shot setting [5,44].

3 Preliminary

Skeleton-based actions. A frame of skeleton graph can be defined as $G = \{\mathbf{X}, \mathbf{A}\}$, where $\mathbf{X} \in \mathbb{R}^{n\times 3}$ is the feature matrix, $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the adjacent matrix, n and 3 is the number of node (joints) and node dimension respectively. From the above definition, a skeleton-based action sequence is $\mathcal{G} = \{G_1, G_2, \cdots, G_M\}$, where M is number of the input frames.

Few-shot action recognition aims to adapt a model into novel classes and classify the unlabeled actions, i.e., *query* actions, given a few labeled actions, i.e., the *support* actions. There usually are three parts including training set $\mathcal{T}_{train}$, validation set $\mathcal{T}_{val}$, and test set $\mathcal{T}_{test}$, in which the action classes of the three parts do not overlap. In a training task, given N classes with K labeled support actions per class, the prototypical representation [36] of each class is $C_k = \frac{1}{K}\sum_{(\mathcal{G}_i^s, y_i^s)} f_\phi(\mathcal{G}_i^s) \times \mathbb{I}(y_i^s = k)$, where $\mathcal{G}_i^s$ is the support action, $\mathbb{I}$ is Indicator function. Let d denote the node's latent dimension, $f_\phi(\cdot) : \mathbb{R}^{M\times n\times 3} \rightarrow \mathbb{R}^{m\times n\times d}$ can be viewed as an action encoder with parameters ϕ. The prediction of a query action $\mathcal{G}^q$ can be formed into the following prototype method:

$$p_\phi(y = k \mid \mathcal{G}^q) = \frac{\exp\left(-dis\left(f_\phi(\mathcal{G}^q), C_k\right)\right)}{\sum_{k'} \exp\left(-dis\left(f_\phi(\mathcal{G}^q), C_{k'}\right)\right)}, \tag{1}$$

where $dis(x, y)$ is the distance of two action sequences. Compared with meta-learning methods like MAML [13], the prototype methods do not need large memory overhead to memorize multiple gradient steps, hence making it possible to incorporate larger backbones like ST-GCNs.

4 Proposed Framework

In essence, designing a distance measurement for query skeleton sequences and support skeleton sequences is the key to predict the query's categories. However, the mainstream methods like Spatial-Temporal Convolution (typically used in ST-GCN [42] and its extensions [23,35,42]) focus on learning integrated spatial-temporal representation, without considering the relation of different actions sequences. In our few-shot setting, directly measuring the integrated spatial-temporal representations are suboptimal with the following issues: 1) the degeneration of spatial representation; 2) the misalignment of temporal sequences. In the next part, we will discuss how the two issues arise and propose a holistic solution with spatial matching as well as temporal alignment.

4.1 Spatial Disentanglement and Activation

Learning Disentangled Skeleton Representation. Existing ST-GCNs get satisfying performance in capturing discriminative spatial-temporal features. However, their graph convolution operators often repeat message passing among skeleton nodes, which eventually leads to indistinguishable node embeddings. The phenomenon is also known as the over-smoothing problem, which does not seriously impact action classification once given abundant training data [45]. However, lacking sufficient data in few-shot learning, this over-smoothing is magnified, leading to the degeneration of nodes' positional representation. For example, the "elbow" node of a skeleton graph may get more spatial information of the "hand" node. If we measure the distance of two skeleton graphs, the degenerated spatial representation will result in noisy distances.

To alleviate the over-smoothing of graph convolution, some methods were proposed via directly dropping edges (DropEdge, [24,32]) or centering and rescaling node representations (PairNorm [45]). However, these methods are suboptimal for the particular skeleton structure. For example, the node "left hand" is relatively near the "left elbow" on a graph, hence is easier to have similar representations via DropEdge or PairNorm. We argue that smoothing the representation of "hands" may lose key action features. Besides that, the forced dropping edges will destroy the integrity or the symmetry of skeleton graphs. Instead, our strategy is disentangling the skeleton representations to keep the key spatial features like "hand" by reducing the independence among skeleton nodes. In the field of matrix analysis, the rank of a matrix is a permutation invariant diversity measure indicating the maximum number of linearly independent vectors in a matrix. Given a skeleton graph representations matrix $\mathbf{H}_{bi} \in \mathbb{R}^{n \times d}$, where b denotes the b-th sequence in a batch, i denotes the i-th skeleton graph in the

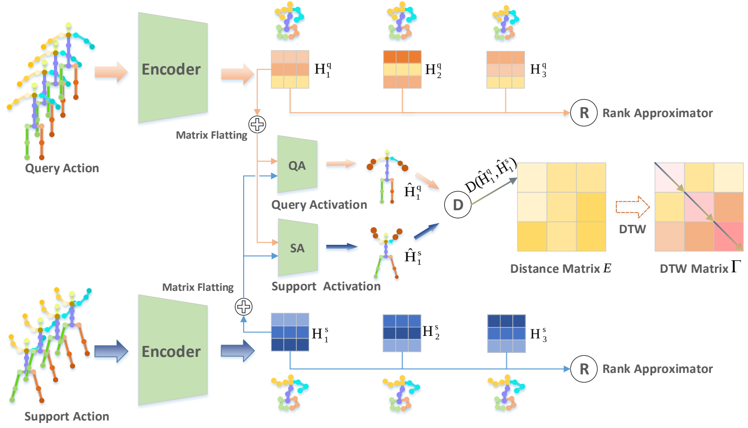


Fig. 2: The Illustration that describes 1-shot action recognition with our framework. $\mathbf{H}^q$ and $\mathbf{H}^s$ are the outputs of *identical* spatial-temporal encoder like ST-GCN [42]. The parameterized modules QA and SA denote query activation and support activation respectively. For clarity, we only demonstrate the matching process of the first skeleton representations, i.e., $\mathbf{H}_1^q$ and $\mathbf{H}_1^s$.

