IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 24 January 2024, accepted 1 February 2024, date of publication 12 February 2024, date of current version 4 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3365448

RESEARCH ARTICLE

Enhancing Few-Shot Action Recognition Using Skeleton Temporal Alignment and Adversarial Training

QINGYANG XU[1,*], JIANJUN YANG[2,*], HONGYI ZHANG[3], XIN JIE[1], AND DANUSHKA BANDARA[4]

[1]College of Computer Science, Zhejiang University, Hangzhou 310000, China
[2]Department of General Practice, Shandong Provincial Third Hospital, Shandong University, Jinan, Shandong 250031, China
[3]School of Software Technology, Zhejiang University, Hangzhou 310000, China
[4]Department of Computer Science and Engineering, Fairfield University, Fairfield, CT 06824, USA

Corresponding authors: Jianjun Yang (qlbsh1@163.com) and Danushka Bandara (dbandara@fairfield.edu)

This work is supported by the Natural Science Foundation of Shandong Province (ZR2021MH227), Jinan Science and Technology Bureau plan (202019181), and the National Natural Science Foundation of China (62071330).

*Qingyang Xu and Jianjun Yang are co-first authors.

ABSTRACT Few-shot human action recognition, a prominent area in computer vision, has garnered increasing attention and broader use in real-life scenarios. Extracting spatio-temporal skeletal information from human movement videos offers interpretable and data-efficient features. However, existing spatio-temporal feature encoders face challenges such as handling sequence boundaries and coping with noise. In order to solve the above problems, this paper proposes a temporal complement method to optimize the Dynamic Time Warping (DTW) algorithm based on the feature representation of the human skeleton sequence. DTW helps to find optimal alignment between sequences by warping them in the time domain. This is quite useful specially in scenarios where training data is limited. However, DTW has the drawback that the optimal alignment path is highly sensitive to errors in the time series distance matrix. Therefore, we apply a Virtual Adversarial Training method to improve the anti-noise capability of the algorithm. Here, We introduce adversarial perturbations in the training phase to the time series distance matrix, thus incentivizing the model to be resilient to such noise. Our method achieves highest accuracy among protonet, DTW and DASTM methods for the 5-way-1-shot setting for the NTU-S (77.7%), and Kinetics (41.2%) datasets. For the 5-way-5-shot setting, our method achieves highest accuracy of 51.8% for Kinetics dataset when compared with the other approaches.

INDEX TERMS Action recognition, few-shot learning, temporal alignment, adversarial training.

I. INTRODUCTION

Human action recognition stands as a prominent subject within computer vision. It involves analyzing human behavior in videos to detect and categorize distinct types of actions [1], [2], [3]. By understanding human movements and behaviors in videos, computers can accomplish various downstream tasks, such as human-computer interaction systems, human behavior tracking, human behavior monitoring, etc. [4] Directly detecting human action from a video poses several challenges. Under normal circumstances, it is difficult for the computer to understand the human movement in the video due to human or other interference factors, such as the change in lighting conditions, different acquisition angles, and background changes. Therefore, the analysis method based directly on video information has not achieved good results in human motion recognition tasks. In 1973, Johansson [5] found through experiments that the movement information of the human body could be represented by abstracting into multiple bone points, and the movement types of different human bodies could be distinguished by recording the spatial position information of bone points and the changes of bone points in time series to indicate the movement types. In recent years, with

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa M. Fouda.

the rapid development of depth camera devices such as Kinect [6] and RealSense [7], such systems can obtain the position of key points of the human body and capture the position information of human bone points based on depth maps [8]. Bone point information is robust because light conditions and background information do not affect it. Currently, in the mainstream human action recognition tasks, many algorithm models realize the classification of human actions by recognizing the action sequence information of the skeleton sequence [9], [10], [11]. By capturing the correlation information of skeleton sequences in spatio-temporal sequences, these algorithm models can distinguish the rules between different movements and realize the classification of human actions. In recent years, with the rapid development of deep learning in various fields, many researchers have been attracted to explore human action recognition tasks based on deep learning algorithms [3], [12], [13]. Deep learning models have achieved great success in human action recognition, mainly due to the massive data collected by researchers [14], [15]. However, in realistic scenarios, such as medical scenarios, doctors must spend a lot of time and energy to collect and annotate human actions. There are also ethical issues around obtaining consent for recording and privacy concerns. Therefore, massive training data is difficult to obtain [16], [17]. In the traditional deep learning paradigm, the model is prone to overfitting when using a small amount of training data and cannot be generalized effectively in the test phase. For scenarios where data is scarce, researchers have proposed a training method called Few-Shot learning [18], [19], where the model can obtain good results using only a small amount of training data. Few-shot learning has recently received widespread attention in industry and academia due to the above-mentioned difficulties in generating large, annotated datasets. In this context, our research aims to use few-shot learning on skeletal data to improve state-of-the-art few-shot action recognition. The Few-Shot human action recognition based on graph matching is usually achieved by designing a skeleton graph spatial structure measurement algorithm and combining it with mainstream temporal alignment algorithms (such as the Dynamic Time Warping(DTW) [20] algorithm) to achieve accurate skeleton sequence measurement. However, the DTW algorithm solves the alignment of action sequences under ideal conditions, searching the alignment path between the start and end positions of two series and accumulating the distance on the path as the alignment result. However, the algorithm does not consider the real world cases of misalignment at both ends, so it fails to get accurate temporal alignment results in some realistic scenarios. Therefore, this paper focuses on the human action recognition task with small samples and analyzes the defects of the DTW algorithm. To further improve the algorithm and optimize the time series alignment process, complementing the time series is proposed to obtain more accurate temporal alignment results. At the same time, the paper proposes adding the Virtual Adversarial Training method to improve the robustness of the DTW algorithm in noisy data. The contributions of this paper are summarized as follows:

- For the skeleton sequences, we propose a temporal alignment algorithm based on time series complement to calculate more accurate time series alignment results.
- During training, we added the Virtual Adversarial Training method to improve the algorithm's robustness to noise.
- Contrast and ablation experiments show the temporal complement and Virtual Adversarial Training show improvements to the DTW algorithm in real world datasets.

The rest of the paper is organized as follows: we first review the related literature in this area, then in the preliminaries section, we define the concepts used, next we describe our method, in the experiment section, we describe the different evaluations done on our method and the results. Finally, the conclusion section summarizes our work and contributions.

II. RELATED WORK

A. FEW-SHOT LEARNING

Few-Shot learning was proposed to address the difficulty of obtaining large numbers of annotated data for traditional machine learning methods [21]. In the Few-Shot learning approach, the model uses prior knowledge to quickly generalize to new tasks with only a small amount of supervised information. In recent years, with the rapid development and application of deep learning [22], [23], [24], more and more deep learning algorithms have been used to solve problems with small samples [18], [19], [25], [26], [27], [28]. Researchers primarily address the few-shot learning problem through two main approaches: one, using sparse samples based on data enhancement, and two, meta-learning mechanisms [29]. Data enhancement algorithms are mainly based on generative models (such as Generative Adversarial Nets (GANs) [30]) to enhance the original data and generate richer samples to improve the generalization ability of the model. Nowadays, mainstream Few-Shot learning adopts the method of meta-learning [19]. Meta-learning differs from traditional machine learning algorithms by making the model learn how to learn. Currently, there are three main meta-learning approaches: measure-based, optimization-based, and model-based learning. The metric-based framework constructs an end-to-end classifier by measuring the distance between samples. The classic algorithm is Prototypical Networks [18]: which determines the feature representations of diverse categories within the sample by deriving prototype representations. This method then classifies samples by measuring their proximity to these prototypes. Currently, measure-based Few-Shot learning stands as the most prevalent approach. Optimization-based meta-learning methods aim to find model initialization parameters that are effective for adapting to new tasks quickly. The classic algorithm is the model-independent meta-learning algorithm MAML [19]. Here, the model learns the initialization suitable for the

training data distribution in the training phase and can rapidly converge to reduce the occurrence of overfitting during the testing process. However, this algorithm is relatively less memory efficient than the metric-based method. The model-based method focuses on training models to adapt to new tasks with very limited data by developing a more versatile underlying model architecture. A classical class of model-based algorithms is Compound Memory Networks (CMN) [31]. It focuses on enhancing the model's ability to handle limited training data. CMN achieves this by incorporating memory modules that efficiently store and retrieve information. Multiple feature vectors are used as keys to strengthen the model's understanding of relevant features. These features are then organized and stored in 2D matrices within the memory modules. This arrangement facilitates the efficient retrieval of pertinent information, even when dealing with instances that occur infrequently over an extended period. CMN aims to improve the model's capacity to remember and utilize critical information, making it particularly useful for tasks where data is scarce, such as few-shot learning scenarios.