sequence, n is the number of node and d is the node's latent dimension, we can maximize its rank to reduce the dependencies among skeletons joints. Directly maximizing the rank of $\mathbf{H}_{bi}$ is a NP-hard problem [31]. A common solution is to use nuclear-norm $||\mathbf{H}_{bi}||_*$ as a surrogate for rank$(\mathbf{H}_{bi})$ [9, 31]:

$$||\mathbf{H}_{bi}||_* = (\sum_{j}^{\min(n,d)} \sigma_i^j) < rank(\mathbf{H}_{bi}), \tag{2}$$

where σ_i^j is the j-th singular value of matrix $\mathbf{H}_{bi}$ and can be calculated through Singular Value Decomposition (SVD) [3]. Pytorch[7] has released a differential SVD tool for for convenient implementation. The constraint can be applied to a batch of skeleton sequences, forming the spatial disentanglement objective:

$$\mathcal{L}_{dis} = -\frac{1}{B * m} \sum_{b}^{B} \sum_{i}^{m} ||\mathbf{H}_{bi}||_*, \tag{3}$$

where B is the number of action sequences in a batch, m is the length of a sequence after temporal convolution.

Learning Spatial Activation. Recovering the spatial representations grounds the spatial matching between the query graph representation $\mathbf{H^q} \in \mathbb{R}^{n \times d}$ and the support graph representation $\mathbf{H^s} \in \mathbb{R}^{n \times d}$. A direct solution to match the query

[7] https://pytorch.org/

graph and support graph is to calculate their Euclidean distance. Based on this, we further consider the harder situation: when matching two similar actions such as squatting down and sitting down, the spatial relation of joints plays an important role to distinguish the two actions. To further pay attention to the important joints, we start with transforming the query's spatial representation according to the representations of support skeletons. This means that the node representations should better be adaptively changed according to different query-support graph pairs. Specifically, when given a query graph representation $\mathbf{H^q}$, the transformed version can be produced as:

$$\hat{\mathbf{H}}^q = \mathrm{S}\left(\frac{\mathbf{W}_1^q\mathbf{H}^q \cdot [\mathbf{W}_2^q\mathbf{H}^s]^T}{\sqrt{d}}\right)\mathbf{W}_3^q\mathbf{H}^q, \tag{4}$$

where $\mathrm{S}(\cdot)$ is the Softmax function, $\mathbf{W}_1^q, \mathbf{W}_2^q, \mathbf{W}_3^q$ are the the transformation matrices and d is the dimension of node representation. Similarly, $\mathbf{H^s}$ can be transformed as:

$$\hat{\mathbf{H}}^s = \mathrm{S}\left(\frac{\mathbf{W}_1^s\mathbf{H}^s \cdot [\mathbf{W}_2^s\mathbf{H}^q]^T}{\sqrt{d}}\right)\mathbf{W}_3^s\mathbf{H}^s, \tag{5}$$

Then the distance of two graphs can be defined as:

$$D(\hat{\mathbf{H}}^q, \hat{\mathbf{H}}^s) = ||\hat{\mathbf{H}}^q - \hat{\mathbf{H}}^s||_F, \tag{6}$$

where $||\cdot||_F$ is the Frobenius norm. The cross-attention form Eq. (4) and Eq. (5) is inspired by the Transformer [37], but learns different inductive bias compared with popular cross-attention methods. Previous cross-attentions usually focus on one of the two parties involved in the measurement process, e.g., only learning the inductive bias for support prototype instead of query examples [11]. We suppose that the $\mathbf{H}^s$ and $\mathbf{H}^q$ have different activation patterns because of asymmetrical calculation in measurement, e.g., the $\mathbf{H}^s$ in Eq. (5) is usually a weighted average representation corresponding to $\mathbf{H}^q$. Hence learning the inductive bias each side with Eq. (4) and Eq. (5) may stimulate different activation patterns.

4.2 Temporal Matching

Given a transformed query sequence $Q = \{\hat{\mathbf{H}}_1^q, \hat{\mathbf{H}}_2^q, \cdots \hat{\mathbf{H}}_m^q\}$ and a support sequence $C_k = \{\hat{\mathbf{H}}_1^s, \hat{\mathbf{H}}_2^s, \cdots \hat{\mathbf{H}}_m^s\}$, there may be the misalignment that $\hat{\mathbf{H}}_i^q$ does not corresponding to $\hat{\mathbf{H}}_i^s$. Therefore, directly calculating the sequence distance with the sum of graph pairs $D(\hat{\mathbf{H}}_i^q, \hat{\mathbf{H}}_i^s)$ will impact the distance measurement between two action sequences. To tackle this misalignment, we introduce Dynamic Time Warping (DTW, [34]), which is a popular temporal alignment method with multiple variants [12,27]. Concretely, for a query-support pair, we can get a distance matrix $E \in \mathbb{R}^{m \times m}$, where each element E_{ij} is calculated with $D(\hat{\mathbf{H}}_i^q, \hat{\mathbf{H}}_j^s)$. Supposing the cumulative distance from a query frame i to a support frame j can be calculated with the following dynamic programming form:

$$\mathit{\Gamma}(i,j) = E(i,j) + \min\{\mathit{\Gamma}(i-1,j-1), \mathit{\Gamma}(i-1,j), \mathit{\Gamma}(i,j-1)\}, \tag{7}$$