B. SKELETON-BASED HUMAN ACTION RECOGNITION

The process of recognizing human actions can be abstracted into a spatio-temporal backbone structure. In the spatial dimension, this backbone comprises nodes representing the body's bone points, connected by edges that encode their relationships. In the temporal dimension, each time point is represented as a skeleton graph, collectively forming a continuous stream of skeleton graph data. Yan et al. introduced Spatial Temporal Graph Convolutional Networks (ST-GCN) [9] for analyzing skeleton sequence data. ST-GCN leverages the Graph Convolutional Network (GCN) [32] for spatial structure and the Temporal Convolutional Network (TCN) [33] for temporal structure. This combination enables information transfer and fusion within the skeleton sequences across spatial and temporal dimensions, resulting in feature representations of human actions. These features are subsequently fed into a standard neural network classifier for human action recognition. Building upon this foundation, Shi et al. proposed Two-stream Adaptive Graph Convolutional Networks (2s-AGCN) [10]. Unlike traditional methods focusing solely on extracting skeleton sequence features, 2s-AGCN incorporates bone point and joint feature modeling, fusing these features before inputting them into the classifier to enhance performance. Another approach is the integration of GCN with LSTM networks (AGC-LSTM) [34]. This method effectively captures more discriminative features of human movement and employs attention mechanisms to enhance key node characteristics within the graph, thereby improving spatial-temporal representation. Furthermore, the Shift Graph Convolutional Network (Shift-GCN) [35] was introduced, which optimizes model parameters and computational complexity. This approach significantly reduces the number of model parameters and computation time. In the same year, researchers proposed the disentangling and unified graph convolution operator (MS-G3D) [11] to tackle human action recognition. This method recognizes that structurally separated joints also exhibit strong correlations. It seeks to extract multi-scale structural features and long-term dependencies beyond local associations of bone points. Although commonly used methods expand the receptive field of GCN to capture more information correlations, they often suffer from weight inconsistency when aggregating information from distant joints, with a bias towards nearby information. Human action recognition based on skeleton sequences can achieve more precise recognition by combining the power of GCN and TCN for feature extraction. This approach integrates human structural information and captures internal and external correlations, enhancing the overall recognition performance. Another approach to improve on GCN results is using Graph Attention networks (GAT) [36], [37], [38]. These methods have improved accounting for spatial and temporal long-range dependencies in skeleton sequences.

C. FEW-SHOT HUMAN ACTION RECOGNITION

Few-shot human action recognition tasks are usually based on video or skeleton-based approaches. Let's first consider the video-based framework, which extracts action features through an encoder. Then, the algorithm is designed based on related downstream tasks. Cao et al. proposed a video classification algorithm, Ordered Temporal Alignment Module (OTAM) [1] which is based on the prototype network, combined with the Temporal Segment Networks (TSN) [2] as a video processing method, and forms a new sequence as input by segmenting and randomly extracting video frames. They used a temporal alignment algorithm to measure the similarity of two videos by calculating the minimum cumulative alignment path cost between sequences. Some researchers believe that different videos significantly differ in length and speed, and an average video representation method could be more reasonable. To address this problem, Perrett et al. proposed Temporal-Relational CrossTransformers (TRX) [39]. This study treats video representation as comprising ordered tuples with varying numbers of frames, and action subsequences at different speeds and time offsets can be compared to solve action matching at different rates and time offsets. In real-world scenarios, there will be a lot of redundant data and noise in videos, and the skeleton-based human action recognition algorithm can solve this issue. Guo et al. proposed Neural Graph Matching (NGM) [40]. By constructing the relationship between the human body in the video and the interacting objects, the authors composed the structural information of the graph. Based on graph representation learning and metric learning, the model realizes human action recognition with a few samples. Reference [41] is based on the way of measurement learning. It measures the distance of different action types to get the classification results. The authors explored the initial posture characteristics and regularization mode of human skeletons and tested the consistency of the actions of autistic patients and doctors in real scenes. Reference [42]

realizes action recognition under the condition that only one labeled sample is given by narrowing the same type of sample features and moving away from different types of sample features. Reference [43] proposed Disentangled and Adaptive Spatial-Temporal Matching (DASTM). The authors found that existing spatio-temporal graph convolution algorithms have an over-smoothing problem when capturing relationships between skeletons, often resulting in nodes that are difficult to distinguish. Therefore, the authors propose a novel spatial matching strategy to mitigate this spatial degradation problem by adaptively disentangling and activating the representations of skeleton joints. To solve the problem of temporal mismatch, the authors use DTW to calculate the sequence distance, which is the final alignment result as the distance between the skeleton sequences. There are few existing skeleton-based few-shot learning algorithms, and there is no special design in temporal alignment to obtain accurate measurement results.

III. PRELIMINARIES

A. SKELETON SEQUENCE DEFINITION

The human skeleton graph sequence can be defined as $\mathcal{G} = \{G_1, G_2, \cdots, G_M\}$, where M represents the length of the sequence, and $G = \{X, A\}$ depicts the skeleton graph. $X \in \mathbb{R}^{n*3}$ in G represents the initial feature matrix of the nodes, $A \in \mathbb{R}^{n*n}$ represents the adjacency matrix of the skeleton graph, n is the number of nodes in the skeleton graph, and the initial feature dimension of each node is 3.

B. PROTOTYPE NETWORK BASED FEW-SHOT LEARNING

The dataset of few-shot learning is divided into training, validation, and test sets. During the few-shot training process, the model is trained on numerous small tasks. Each task involves the random selection of a limited number of samples. For instance, we randomly choose C samples from various types, and within each type, K samples are labeled. This selected data forms the support set, which contains labeled samples, while the remaining samples constitute the query set, which lacks labels and is used for prediction. This training methodology is known as episode training [19]. Typically, this extraction method is denoted as C-way-K-shot, with K usually taking values between 1 and 5. Next, we'll describe the Prototype Network [18]-based framework for few-shot learning. The Prototype Network belongs to the metric-based approach to few-shot learning. It constructs an end-to-end classifier and, during the episode training process, learns to discern differences between various sample types. This is achieved by measuring the distance between samples and modeling it as a distance distribution to perform classification. Now we define the prototypes and the distance function for the skeleton graph sequences. Given N action types, each type contains K samples, and the prototype of each type is represented as $C_k = \frac{1}{K}\sum_{(\mathcal{G}_i^s, y_i^s)} f_\phi(\mathcal{G}_i) \times \mathbb{I}(y_i^s = k)$. Where $\mathcal{G}_i^s$ represents the labeled skeletal point action sequence, k represents the type k action prototype, and $\mathbb{I}$ represents the Indicator function. That is, when the label y_i of the sample corresponds to the type k of the sample, it is 1, otherwise it is 0. $f_\phi(\cdot) : \mathbb{R}^{M\times n\times 3} \rightarrow \mathbb{R}^{m\times n\times d}$, represents the feature mapping method. In this study several encoders (ST-GCN/2s-AGCN/MS-G3D) are evaluated to encode the feature representation of skeleton sequences. ϕ indicates a learnable parameter. Based on this, the query set's action type is predicted by the closest prototype based on the distance function. The specific formula is as follows:

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

where $d(\cdot,\cdot)$ represents the distance function between two skeleton sequences, based on this measure, the sample of the query set belongs to the type of the closest prototype. The parameters are updated By calculating the negative log-likelihood function $\mathcal{J}(\phi) = -\log p_\phi(y = k \mid \mathcal{G}^q)$ as the loss function.