Algorithm 1 Training Algorithm for Few-Shot Action Recognition

```
Input: The training data $\mathcal{T}_{train}$
Parameter: Model parameters $\theta$, including action encoder parameters $\phi$, spatial
transformation parameters sets $\{\mathbf{W}\} = \mathbf{W}_1^q, \mathbf{W}_2^q, \mathbf{W}_3^q, \mathbf{W}_1^s, \mathbf{W}_2^s, \mathbf{W}_3^s$.
Output: The learned $\theta$ and $\{\mathbf{W}\}$.

while not convergence do
    for $step = 0 \rightarrow T$ do
        Sample a N-way-K-shot classification task with query actions $\{\mathcal{G}_i^q, Y_i^q\}_{i=1}^{N^q}$ and
        support actions $\{\mathcal{G}_i^s, Y_i^s\}_{i=1}^{N^s}$
        Compute all support action prototypes $\{C_k\}_{k=1}^{N}$ using $\{\mathcal{G}_i^s, Y_i^s\}_{i=1}^{N^s}$, where
        $C_k = \{\mathbf{H}_1^s, \mathbf{H}_2^s, \cdots \mathbf{H}_m^s\}$
        Compute all query action representations $\{Q_i\}_{i=1}^{N^q}$, $Q_i = \{\mathbf{H}_1^q, \mathbf{H}_2^q, \cdots, \mathbf{H}_m^q\}$
        for each $Q_i$, compute its distance with each support sequence $C_k$ by Algo-
        rithm 2
        Compute the labels for all query actions with Eq. (1)
        For all query actions, compute $\mathcal{L}_{dis}$ with Eq. (3)
        Compute $\mathcal{L}_{match}$ via Eq. (8)
        Update $\theta$ and $\{\mathbf{W}\}$ by $\mathcal{L}_{match}$.
    end for
end while
```

hence we can utilize $d(Q, C_k) = \Gamma(m, m)$ as the sequence distance. To get a differentiable distance, the minimization function $min(\cdot, \cdot)$ can be replaced with a differentiable version [27].

4.3 The Learning Objective

According to Eq. (1) and Eq. (3), the overall learning objective can be derived:

$$\mathcal{L}_{match} = -\frac{1}{N^q} \sum_{i}^{N^q} \log p_\phi(\hat{y}_i = y_i \mid \mathcal{G}_i^q) + \lambda \mathcal{L}_{dis}, \tag{8}$$

where N^q is the number of query actions, $\hat{y}_i$ and y_i is the predicted label and ground truth label for $\mathcal{G}_i^q$, λ denotes the weight of $\mathcal{L}_{dis}$. Algorithm 1 and Algorithm 2 demonstrate the overall training process of our algorithm. Fig. 2 illustrates the graphical framework.

5 Experiments

In this section, we will evaluate our approach and baselines on two public large scale datasets, trying to answer the following questions: (1) What's the performance of primitive baselines like using ST-GCN [42]+ProtoNet [36] without any spatial-temporal alignment? (2) Is our rank maximization strategy working better than typical methods that tackle the over-smoothness of ST-GCNs? (3) How does the activation strategy work for each part of the skeletons?

Algorithm 2 Matching Algorithm for Skeleton Sequences

Input: A query action representation $Q = \{\mathbf{H}_1^q, \mathbf{H}_2^q, \cdots, \mathbf{H}_m^q\}$, and a support prototype $C_k = \{\mathbf{H}_1^s, \mathbf{H}_2^s, \cdots \mathbf{H}_m^s\}$
Parameter: Spatial transformation parameters $\mathbf{W}_1^q, \mathbf{W}_2^q, \mathbf{W}_3^q, \mathbf{W}_1^s, \mathbf{W}_2^s, \mathbf{W}_3^s$
Output: $\Gamma(m, m)$, the distance between Q and C_k

```
Initialize distance matrix $E \in \mathbb{R}^{m \times m}$
for $i = 0 \to m$ do
    for $j = 0 \to m$ do
        Compute $\hat{\mathbf{H}}_i^q$ with $\mathbf{H}_i^q$ and $\mathbf{H}_j^s$ using Eq. (4)
        Compute $\hat{\mathbf{H}}_j^s$ with $\mathbf{H}_j^s$ and $\mathbf{H}_i^q$ using Eq. (5)
        $E_{ij} = D(\hat{\mathbf{H}}_i^q, \hat{\mathbf{H}}_j^s)$ with Eq. (6)
    end for
end for
Compute accumulation distance matrix $\Gamma$ with $E$ using Eq. (7)
```

5.1 Datasets

We firstly introduce the used datasets including NTU RGB+D 120 [22] and Kinetics [17].

NTU RGB+D 120 [22] is a large-scale dataset with 3D joints annotations for human action recognition tasks, containing 113,945 skeleton sequences with 25 body joints for each skeleton. In our experiments, we use 120 action categories, including 80, 20 and 20 categories as training, validation and test categories. For each category, we randomly take 60 samples and 30 samples, denoted as two subsets "**NTU-S**" and "**NTU-T**", respectively. Please see our Appendix A for more details.