IV. METHOD

A. SPATIAL ACTIVATION

Measuring the distance between skeleton graphs traditionally involves calculating the Euclidean distance between nodes in these graphs. However, it's essential to consider that different nodes might have varying significance when representing different actions using skeleton graphs. To address this issue, this paper adopts the node activation algorithm introduced in DASTM [43]. This algorithm focuses on activating key nodes (joints) in matching pairs and emphasizes the calculation of distances between these key nodes. To provide a more detailed explanation of the Few-Shot learning framework, let's consider the node feature representations $H^q \in \mathbb{R}^{n\times d}$ for the query set and the graph representations $H^s \in \mathbb{R}^{n\times d}$ for the support set. These representations are generated by an identical spatial-temporal encoder such as ST-GCN, where n represents the number of nodes in the skeleton graph, and d represents the feature dimension of each node. These features are then subjected to linear transformations, followed by the calculation of the dot product to measure their correlation. This dot product calculation is employed to adjust the weights of nodes in the original skeleton graph feature representation. The formula for this operation is as follows:

$$\hat{H}^q = \text{SoftMax}\left(\frac{W_1^q H^q \cdot [W_2^q H^s]^T}{\sqrt{d}}\right) W_3^q H^q, \quad (2)$$

where W_1^q, W_2^q, and W_3^q represent the linear transformation matrix, and d represents the dimension of node representation. Similarly, the H^s representation of the support set skeleton graph can be transformed by the following formula:

$$\hat{H}^s = \text{SoftMax}\left(\frac{W_1^s H^s \cdot [W_2^s H^q]^T}{\sqrt{d}}\right) W_3^s H^s, \quad (3)$$

Therefore, we can define the distance representation of matching pairs of the query set and the support set skeleton graph as:

$$D(\hat{H}^q, \hat{H}^s) = ||\hat{H}^q - \hat{H}^s||_F, \quad (4)$$

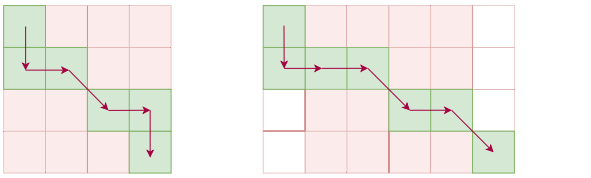

FIGURE 1. Comparison of temporal alignment path. (a) is the path of the original Dynamic Time Warping, and (b) is the path of Dynamic Time Warping with temporal complement.

where $||\cdot||_F$ stands for the Frobenius norm [44], which is the square sum of the absolute values of each element in the matrix.

B. TEMPORAL COMPLEMENT AND ALIGNMENT

This section proposes a temporal alignment algorithm to optimize the human skeleton sequence distance measurement. This method is based on the Dynamic Time Warping algorithm and completes both ends of the skeleton sequence, as shown in Figure 1:

Specifically, after softmax activation in the previous section, we get the feature representation of the query set sequence $Q = \left\{\hat{H}_1^q, \hat{H}_2^q, \cdots \hat{H}_m^q\right\}$ and the support set sequence $C_k = \left\{\hat{H}_1^s, \hat{H}_2^s, \cdots \hat{H}_m^s\right\}$. Then complete the query set Q to $Q = \left\{\hat{H}_0^q, \hat{H}_1^q, \hat{H}_2^q, \cdots \hat{H}_m^q, \hat{H}_{m+1}^q\right\}$, To clarify the process, let's break it down step by step. First, we set the distance of sequence Q to be zero for both the starting and ending positions, while keeping the support set unchanged. Next, we calculate the distance matrix $E' \in \mathbb{R}^{m\times(m+2)}$ corresponding to the time point of Q and C_k. Each element in this matrix, denoted as E', is determined by $D(\hat{H}^q, \hat{H}^s)$. Now, utilizing the Dynamic Time Warping algorithm, we align these two temporal sequences and compute their cumulative distance path. The formula for this operation is as follows:

$$\Gamma(i,j) = E'(i,j) + \begin{cases} \min\{\Gamma(i-1,j-1), \Gamma(i-1,j), \Gamma(i,j-1)\}, & j = 1 \text{ or } m+1 \\ \min\{\Gamma(i-1,j-1), \Gamma(i,j-1)\}, & \text{otherwise}, \end{cases} \tag{5}$$

where:

- $\Gamma(i,j)$ is the accumulated distance matrix at position (i,j).
- $E'(i,j)$ is the local cost or distance between corresponding elements of the two sequences at positions i and j.
- $\min\{\Gamma(i-1,j-1), \Gamma(i-1,j), \Gamma(i,j-1)\}$ is the minimum of the three neighboring cells (diagonal, above, and left) in the accumulated cost matrix. This represents the optimal alignment considering three possible moves: diagonal (match), above (insertion), and left (deletion).
- $\min\{\Gamma(i-1,j-1), \Gamma(i,j-1)\}$ is the minimum of the two neighboring cells (diagonal and left) when j is not at the beginning or end.

The recurrence relation calculates the accumulated cost matrix $\Gamma(i,j)$ based on the local cost $E'(i,j)$ and the minimum cost among the possible moves for each cell. Since the query set Q is expanded at the start and end positions, the entire temporal alignment process would be carried out at the longer end of the whole sequence. The overall model is trained based on backpropagation. Since *min* function is not differentiable, it cannot be used in the backpropagation optimization algorithm of deep learning models. Therefore researchers proposed Soft-DTW [45], which approximated the minimum function into the following differentiable form:

$$\min^{\gamma}\{a_1, \ldots, a_n\} := \begin{cases} \min_{i\le n} a_i, & \gamma = 0, \\ -\gamma \log \sum_{i=1}^{n} e^{-a_i/\gamma}, & \gamma > 0, \end{cases} \tag{6}$$

where $\{a_1, \ldots, a_n\}$ represents the minimized sequence content, and γ represents the smoothing parameter. When $\gamma = 0$, the soft minimum is equivalent to the regular minimum operation. However, when $\gamma > 0$, it introduces a differentiable approximation to the minimum. The second case involves taking the negative of γ multiplied by the logarithm of the sum of exponentials, providing a smooth, differentiable approximation to the minimum. The measures of the two sequences are expressed by the final position of the distance accumulation matrix $\Gamma(m, m+1)$. To ensure the symmetry of the calculation results, we expand support set C_k at the beginning and end positions in the same way, keeping the length of the query set Q sequence unchanged, then calculate the alignment result $\Gamma'(m+1, m)$ of the expanded sequences. Finally, average the two results as the distance measure between the query set Q and the support set Ck. The formula is as follows:

$$d(\mathcal{Q}, C_k) = mean(\Gamma(m, m+1), \Gamma'(m+1, m)), \tag{7}$$

Above equation provides a summary measure of the overall similarity or dissimilarity by considering the distance when support set is expanded in the beginning and end positions.

C. VIRTUAL ADVERSARIAL TRAINING (VAT) [46]

During the process of temporal alignment, the Dynamic Time Warping (DTW) algorithm heavily depends on the distance matrix of two sequences. Errors in this matrix directly influence the alignment path found through dynamic programming, leading to inaccurate measurements. This section discusses how we enhance DTW's ability to handle noise. In the training phase, we introduce adversarial perturbations to the time series distance matrix, resulting in different distance matrices. By exploring various cumulative distance paths during training, the model learns to adapt to different temporal alignment outcomes. In the testing phase, the model can cope with the influence of noise on the distance matrix, thus enhancing the model's overall ability to generalize and its robustness. For the distance matrix $E' \in \mathbb{R}^{m\times(m+2)}$ calculated in the previous section, we use

$\mathcal{D}(E', \theta)$ to represent the temporal alignment search path and the alignment result obtained by accumulating the temporal alignment path distance, where θ represents the parameters of the model. Then, a Gaussian distribution is used to generate random perturbations $d \in \mathbb{R}^{m\times(m+2)}$, which acts on the distance matrix E'. The alignment result $\mathcal{D}(E' + d, \theta)$ is calculated similarly. To approximate the measurement of time series distance based on these different time series distance matrices, we employ the KL divergence [47]. The choice of KL divergence over other distance metrics (ex: Wasserstein distance) is due to KL divergence not being sensitive to the exact location of the perturbations [48]. This property is important to prevent large fluctuations of the loss during training. The specific formula is as follows:

$$\mathcal{L}_{adv} = KL(\mathcal{D}(E' + d, \theta)||\mathcal{D}(E', \theta)), \tag{8}$$

Using the approximate result $\mathcal{L}_{adv}$ of KL divergence as the loss function, the gradient $\nabla\mathcal{L}_{adv}$ of the model is obtained through backpropagation. Treating it as a perturbation that acts on the distance matrix E' of the sequences, the model is most likely to make mistakes in this direction. Similar to the previous process, applying the generated virtual adversarial perturbations $\nabla\mathcal{L}_{adv}$ to the distance matrix E' and using the KL divergence to approximately calculate the output results $\mathcal{D}(E', \theta)$ and $\mathcal{D}(E'+d, \theta)$ before and after adding the random perturbations $\nabla\mathcal{L}_{adv}$, $\mathcal{L}_{reg}$ is obtained as the regular term of the overall loss function of the model. The specific formula is as follows:

$$\mathcal{L}_{reg} = KL(\mathcal{D}(E' + \nabla\mathcal{L}_{adv}, \theta)||\mathcal{D}(E', \theta)), \tag{9}$$

D. MODEL FRAMEWORK AND LOSS FUNCTION

The overall framework of the model is shown in Figure 2. In many cases, when multiple layers of GCN are applied consecutively, the discriminability of node representations diminishes—-commonly referred to as over-smoothing [43] of graph node feature representation. In order to obtain more accurate skeleton graph measurement results, this paper adopts the algorithm proposed by DASTM [43] to maximize the rank of the skeleton feature matrix, ensuring a diverse set of skeleton point feature representations. Given the skeleton graph feature matrix $H_{bi} \in \mathbb{R}^{n\times d}$ encoded by the encoder (such as ST-GCN), where b represents the length of the skeleton sequence, i represents the skeleton graph in the sequence. DASTM maximizes the rank of the skeleton graph feature matrix to disentangle the inter-node feature representation in the graph. Specifically, calculating the nuclear-norm [49] represents $\|H_{bi}\|_*$ [50], [51] of the matrix as an approximation of the rank of the matrix H_{bi} [51], as follows:

$$\|H_{bi}\|_* = \sum_{j}^{\min(n,d)} \left(\sigma_i^j\right) < \text{rank}\left(H_{bi}\right), \tag{10}$$

where σ_i represents the i-th singular value of the matrix H_{bi}, which can be calculated by Singular Value Decomposition (SVD) [52]. After constraining the feature representation of the skeleton graph in the training phase, the spatial disentanglement Formula 10 is taken as a part of the loss function to optimize the model learning:

$$\mathcal{L}_{\text{dis}} = -\frac{1}{B * m}\sum_{b}^{B}\sum_{i}^{m}\|H_{bi}\|_*, \tag{11}$$

where B represents the number of samples in a batch during training, and m represents the length of the skeleton sequence. Finally, the model's loss function comprises the measurement results of the skeleton sequences, the regular term $\mathcal{L}_{\text{dis}}$ and $\mathcal{L}_{reg}$. The loss function is as follows:

$$\mathcal{L}_{\text{match}} = -\frac{1}{N^q}\sum_{i}^{N^q}\log p_\phi\left(\hat{y}_i = y_i \mid \mathcal{G}_i^q\right) + \lambda\mathcal{L}_{\text{dis}} + \alpha\mathcal{L}_{\text{reg}}, \tag{12}$$

where N^q represents the number of query samples, $\hat{y}_i$, and y_i represent the predicted label and true label for the action sample $\mathcal{G}_i^q$, λ represents the weight coefficient of the regular term $\mathcal{L}_{dis}$, α represents the weight coefficient of the $\mathcal{L}_{reg}$. Hyperparameter analysis is required to determine the value of these coefficients.

V. EXPERIMENT

A. DATASETS INTRODUCTION

In this section, we will provide a brief overview of the three datasets utilized in our experiments. The first dataset source, NTU RGB+D 120 [14], was curated by Nanyang Technological University for the purpose of human action recognition. It comprises an extensive collection of 113,945 skeleton sequences, encompassing 120 distinct action types. These actions consist of 82 daily activities, 12 actions related to human health, and 26 interactive actions involving two individuals. This study randomly sampled data from all 120 action types. Two datasets, namely NTU-S and NTU-T, were formed by randomly selecting 60 and 30 human action samples for each action type, respectively. Within these datasets, the human body structure is represented by 25 nodes, each containing spatial 3D position information. This spatial data is used as the initial feature for our analysis. For each dataset, a random selection process allocated 80 action types for training, 20 for validation, and 20 for testing. The third dataset, Kinetics [15], is a collection of 260,232 human action videos compiled by DeepMind researchers. It encompasses 400 different action categories, each containing 400 human action videos. The original dataset primarily comprises video information, which differs from our approach focused on skeleton-based action recognition. To adapt it to our research, we employed OpenPose [53] to extract human skeleton sequences from the video data. From each human body structure, we extracted 18 skeleton nodes to construct a graph structure. Each node was characterized by three initial features: the horizontal and vertical distances from the point to the upper left corner

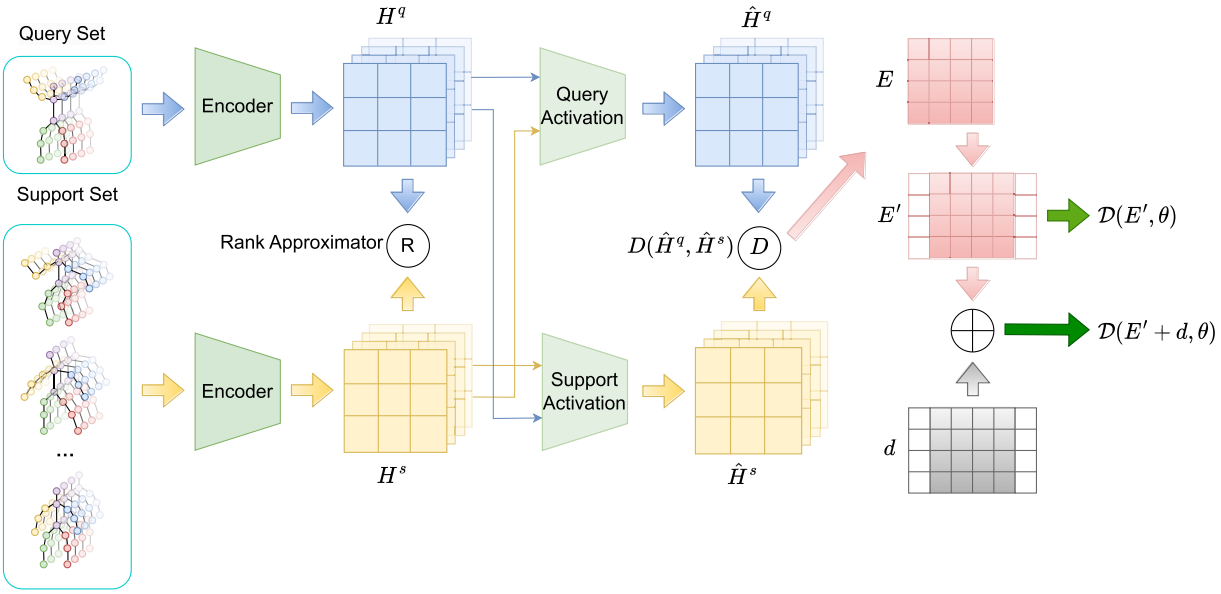

FIGURE 2. The Framework for Few-Shot Human Action Recognition Based on spatial-temporal alignment: The Query set is the set of examples used to evaluate the model performance. The support set is the (small number) of labeled examples which supports the learning process. Both of these skeleton sets are passed through the encoders (ST-GCN/2S-AGCN/MS-G3D) to obtain their feature representations. The rank approximator assists in learning a low-rank representation of the support set, enabling the model to generalize and make accurate predictions on the query set. The query and support activations output the representations of the query and support sets respectively. D represents the distance calculation using Dynamic time warping. Which is further elaborated in the methodology section of this paper.

of the image and the confidence score of identifying the bone node. Our approach for this dataset was similar to that used for NTU RGB+D 120. We randomly selected 120 different human action types and chose 100 samples for each action type. The training, validation, and test sets for Kinetics were divided similarly to NTU-S and NTU-T.