Kinetics skeleton dataset [17] is sourced from YouTube videos. The dataset contains 260,232 videos over 400 classes, where each skeleton graph has 18 body joints after pose estimation, along with their 2D spatial coordinates and the prediction confidence score from OpenPose [6] as the initial joint features. In our experiments, we only use the first 120 actions with 100 samples per class. The number of training/validation/test partitions is identical to NTU RGB+D 120 (please see our Appendix A for more detailed separation).

5.2 Baselines

The baselines includes the following categories for few-shot action recognition: **ProtoNet** [36]; temporal alignment **DTW** [34]; the methods for spatial recover , such as **PairNorm** [45] and **DropEdge** [32]; spatial alignment or graph metric learning like **NGM** [15]. Note that all the above methods use ProtoNet as classifier head for few-shot action recognition. Besides that, the baselines PairNorm, DropEdge and NGM are combined with temporal alignment DTW.

ProtoNet [36] treats the representation of action sequences as vectors and computes the Euclidean similarity of the vectors without any spatial matching and temporal alignment.

DTW calculates an optimal match between two sequences using dynamic programming, and is a popular temporal alignment method adopted by previous video-based few-shot action recognition.

PairNorm [45] aims to tackle over-smoothing in graph neural networks (GNNs). By centering and rescaling node's representations, PairNorm uses a normalization after graph convolution layer to prevent all node embeddings from becoming too similar.

DropEdge [32] randomly drops a few edges in input graphs to make nodes aggregation diverse from their neighbors. Both PairNorm and DropEdge are designed to alleviate the over-smoothing problem in large noisy graphs. To the best of our knowledge, the two strategies are the first to be applied to skeleton graphs.

NGM [15] jointly learns a graph generator and a graph matching metric function in an end-to-end fashion to optimize the few-shot learning objective.

DASTM* and **DASTM**** denote our ablation models with Rank Maximization and Spatial Activation, respectively.

5.3 Implementation Details

Data preparation. We randomly sample N classes with each class containing K actions as the support set. Correspondingly, we have N categories including K actions of query set, where the query set has the same classes as the support set. Thus each episode has a total of $N \times (K + K)$ examples as a training or test batch. For each skeleton sequence, we pre-process the skeleton sequences following pre-processing video procedure as TSN [39]. For different datasets, we uniformly sample 50 and 30 frames per skeleton sequence in Kinetics and NTU-T/S. This uniform sampling provides identical sequence lengths for the support and query actions.

Spatial-temporal backbones. To encode the action skeleton sequence, we adopt typical **ST-GCN** [42], **2s-AGCN** [35] and **MS-G3D** [23] as the backbones. ST-GCN proposes spatial-temporal graph convolution on skeleton sequences. 2s-AGCN uses joints and bones information to learn data-dependent graph structure. MS-G3D proposes multi-scale aggregation scheme to disentangle the importance of nodes in different neighborhoods and cross-spacetime edges to capture high-level node interaction. Appendix C provides smaller backbones and corresponding performances.

Model training and evaluation All models are trained with Adam [18] optimizer, using an initial learning rate 0.001. The weight λ of $\mathcal{L}_{dis}$ is set with 0.1 according to the validation sets for all experiments. With randomly sampling 1,000 episodes in training and 500 episodes in test, each experiment is repeated **3** times to calculate mean accuracy with standard deviation. Furthermore, all experiments are constructed using Pytorch and performed on Ubuntu 18.04 with one GeForce RTX 3090 GPU.The training codes can be found here [8]. Each training task may need about 10h.

5.4 Results

As shown in Tab. 1 and Tab. 2, we compare our method with mentioned baselines using 3 datasets and 3 backbones. 5-way-k-shot denotes performing 5-way classification using k labeled support example per class.

Strong baselines are constructed for skeleton-based few-shot action recognition. With the spatial-temporal convolution, only using ProtoNet classifier can produce 71.2% 1-shot classification accuracy on NTU-T (see ProtoNet baseline in Tab. 1), even though this dataset only contains 30 action examples per class. This performance demonstrates the potential of simply spatial-temporal convolution for skeleton-based few-shot action recognition.

Generalized methods coping over-smoothing may harm the few-shot task. We perform two comparison experiments containing DropEdge and PairNorm (denoted with DropEdge and PairNorm in Tab. 1 and Tab. 2). For DropEdge, we find only randomly dropping 4 edges already destroys the small skeleton graphs and harms the improvement brought by DTW. In contrast, PairNorm makes the adjacent nodes have similar representations with pushing away the non-adjacent nodes. However, making the adjacent nodes have similar representations may smooth the key joints' features such as "hands" and "elbows", which harms the matching process (see PairNorm and DTW in Tab. 1).

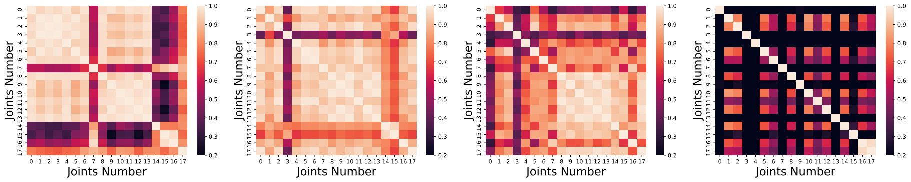
Table 1: The 5-way-1-shot action classification accuracies (%).