B. BASELINES

The baseline algorithms used in this paper include:

1) PROTONET [18]

ProtoNet is based on the prototype network framework, directly using Euclidean-distance measurement to classify human actions after feature extraction by a spatial-temporal encoder;

2) DTW [20]

DTW is based on the prototype network framework, takes Dynamic Time Warping as the temporal measurement algorithm;

3) DASTM [43]

The model structure of DASTM is the same as that of DTW. By Learning disentangled skeleton representation and activating the key nodes in the skeleton graph, this model can obtain a better graph representation space.

C. IMPLEMENTATION DETAILS

The model undergoes training using an episodic approach [19]. Within the training phase, we begin by randomly selecting N action types. We assemble a support set consisting of K labeled samples for each of these types. Simultaneously, we exclude the support set samples related to these N action types and randomly gather the remaining data to form a query set. Each skeleton sequence in the dataset undergoes a video preprocessing procedure known as TSN [2]. For both the NTU-S and NTU-T datasets, we randomly select 30 frames from each sequence, while in the case of the Kinetics dataset, 50 frames are chosen as the initial input for the model. This uniform sampling strategy ensures that both the support and query sets have the same sequence length for input to the model.

The model's feature encoder is based on established architectures like ST-GCN, 2s-AGCN, and MS-G3D [9], [10], [11], which serve as the foundation for our experiments. Training is performed on an NVIDIA GeForce RTX 3090 GPU using the PyTorch deep learning framework. We employ the Adam optimization algorithm [54], with an initial learning rate of 0.001 and a learning rate decay mechanism. Specifically, if, after ten consecutive epochs, the validation set's accuracy does not improve, we halve the learning rate and continue training. During the training phase, each epoch consists of 500 subtasks, which are further categorized as 5-way-1-shot and 5-way-5-shot. In other words, we randomly select five action types for each subtask, with either 1 or 5 labeled samples per type. The model is trained end-to-end, with human skeleton sequences as input. The model's loss function is calculated according to Equation 12. To ensure robustness, we perform each experiment with three different random seeds and calculate the average accuracy of action recognition and the specific variance.

TABLE 1. Accuracies of Few-Shot action recognition based on 5-way-1-shot.

Backbone	Datasets / Methods	NTU-T	NTU-S	Kinetics
ST-GCN	ProtoNet	$71.2_{\pm 1.5}$	$73.3_{\pm 0.3}$	$37.4_{\pm 0.4}$
	DTW	$72.2_{\pm 2.1}$	$73.5_{\pm 0.4}$	$38.4_{\pm 0.2}$
	DASTM	$\mathbf{75.1}_{\pm 1.8}$	$76.2_{\pm 0.3}$	$39.3_{\pm 0.1}$
	Ours	$73.1_{\pm 1.7}$	$\mathbf{76.3}_{\pm 0.3}$	$\mathbf{39.4}_{\pm 0.5}$
2S-AGCN	ProtoNet	$68.1_{\pm 0.5}$	$72.8_{\pm 0.3}$	$38.4_{\pm 0.2}$
	DTW	$70.8_{\pm 1.4}$	$71.5_{\pm 1.2}$	$\mathbf{40.9}_{\pm 0.4}$
	DASTM	$\mathbf{73.3}_{\pm 0.6}$	$74.0_{\pm 0.7}$	$40.8_{\pm 0.3}$
	Ours	$71.2_{\pm 1.8}$	$\mathbf{74.4}_{\pm 0.3}$	$40.8_{\pm 0.1}$
MS-G3D	ProtoNet	$70.1_{\pm 1.0}$	$73.6_{\pm 0.2}$	$39.5_{\pm 0.3}$
	DTW	$72.4_{\pm 0.2}$	$73.9_{\pm 0.4}$	$40.6_{\pm 0.2}$
	DASTM	$\mathbf{75.0}_{\pm 0.9}$	$76.3_{\pm 1.2}$	$41.1_{\pm 0.2}$
	Ours	$73.1_{\pm 1.8}$	$\mathbf{77.7}_{\pm 0.4}$	$\mathbf{41.2}_{\pm 0.5}$

TABLE 2. Accuracies of Few-Shot action recognition based on 5-way-5-shot.

Backbone	Datasets / Methods	NTU-T	NTU-S	Kinetics
ST-GCN	ProtoNet	$81.8_{\pm 0.2}$	$84.3_{\pm 0.3}$	$46.8_{\pm 0.4}$
	DTW	$81.0_{\pm 0.6}$	$81.5_{\pm 0.5}$	$47.9_{\pm 0.3}$
	DASTM	$83.0_{\pm 0.1}$	$85.5_{\pm 0.3}$	$48.9_{\pm 0.1}$
	Ours	$\mathbf{83.3}_{\pm 0.4}$	$\mathbf{85.9}_{\pm 0.3}$	$\mathbf{49.5}_{\pm 0.2}$
2S-AGCN	ProtoNet	$81.9_{\pm 0.1}$	$84.2_{\pm 0.1}$	$50.5_{\pm 0.2}$
	DTW	$81.2_{\pm 0.9}$	$82.5_{\pm 0.8}$	$50.8_{\pm 0.3}$
	DASTM	$83.8_{\pm 0.8}$	$\mathbf{86.8}_{\pm 0.3}$	$50.9_{\pm 0.9}$
	Ours	$\mathbf{83.9}_{\pm 1.2}$	$85.2_{\pm 0.3}$	$\mathbf{51.0}_{\pm 0.3}$
MS-G3D	ProtoNet	$82.3_{\pm 0.2}$	$85.3_{\pm 0.1}$	$50.0_{\pm 0.3}$
	DTW	$81.3_{\pm 0.3}$	$83.2_{\pm 0.4}$	$50.0_{\pm 0.2}$
	DASTM	$\mathbf{84.3}_{\pm 0.3}$	$\mathbf{87.3}_{\pm 1.2}$	$51.1_{\pm 0.9}$
	Ours	$83.3_{\pm 1.5}$	$85.8_{\pm 1.4}$	$\mathbf{51.8}_{\pm 0.3}$

D. ANALYSIS

Table 1 shows the results of the 5-way-1-shot experiment, and Table 2 shows the results of the 5-way-5-shot experiment. The algorithm introduced in this paper demonstrates a human action recognition accuracy exceeding 70% on the relatively clean NTU-T and NTU-S datasets, underscoring its effectiveness. Notably, in the 5-category NTU-S experiment, each test process includes only one labeled sample, yet the average human action recognition accuracy across various backbone networks is an impressive 76.1%, surpassing existing methods. However, when considering the results on the Kinetics dataset, the improvement in recognition accuracy offered by DASTM over DTW is less pronounced. This can be attributed to the substantial noise present in Kinetics, including images from different sources, varying video quality, and imprecise human body movement and posture estimations within the videos. However, our model incorporating Virtual Adversarial Training to enhance the fault tolerance of the Dynamic Time Warping algorithm in the temporal alignment process results in performance improvements. In summary, the few-shot human action recognition algorithm proposed in this paper consistently delivers improved results across diverse experimental settings and datasets.

TABLE 3. Ablation study of Few-Shot action recognition based on 5-way-1-shot.

Backbone	Datasets / Methods	NTU-T	NTU-S	Kinetics
ST-GCN	DTW	$72.2_{\pm 2.1}$	$73.5_{\pm 0.4}$	$38.4_{\pm 0.2}$
	DTW+TC	$72.0_{\pm 0.5}$	$71.7_{\pm 0.6}$	$39.1_{\pm 0.3}$
	DTW+VAT	$72.0_{\pm 0.9}$	$\mathbf{74.4}_{\pm 0.4}$	$38.5_{\pm 0.6}$
	DTW+TC+VAT	$\mathbf{73.0}_{\pm 0.8}$	$71.8_{\pm 0.5}$	$\mathbf{39.1}_{\pm 0.3}$
2S-AGCN	DTW	$70.8_{\pm 1.4}$	$71.5_{\pm 1.2}$	$40.9_{\pm 0.4}$
	DTW+TC	$\mathbf{70.9}_{\pm 0.9}$	$72.8_{\pm 0.5}$	$40.9_{\pm 0.4}$
	DTW+VAT	$70.3_{\pm 1.2}$	$\mathbf{73.6}_{\pm 0.3}$	$40.8_{\pm 0.1}$
	DTW+TC+VAT	$70.1_{\pm 1.8}$	$71.1_{\pm 0.9}$	$\mathbf{41.2}_{\pm 0.3}$
MS-G3D	DTW	$72.4_{\pm 0.2}$	$73.9_{\pm 0.4}$	$40.6_{\pm 0.2}$
	DTW+TC	$70.6_{\pm 0.6}$	$74.9_{\pm 1.6}$	$40.8_{\pm 0.2}$
	DTW+VAT	$69.1_{\pm 0.3}$	$\mathbf{75.0}_{\pm 1.7}$	$41.1_{\pm 0.1}$
	DTW+TC+VAT	$\mathbf{72.9}_{\pm 0.3}$	$74.9_{\pm 0.4}$	$\mathbf{41.5}_{\pm 0.3}$

TABLE 4. Ablation study of Few-Shot action recognition based on 5-way-5-shot.

Backbone	Datasets / Methods	NTU-T	NTU-S	Kinetics
ST-GCN	DTW	$81.0_{\pm 0.6}$	$81.5_{\pm 0.5}$	$47.9_{\pm 0.3}$
	DTW+TC	$\mathbf{82.2}_{\pm 0.3}$	$81.4_{\pm 0.5}$	$48.9_{\pm 0.2}$
	DTW+VAT	$80.8_{\pm 0.8}$	$80.8_{\pm 0.7}$	$48.5_{\pm 0.3}$
	DTW+TC+VAT	$79.6_{\pm 1.7}$	$\mathbf{81.9}_{\pm 0.3}$	$\mathbf{48.9}_{\pm 0.1}$
2S-AGCN	DTW	$81.2_{\pm 0.9}$	$82.5_{\pm 0.8}$	$50.8_{\pm 0.3}$
	DTW+TC	$\mathbf{82.6}_{\pm 0.3}$	$80.8_{\pm 0.6}$	$50.2_{\pm 0.1}$
	DTW+VAT	$80.8_{\pm 0.7}$	$\mathbf{82.7}_{\pm 0.3}$	$\mathbf{51.2}_{\pm 0.6}$
	DTW+TC+VAT	$80.3_{\pm 1.3}$	$81.7_{\pm 1.1}$	$50.7_{\pm 0.4}$
MS-G3D	DTW	$81.3_{\pm 0.3}$	$83.2_{\pm 0.4}$	$50.0_{\pm 0.2}$
	DTW+TC	$82.5_{\pm 0.3}$	$\mathbf{84.8}_{\pm 0.5}$	$50.9_{\pm 0.4}$
	DTW+VAT	$82.5_{\pm 0.5}$	$84.0_{\pm 0.9}$	$\mathbf{51.1}_{\pm 0.5}$
	DTW+TC+VAT	$\mathbf{82.6}_{\pm 0.5}$	$83.4_{\pm 0.4}$	$51.0_{\pm 0.3}$

E. ABLATION STUDY

We conducted ablation studies focused on DTW to provide a more thorough assessment of the effectiveness of our algorithm presented in this article. Our approach involved combining different components with the DTW algorithm, resulting in the following configurations:

1) DTW: DTW distance applied to the prototype framework as described above.
2) DTW+TC: This variation includes the temporal completion algorithm based on DTW (DTW+TC).
3) DTW+VAT: Here, we integrated Virtual Adversarial Training (VAT) with the DTW algorithm (DTW+VAT).
4) DTW+TC+VAT: This version combines both the temporal completion algorithm and Virtual Adversarial Training with the DTW algorithm (DTW+TC+VAT).

The results of these ablation studies are presented in table 3 and table 4 for reference. To more reasonably evaluate the contribution of each item in the model to the experimental results, we also conducted ablation experiments on the Rank Approximator(RA) and the regularization term(reg) proposed in this article. The specific results are shown in table 5 and table 6, demonstrating that aligning the Rank Approximator with the temporal alignment algorithm can achieve good results.

TABLE 5. Ablation study on Rank Approximator based on 5-way-1-shot.

Backbone	Datasets / Methods	NTU-T	NTU-S	Kinetics
ST-GCN	DTW	$72.2_{\pm 2.1}$	$73.5_{\pm 0.4}$	$38.4_{\pm 0.2}$
	DTW+RA	$\mathbf{75.1}_{\pm 1.8}$	$76.2_{\pm 0.3}$	$39.3_{\pm 0.1}$
	DTW+reg	$73.0_{\pm 0.8}$	$71.8_{\pm 0.5}$	$39.1_{\pm 0.3}$
	DTW+RA+reg	$73.1_{\pm 1.7}$	$\mathbf{76.3}_{\pm 0.3}$	$\mathbf{39.4}_{\pm 0.5}$
2S-AGCN	DTW	$70.8_{\pm 1.4}$	$71.5_{\pm 1.2}$	$40.9_{\pm 0.4}$
	DTW+RA	$\mathbf{73.3}_{\pm 0.6}$	$74.0_{\pm 0.7}$	$40.8_{\pm 0.3}$
	DTW+reg	$70.1_{\pm 1.8}$	$71.1_{\pm 0.9}$	$\mathbf{41.2}_{\pm 0.3}$
	DTW+RA+reg	$71.2_{\pm 1.8}$	$\mathbf{74.4}_{\pm 0.3}$	$40.8_{\pm 0.1}$
MS-G3D	DTW	$72.4_{\pm 0.2}$	$73.9_{\pm 0.4}$	$40.6_{\pm 0.2}$
	DTW+RA	$\mathbf{75.0}_{\pm 0.9}$	$76.3_{\pm 1.2}$	$41.1_{\pm 0.2}$
	DTW+reg	$72.9_{\pm 0.3}$	$74.9_{\pm 0.4}$	$\mathbf{41.5}_{\pm 0.3}$
	DTW+RA+reg	$73.1_{\pm 1.8}$	$\mathbf{77.7}_{\pm 0.4}$	$41.2_{\pm 0.5}$

TABLE 6. Ablation study on the Rank Approximator based on 5-way-5-shot.