Backbones	ST-GCN			2s-AGCN			MS-G3D		
Methods	NTU-T	NTU-S	Kinetics	NTU-T	NTU-S	Kinetics	NTU-T	NTU-S	Kinetics
ProtoNet	71.2$_{\pm 1.5}$	73.3$_{\pm 0.3}$	37.4$_{\pm 0.4}$	68.1$_{\pm 0.5}$	72.8$_{\pm 0.3}$	38.4$_{\pm 0.2}$	70.1$_{\pm 1.0}$	73.6$_{\pm 0.2}$	39.5$_{\pm 0.3}$
DTW	74.0$_{\pm 2.1}$	73.5$_{\pm 0.4}$	39.2$_{\pm 0.2}$	70.8$_{\pm 1.4}$	71.5$_{\pm 1.2}$	40.9$_{\pm 0.4}$	72.4$_{\pm 0.2}$	73.9$_{\pm 0.4}$	40.6$_{\pm 0.2}$
NGM	71.8$_{\pm 1.2}$	75.7$_{\pm 0.4}$	39.1$_{\pm 0.3}$	72.2$_{\pm 1.0}$	73.2$_{\pm 0.6}$	40.9$_{\pm 0.2}$	73.5$_{\pm 0.3}$	**76.9**$_{\pm 0.4}$	40.8$_{\pm 0.3}$
PairNorm	72.9$_{\pm 0.5}$	72.8$_{\pm 0.4}$	39.3$_{\pm 0.7}$	70.0$_{\pm 0.5}$	70.8$_{\pm 0.3}$	40.9$_{\pm 0.3}$	71.0$_{\pm 0.8}$	70.8$_{\pm 0.9}$	40.7$_{\pm 0.7}$
DropEdge	67.3$_{\pm 1.9}$	70.7$_{\pm 0.7}$	38.9$_{\pm 0.9}$	70.1$_{\pm 0.4}$	72.6$_{\pm 0.2}$	39.9$_{\pm 0.3}$	68.7$_{\pm 0.4}$	69.5$_{\pm 0.7}$	39.4$_{\pm 0.3}$
DASTM*	74.5$_{\pm 1.9}$	73.4$_{\pm 0.6}$	39.5$_{\pm 0.9}$	72.4$_{\pm 0.9}$	72.9$_{\pm 0.5}$	40.9$_{\pm 0.4}$	72.7$_{\pm 0.6}$	74.4$_{\pm 1.6}$	40.7$_{\pm 0.2}$
DASTM**	74.4$_{\pm 2.9}$	75.9$_{\pm 0.3}$	**39.8**$_{\pm 0.1}$	72.9$_{\pm 1.5}$	**74.6**$_{\pm 0.3}$	**41.0**$_{\pm 0.1}$	74.1$_{\pm 0.3}$	75.5$_{\pm 1.7}$	41.0$_{\pm 0.1}$
DASTM	**75.1**$_{\pm 1.8}$	**76.2**$_{\pm 0.3}$	39.3$_{\pm 0.1}$	**73.3**$_{\pm 0.6}$	74.0$_{\pm 0.7}$	40.8$_{\pm 0.3}$	**75.0**$_{\pm 0.9}$	76.3$_{\pm 1.2}$	**41.1**$_{\pm 0.2}$

Table 2: The 5-way-5-shot action classification accuracies (%).

Backbones	ST-GCN			2s-AGCN			MS-G3D		
Methods	NTU-T	NTU-S	Kinetics	NTU-T	NTU-S	Kinetics	NTU-T	NTU-S	Kinetics
ProtoNet	81.1$_{\pm 0.2}$	84.3$_{\pm 0.3}$	46.8$_{\pm 0.4}$	81.9$_{\pm 0.1}$	84.2$_{\pm 0.1}$	50.5$_{\pm 0.2}$	82.3$_{\pm 0.2}$	85.3$_{\pm 0.1}$	50.0$_{\pm 0.3}$
DTW	81.0$_{\pm 0.6}$	81.5$_{\pm 0.5}$	47.9$_{\pm 0.3}$	81.2$_{\pm 0.9}$	82.5$_{\pm 0.8}$	50.8$_{\pm 0.3}$	81.3$_{\pm 0.3}$	83.2$_{\pm 0.4}$	50.0$_{\pm 0.2}$
NGM	81.4$_{\pm 0.5}$	84.2$_{\pm 0.4}$	48.6$_{\pm 0.4}$	83.2$_{\pm 0.3}$	85.9$_{\pm 0.4}$	49.8$_{\pm 0.3}$	83.1$_{\pm 0.4}$	86.7$_{\pm 0.2}$	50.7$_{\pm 0.3}$
PairNorm	81.8$_{\pm 0.4}$	81.4$_{\pm 0.3}$	48.6$_{\pm 0.5}$	80.0$_{\pm 0.3}$	80.3$_{\pm 0.2}$	50.4$_{\pm 0.4}$	81.6$_{\pm 0.7}$	82.5$_{\pm 0.9}$	50.6$_{\pm 0.1}$
DropEdge	77.9$_{\pm 1.5}$	78.6$_{\pm 0.7}$	48.2$_{\pm 0.2}$	80.5$_{\pm 0.3}$	83.1$_{\pm 0.6}$	50.2$_{\pm 0.3}$	80.9$_{\pm 0.4}$	80.2$_{\pm 0.6}$	50.1$_{\pm 0.2}$
DASTM*	81.8$_{\pm 0.6}$	82.4$_{\pm 0.5}$	48.8$_{\pm 0.2}$	81.8$_{\pm 0.3}$	83.6$_{\pm 0.6}$	51.0$_{\pm 0.1}$	81.9$_{\pm 0.1}$	84.1$_{\pm 0.6}$	**51.3**$_{\pm 0.2}$
DASTM**	82.9$_{\pm 0.8}$	85.3$_{\pm 0.7}$	**49.2**$_{\pm 1.0}$	83.5$_{\pm 0.4}$	86.3$_{\pm 0.7}$	**51.3**$_{\pm 0.6}$	84.2$_{\pm 0.5}$	86.4$_{\pm 1.6}$	51.1$_{\pm 0.5}$
DASTM	**83.0**$_{\pm 0.1}$	**85.5**$_{\pm 0.3}$	48.9$_{\pm 0.1}$	**83.8**$_{\pm 0.8}$	**86.8**$_{\pm 0.3}$	50.9$_{\pm 0.2}$	**84.9**$_{\pm 0.3}$	**87.3**$_{\pm 1.2}$	51.1$_{\pm 0.9}$