Backbone	Datasets / Methods	NTU-T	NTU-S	Kinetics
ST-GCN	DTW	$81.0_{\pm 0.6}$	$81.5_{\pm 0.5}$	$47.9_{\pm 0.3}$
	DTW+RA	$83.0_{\pm 0.1}$	$85.5_{\pm 0.3}$	$48.9_{\pm 0.1}$
	DTW+reg	$79.6_{\pm 1.7}$	$81.9_{\pm 0.3}$	$48.9_{\pm 0.1}$
	DTW+RA+reg	$\mathbf{83.3}_{\pm 0.4}$	$\mathbf{85.9}_{\pm 0.3}$	$\mathbf{49.5}_{\pm 0.2}$
2S-AGCN	DTW	$81.2_{\pm 0.9}$	$82.5_{\pm 0.8}$	$50.8_{\pm 0.3}$
	DTW+RA	$83.8_{\pm 0.8}$	$\mathbf{86.8}_{\pm 0.3}$	$50.9_{\pm 0.9}$
	DTW+reg	$80.3_{\pm 1.3}$	$81.7_{\pm 1.1}$	$50.7_{\pm 0.4}$
	DTW+RA+reg	$\mathbf{83.9}_{\pm 1.2}$	$85.2_{\pm 0.3}$	$\mathbf{51.0}_{\pm 0.3}$
MS-G3D	DTW	$81.3_{\pm 0.3}$	$83.2_{\pm 0.4}$	$50.0_{\pm 0.2}$
	DTW+RA	$\mathbf{84.3}_{\pm 0.3}$	$\mathbf{87.3}_{\pm 1.2}$	$51.1_{\pm 0.9}$
	DTW+reg	$82.6_{\pm 0.5}$	$83.4_{\pm 0.4}$	$51.0_{\pm 0.3}$
	DTW+RA+reg	$83.3_{\pm 1.5}$	$85.8_{\pm 1.4}$	$\mathbf{51.8}_{\pm 0.3}$

1) ANALYSIS OF TEMPORAL COMPLEMENT

Temporal completion, when compared to DTW, demonstrates enhanced accuracy on the NTU-S and NTU-T datasets. This improvement primarily stems from optimizing the alignment process at the beginning and end positions, building upon the original Dynamic Time Warping algorithm. As illustrated in figure 1, the supplementation of the starting and ending positions in the sequences allows the alignment path of the two sequences to disregard the alignment results at these positions in the original sequence. This results in increased accuracy in temporal alignment.

2) ANALYSIS OF VAT

Given the substantial noise present in the Kinetics dataset, Virtual Adversarial Training (VAT) effectively enhances the model's generalization capability. VAT addresses the issue of the DTW algorithm's weak anti-noise capacity during temporal alignment. This section compares the temporal alignment path results of DTW with and without the addition of virtual adversarial perturbations. In the test phase, we visualize the distance accumulation matrix Γ generated by the DTW algorithm and compare the results, as shown in figure 3. Green arrows indicate the temporal alignment results. As depicted in the figure, the temporal alignment path of the original DTW algorithm largely remains along the diagonal between sequences, with minimal changes. Any noise encountered leads to the incorporation of the noisy results into the sequence alignment. The inclusion of virtual adversarial perturbations enriches the temporal alignment paths. During training, the model becomes adaptable to variations in the sequence distance matrix, thereby offering flexibility in resolving noise-related issues. Consequently, adding virtual adversarial perturbations to the distance matrix enhances alignment results' flexibility and boosts the DTW algorithm's robustness.

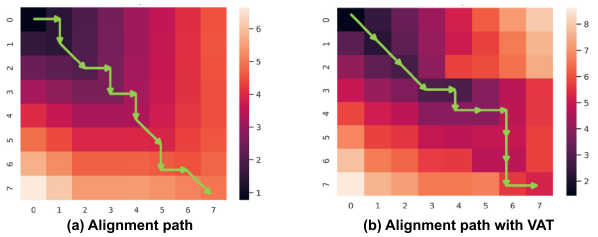

FIGURE 3. Comparison of temporal alignment paths. (a) and (b) are the alignment paths before and after adding virtual adversarial perturbations.

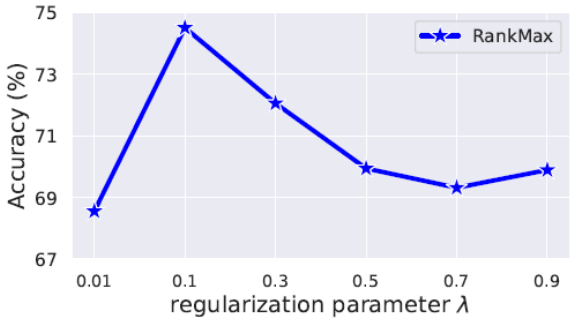

FIGURE 4. Hyperparameter analysis of Matrix rank maximization regularization term.

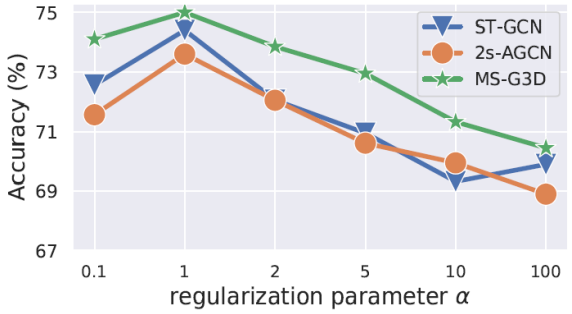

FIGURE 5. Hyperparameter analysis of virtual adversarial perturbation loss regularization term.

F. HYPERPARAMETER STUDY

This section analyzes the impact of hyperparameters in the model on experimental results. Figure 4 shows experimental results with various regularization term weight parameters λ from equation 12. Among them, the model performance for $\lambda = 0.01$ is low, mainly due to the inconspicuous feature disentanglement. When λ is close to 1, excessive force to suppress the relationship represented by the skeleton graph nodes will also lose the original skeleton graph features, resulting in unsatisfactory results. Therefore, we use a smaller weight parameter λ=0.1 to retain the skeleton graph node representation.

Figure 5 shows different experimental results based on the NTU-S after feature extraction on different backbones and regularization term weights α settings. Among them, the model performance for α=0.1 is low. The main reason is that the influence of the regular term virtual adversarial perturbation is small and cannot effectively provide the model with strong fault tolerance. When the value of α is greater than 1, the tolerance to noise during the temporal alignment process is excessively forced, causing some action samples with less noise to be amplified by the regular term, which amplifies the problem. Therefore, when the weight coefficient α is around 1, we get the best performance.

VI. CONCLUSION

This paper analyzes the issues and limitations of the Dynamic Time Warping (DTW) algorithm, offering improvements to its temporal alignment results, ultimately increasing the accuracy of Few-Shot human action recognition. The study identifies two prominent shortcomings in the DTW algorithm:

- Obligatory Alignment of Sequence Boundaries: DTW mandates the alignment of sequence start and end positions.
- Vulnerability to Noise: The algorithm's performance significantly falters when noise is introduced into the sequence.

The direct application of the DTW algorithm for measuring sequence distances yielded suboptimal experimental results in our experiments. Our results improved after addressing the above two concerns by:

- Temporal Completion: Solving the necessity for aligning start and end positions.
- Virtual Adversarial Training (VAT): Enhancing the model's noise resistance and overall robustness.

The comparative analysis of experimental results across different datasets and backbone networks underscores the efficacy of the proposed algorithm in improving the accuracy of Few-Shot human action recognition tasks.

REFERENCES

[1] K. Cao, J. Ji, Z. Cao, C.-Y. Chang, and J. C. Niebles, "Few-shot video classification via temporal alignment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10618–10627.

[2] K. Liu, F. Liu, H. Wang, N. Ma, J. Bu, and B. Han, "Partition speeds up learning implicit neural representations based on exponential-increase hypothesis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 5474–5483.

[3] H. Wang, P. Zhang, X. Zhu, I. W. Tsang, L. Chen, C. Zhang, and X. Wu, "Incremental subgraph feature selection for graph classification," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 1, pp. 128–142, Jan. 2017.

[4] H.-B. Zhang, Y.-X. Zhang, B. Zhong, Q. Lei, L. Yang, J.-X. Du, and D.-S. Chen, "A comprehensive survey of vision-based human action recognition methods," *Sensors*, vol. 19, no. 5, p. 1005, Feb. 2019.