[8] https://github.com/NingMa-AI/DASTM



(a) None (b) PairNorm (c) DropEdge (d) RankMax(Ours)

Fig. 3: The intra-skeleton joints similarity heatmaps according to different spatial preserving strategies (see more visualizations in our Appendix D). All the skeletons are sampled from Kinetics dataset with 18 joints per skeleton. (a). None, no spatial preserving strategy is used. Almost all the upper limb joints (number 0-6) and lower limb joints (number 8-13) are distinguishable. (b). PairNorm method produces similar representations for the upper limb joints (number 5-7) and lower limb joints (8-13), due to its centering and rescaling operations. (c). DropEdge, randomly dropping edges of skeleton graphs can not alleviate the smoothness representations such as lower limb joints (number 8-13). (d). Our RankMax method, the representative joints such as hands (number 4 and 7) and feet (number 13) are disentangled from other joints, providing more specific spatial features for skeleton matching.

The heatmaps in Fig. 3 also illustrate the joints' smoothness with PairNorm and DropEdge. An extreme case happens on Kinetics dataset, on which nearly all methods (including ours) do not get significant improvements compared with DTW. One of the reasons might be the large data noise in Kinetics, in which a large proportion of actions even can not be efficiently distinguished via skeletons. Given noisy skeleton distance, the optimal alignment path for DTW is fragile and the improvements for DTW are overwhelmed by this noise. Our method still successfully obtain gains for DTW, while there are no performance improvements or even drops over DTW by PairNorm, Dropedge on Kinetics.

Comparison with existing skeleton-based few-shot methods. NGM proposes the first skeleton-based few-shot recognition method with graph matching. We tried to implement an enhanced version with deeper edge-weight learning layers in ST-GCNs, adding time alignment via DTW and learning a more powerful graph tensor with Transformer's self-attention. Although these additional technics result in a strong baseline, we find that our mutual-activation between query and support skeletons still works better than the self-activation in NGM.

5.5 Ablation Studies

In this section, we demonstrate how each component contributes to the overall performances.

Analysis of spatial disentanglement. In our framework, the disentanglement of spatial information is achieved by maximizing the rank of skeleton representation matrix. Compared with temporal alignment baseline DTW, this

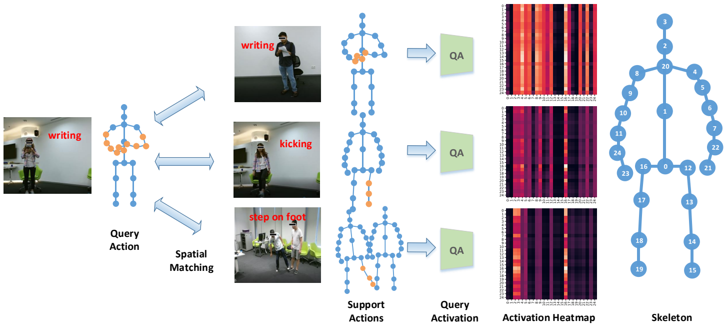


Fig. 4: The spatial activation illustration containing query-support pairs and corresponding inter-skeleton activation heatmaps on NTU-T dataset (best view in color). Each pair represents a matching process between a query action and a support action, and each heatmap is derived from the output of the Softmax function in Eq. (4).

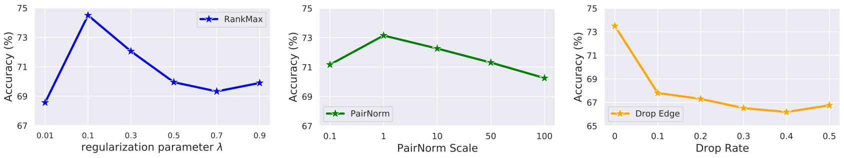
disentanglement strategy achieves up to 1.4% improvements on NTU-T with 2s-AGCN (see Tab. 1). The reason behind is that our rank maximizing strategy encourages skeleton joints to have independent representation, which provides more spatial information for skeleton matching. For example, in Fig. 3d, the hand joints (number 4, 7) and foot joints (number 13) are the disentangled key parts to describe one action. This result also demonstrates that the spatial information was not fully considered in previous skeleton-based few-shot action recognition.

The explainability of spatial activation. To identify the intra-class and inter-class spatial activation patterns, we collect the outputs of the Softmax function in Eq. (4) and visualize them with heatmap (see Fig. 4). In the first matching pair "writing"-"writing", the query's upper limb joints (number 2-6 and 8-11) are activated. In the pair "writing"- "kicking", we can observe that the query's activation has much lower responses, which indicates the query action and support action may belong to different classes. We hope the explainability will bring more interesting works in future action recognition tasks.

Analysis of temporal alignment. To demonstrate the improvement of our spatial matching strategy, we also perform ablation studies on temporal alignment (see DTW in Tab. 1). Based on DTW, our spatial matching strategy gets up to 3.5% improvement on 1-shot tasks. However, for 5-shot tasks (see DTW in Tab. 2), DTW may be suboptimal compared with ProtoNet due to its sensibility to skeleton noise, which may bring suboptimal matching path in DTW. Tackling the fragility of DTW falls slightly out of the scope of this study. One of our future works is to enhance the robustness for DTW's path selection.

5.6 Hyper-Parameter Analyses

The degree of spatial disentanglement, PairNorm Scale and Drop Rate are three key hyper-parameters in our framework, PairNorm and Dropedge, respectively. Fig. 5a illustrates the accuracy changing with different weight λ using DASTM* consisting of temporal alignment DTW and Rank Maximization. When $\lambda = 0.01$ is small, the model gets much lower performance due to limited spatial disentanglement. When λ is close to 1, the compulsive disentanglement of all joints damages the original spatial relation. This failed situation also verifies our claim in the previous section: only a part of joints that maintain critical positional information need to be disentangled to help skeleton matching. In practice, we find disentangling a few joints is more helpful for skeleton matching, e.g., about 7 joints are disentangled in Fig. 3d. We suggest using a small λ like 0.1 or 0.05 to encourage the model to adaptively select a part critical joints. Besides that, Fig. 5b and Fig. 5c demonstrate two baselines' hyper-parameter sensibility (please see the Appendix B for more details).



(a) RankMax (Ours) (b) PairNorm (c) DropEdge

Fig. 5: The hyper-parameter sensibility for RankMax(ours), PairNorm and DropEdge, respectively. All tasks are 5-way-1-shot on NTU-T using ST-GCN.

6 Conclusion and Future Works

We propose a novel skeleton representation and matching solution for few-shot action recognition. The proposed method tries to capture key joints from the disentanglement view, hence bring more explainability for concrete few-shot action recognition tasks. Compared with the well studied video-based solutions, it is the first time exploring skeleton-based few-shot action recognition with the powerful representation ability of modern ST-GCNs. We hope more skeleton-based few-shot works can be explored in the future.

Acknowledgement: This work is supported by the National Key Research and Development Program (Grant No. 2019YFF0302601), National Natural Science Foundation of China (Grant No: 61972349, 62106221) and Multi-Center Clinical Research Project in National Center (No. S20A0002).

References

1. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv:1803.01271 (2018)
2. Ben-Ari, R., Nacson, M.S., Azulai, O., Barzelay, U., Rotman, D.: Taen: Temporal aware embedding network for few-shot action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 2786–2794 (June 2021)
3. Bhatia, R.: Matrix analysis (2013)
4. Cao, C., Li, Y., Lv, Q., Wang, P., Zhang, Y.: Few-shot action recognition with implicit temporal alignment and pair similarity optimization (2020)
5. Cao, K., Ji, J., Cao, Z., Chang, C.Y., Niebles, J.C.: Few-shot video classification via temporal alignment. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10615–10624 (2020). https://doi.org/10.1109/CVPR42600.2020.01063
6. Cao, Z., Hidalgo, G., Simon, T., Wei, S.E., Sheikh, Y.: Openpose: realtime multi-person 2d pose estimation using part affinity fields. IEEE transactions on pattern analysis and machine intelligence **43**(1), 172–186 (2019)
7. Careaga, C., Hutchinson, B., Hodas, N., Phillips, L.: Metric-based few-shot learning for video action recognition (2019)
8. Chen, Y., Zhang, Z., Yuan, C., Li, B., Deng, Y., Hu, W.: Channel-wise topology refinement graph convolution for skeleton-based action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 13359–13368 (October 2021)
9. Cui, S., Wang, S., Zhuo, J., Li, L., Huang, Q., Tian, Q.: Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
10. Cuturi, M., Blondel, M.: Soft-dtw: A differentiable loss function for time-series. In: Proceedings of the 34th International Conference on Machine Learning - Volume 70. p. 894–903. ICML'17, JMLR.org (2017)
11. Doersch, C., Gupta, A., Zisserman, A.: Crosstransformers: spatially-aware few-shot transfer. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 21981–21993. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper/2020/file/fa28c6cdf8dd6f41a657c3d7caa5c709-Paper.pdf
12. Dvornik, N., Hadji, I., Derpanis, K.G., Garg, A., Jepson, A.D.: Drop-dtw: Aligning common signal between sequences while dropping outliers. NeurIPS (2021)
13. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 1126–1135. PMLR (06–11 Aug 2017), https://proceedings.mlr.press/v70/finn17a.html
14. Fu, Y., Zhang, L., Wang, J., Fu, Y., Jiang, Y.G.: Depth guided adaptive meta-fusion network for few-shot video recognition. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 1142–1151 (2020)
15. Guo, M., Chou, E., Huang, D.A., Song, S., Yeung, S., Fei-Fei, L.: Neural graph matching networks for fewshot 3d action recognition. In: Proceedings of the European conference on computer vision (ECCV). pp. 653–669 (2018)

16. Hong, J., Fisher, M., Gharbi, M., Fatahalian, K.: Video pose distillation for few-shot, fine-grained sports action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9254–9263 (October 2021)
17. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015), http://arxiv.org/abs/1412.6980
19. Konecny, J., Hagara, M.: One-shot-learning gesture recognition using hog-hof features. Journal of Machine Learning Research **15**(72), 2513–2532 (2014), http://jmlr.org/papers/v15/konecny14a.html
20. Li, S., Liu, H., Qian, R., Li, Y., See, J., Fei, M., Yu, X., Lin, W.: Ta2n: Two-stage action alignment network for few-shot action recognition (2021)
21. Lin, J., Gan, C., Wang, K., Han, S.: Tsm: Temporal shift module for efficient video understanding. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 7082–7092 (2019)
22. Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.Y., Kot, A.C.: Ntu rgb+d 120: A large-scale benchmark for 3d human activity understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(10), 2684–2701 (2020)
23. Liu, Z., Zhang, H., Chen, Z., Wang, Z., Ouyang, W.: Disentangling and unifying graph convolutions for skeleton-based action recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 143–152 (2020)
24. Luo, D., Cheng, W., Yu, W., Zong, B., Ni, J., Chen, H., Zhang, X.: Learning to drop: Robust graph neural network via topological denoising. In: Proceedings of the 14th ACM International Conference on Web Search and Data Mining. p. 779–787. WSDM '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3437963.3441734, https://doi.org/10.1145/3437963.3441734
25. Memmesheimer, R., Häring, S., Theisen, N., Paulus, D.: Skeleton-dml: Deep metric learning for skeleton-based one-shot action recognition. arXiv preprint arXiv:2012.13823 (2020)
26. Ni, X., Song, S., Tai, Y.W., Tang, C.K.: Semi-supervised few-shot atomic action recognition (2020)
27. Nielsen, F., Sun, K.: Guaranteed bounds on information-theoretic measures of univariate mixtures using piecewise log-sum-exp inequalities. Entropy **18**(12) (2016). https://doi.org/10.3390/e18120442, https://www.mdpi.com/1099-4300/18/12/442
28. Patravali, J., Mittal, G., Yu, Y., Li, F., Chen, M.: Unsupervised few-shot action recognition via action-appearance aligned meta-adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 8484–8494 (October 2021)
29. Perrett, T., Masullo, A., Burghardt, T., Mirmehdi, M., Damen, D.: Temporal-relational crosstransformers for few-shot action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 475–484 (June 2021)

30. Qi, M., Qin, J., Zhen, X., Huang, D., Yang, Y., Luo, J.: Few-shot ensemble learning for video classification with slowfast memory networks. In: Proceedings of the 28th ACM International Conference on Multimedia. p. 3007–3015. MM '20, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3394171.3416269, https://doi.org/10.1145/3394171.3416269
31. Recht, B., Fazel, M., Parrilo, P.A.: Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. SIAM Review **52**(3), 471–501 (2010). https://doi.org/10.1137/070697835
32. Rong, Y., Huang, W., Xu, T., Huang, J.: Dropedge: Towards deep graph convolutional networks on node classification. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=Hkx1qkrKPr
33. Sabater, A., Santos, L., Santos-Victor, J., Bernardino, A., Montesano, L., Murillo, A.C.: One-shot action recognition in challenging therapy scenarios. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 2777–2785 (June 2021)
34. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing **26**(1), 43–49 (1978). https://doi.org/10.1109/TASSP.1978.1163055
35. Shi, L., Zhang, Y., Cheng, J., Lu, H.: Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12026–12035 (2019)
36. Snell, J., Swersky, K., Zemel, R.S.: Prototypical networks for few-shot learning. arXiv preprint arXiv:1703.05175 (2017)
37. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017), https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
38. Wang, J., Wang, Y., Liu, S., Li, A.: Few-shot fine-grained action recognition via bidirectional attention and contrastive meta-learning. In: Proceedings of the 29th ACM International Conference on Multimedia. p. 582–591. MM '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3474085.3475216, https://doi.org/10.1145/3474085.3475216
39. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision – ECCV 2016. pp. 20–36. Springer International Publishing, Cham (2016)
40. Wang, X., Ye, W., Qi, Z., Zhao, X., Wang, G., Shan, Y., Wang, H.: Semantic-guided relation propagation network for few-shot action recognition. In: Proceedings of the 29th ACM International Conference on Multimedia. p. 816–825. MM '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3474085.3475253, https://doi.org/10.1145/3474085.3475253
41. Xian, Y., Korbar, B., Douze, M., Schiele, B., Akata, Z., Torresani, L.: Generalized many-way few-shot video classification. In: Bartoli, A., Fusiello, A. (eds.) Computer Vision – ECCV 2020 Workshops. pp. 111–127. Springer International Publishing, Cham (2020)

42. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Thirty-second AAAI conference on artificial intelligence (2018)
43. Zhang, H., Zhang, L., Qi, X., Li, H., Torr, P., Koniusz, P.: Few-shot action recognition with permutation-invariant attention. In: Proceedings of the European Conference on Computer Vision (ECCV 2020). vol. 12350. Springer (2020)
44. Zhang, S., Zhou, J., He, X.: Learning Implicit Temporal Alignment for Few-shot Video Classification. IJCAI (2021)
45. Zhao, L., Akoglu, L.: Pairnorm: Tackling oversmoothing in gnns. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=rkecl1rtwB
46. Zhu, L., Yang, Y.: Compound memory networks for few-shot video classification. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)
47. Zhu, Z., Wang, L., Guo, S., Wu, G.: A closer look at few-shot video classification: A new baseline and benchmark (2021)