[5] G. Johansson, "Visual perception of biological motion and a model for its analysis," *Perception Psychophys.*, vol. 14, no. 2, pp. 201–211, 1973.

[6] K. Wagner, "Kinect 2 full video walkthrough: The Xbox sees you like never before," 2013, vol. 8.

[7] A. Zabatani, V. Surazhsky, E. Sperling, S. B. Moshe, O. Menashe, D. H. Silver, Z. Karni, A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Intel® RealSense™ SR300 coded light depth camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2333–2345, Oct. 2020.

[8] F. Han, B. Reily, W. Hoff, and H. Zhang, "Space-time representation of people based on 3D skeletal data: A review," *Comput. Vis. Image Understand.*, vol. 158, pp. 85–105, May 2017.

[9] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, Apr. 2018.

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

[11] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang, "Disentangling and unifying graph convolutions for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 143–152.

[12] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 6202–6211.

[13] H. Wang, Z. Cui, Y. Chen, M. Avidan, A. B. Abdallah, and A. Kronzer, "Predicting hospital readmission via cost-sensitive deep learning," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 15, no. 6, pp. 1968–1978, Nov. 2018.

[14] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. C. Kot, "NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2684–2701, Oct. 2020.

[15] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017, *arXiv:1705.06950*.

[16] X. Cao, X. Li, L. Ma, Y. Huang, X. Feng, Z. Chen, H. Zeng, and J. Cao, "AggPose: Deep aggregation vision transformer for infant pose estimation," 2022, *arXiv:2205.05277*.

[17] M.-R. Tseng, A. Gupta, C.-K. Tang, and Y.-W. Tai, "HAA4D: Few-shot human atomic action recognition via 3D spatio-temporal skeletal alignment," 2022, *arXiv:2202.07308*.

[18] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, Dec. 2017, pp. 4077–4087.

[19] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1126–1135.

[20] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-26, no. 1, pp. 43–49, Feb. 1978.

[21] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, Dec. 2012, pp. 1097–1105.

[23] H. Wang and P. Avillach, "Diagnostic classification and prognostic prediction using common genetic variants in autism spectrum disorder: Genotype-based deep learning," *JMIR Med. Informat.*, vol. 9, no. 4, Apr. 2021, Art. no. e24754.

[24] H. Wang, Q. Zhang, J. Wu, S. Pan, and Y. Chen, "Time series feature learning with labeled and unlabeled data," *Pattern Recognit.*, vol. 89, pp. 55–66, May 2019.

[25] C. Zhou, H. Wang, S. Zhou, Z. Yu, D. Bandara, and J. Bu, "Hierarchical knowledge propagation and distillation for few-shot learning," *Neural Netw.*, vol. 167, pp. 615–625, Oct. 2023.

[26] H. Wang, L. Chi, and Z. Zhao, "ASDPred: An end-to-end autism screening framework using few-shot learning," in *Proc. 31st ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2022, pp. 5004–5008.

[27] H. Wang, J. Wu, P. Zhang, and Y. Chen, "Learning shapelet patterns from network-based time series," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 3864–3876, Jul. 2019.

[28] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1199–1208.

[29] S. Bengio, Y. Bengio, J. Cloutier, and J. Gecsei, "On the optimization of a synaptic learning rule," in *Optimality in Artificial and Biological Neural Networks*, vol. 2. U.K.: Routledge, Jun. 2013.

[30] A. Mehrotra and A. Dukkipati, "Generative adversarial residual pairwise networks for one shot learning," 2017, *arXiv:1703.08033*.
[31] L. Zhu and Y. Yang, "Compound memory networks for few-shot video classification," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 751–766.
[32] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.
[33] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.
[34] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional LSTM network for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1227–1236.
[35] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, "Skeleton-based action recognition with shift graph convolutional network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 183–192.
[36] H. Wang, G. Tao, J. Ma, S. Jia, L. Chi, H. Yang, Z. Zhao, and J. Tao, "Predicting the epidemics trend of COVID-19 using epidemiological-based generative adversarial networks," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 2, pp. 276–288, Feb. 2022.
[37] Z. Zhao, Y. Zheng, Z. Zhang, H. Wang, Y. Zhao, and C. Li, "Exploring spatio-temporal representations by integrating attention-based bidirectional-LSTM-RNNs and FCNs for speech emotion recognition," in *Proc. Interspeech*, 2018, pp. 272–276, doi: 10.21437/Interspeech.2018-1477.
[38] Z. Li, H. Wang, P. Zhang, P. Hui, J. Huang, J. Liao, J. Zhang, and J. Bu, "Live-streaming fraud detection: A heterogeneous graph neural network approach," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2021, pp. 3670–3678.
[39] T. Perrett, A. Masullo, T. Burghardt, M. Mirmehdi, and D. Damen, "Temporal-relational crosstransformers for few-shot action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 475–484.
[40] M. Guo, E. Chou, D.-A. Huang, S. Song, S. Yeung, and L. Fei-Fei, "Neural graph matching networks for fewshot 3D action recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 653–669.
[41] A. Sabater, L. Santos, J. Santos-Victor, A. Bernardino, L. Montesano, and A. C. Murillo, "One-shot action recognition in challenging therapy scenarios," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 2777–2785.
[42] R. Memmesheimer, S. Haring, N. Theisen, and D. Paulus, "Skeleton-DML: Deep metric learning for skeleton-based one-shot action recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 3702–3710.
[43] N. Ma, H. Zhang, X. Li, S. Zhou, Z. Zhang, J. Wen, H. Li, J. Gu, and J. Bu, "Learning spatial-preserved skeleton representations for few-shot action recognition," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 174–191.
[44] D. I. Merino, "Topics in matrix analysis," Ph.D. dissertation, Johns Hopkins Univ., Baltimore, MD, USA, 1992.
[45] M. Cuturi and M. Blondel, "Soft-DTW: A differentiable loss function for time-series," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 894–903.
[46] T. Miyato, S.-I. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1979–1993, Aug. 2018.
[47] H. C. Carver, A. O'Toole, and T. Raiford, *The Annals of Mathematical Statistics*. Ann Arbor, MI, USA: Edwards, 1930.
[48] W. K. H. Wu, A. C. S. Chung, and H. H. N. Lam, "Multi-resolution LC-MS images alignment using dynamic time warping and Kullback–Leibler distance," in *Proc. 19th IEEE Int. Conf. Image Process.*, Sep. 2012, pp. 1681–1684.
[49] G. Belitskii, *Matrix Norms and Their Applications*, vol. 36. Cambridge, MA, USA: Birkhaäser, 2013.
[50] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.
[51] S. Cui, S. Wang, J. Zhuo, L. Li, Q. Huang, and Q. Tian, "Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 3941–3950.
[52] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
[53] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7291–7299.
[54] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

QINGYANG XU received the B.S. degree in mathematics and applied mathematics from Nankai University, Tianjin, China. Currently, she is a Research Assistant with the College of Computer Science, Zhejiang University. Her research interests include federated learning and data mining.

JIANJUN YANG received the M.S. degree in neurology from the Cheeloo College of Medicine, Shandong University, China, in 2011. He is currently pursuing the Ph.D. degree with the Shandong University of Traditional Chinese Medicine, China. He is the Director of the Department of General Practice of Shandong Provincial Third Hospital, Shandong University. He is also the Master Tutor of general medicine and neurology with Binzhou Medical University, China. His research interests include small cerebral vascular disease, vascular cognitive impairment, and Alzheimer's disease.

HONGYI ZHANG received the M.S. degree in computer science from Zhejiang University, Hangzhou, China. Currently, he is a Research Staff with Alibaba Group. His research interests include machine learning and data mining.

XIN JIE received the B.S. degree in communication engineering from the Nanjing Institute of Technology, China, in 2020. He is currently pursuing the M.S. degree in computer science with Zhejiang University, China. His current research interests include data mining and natural language processing.

DANUSHKA BANDARA received the bachelor's degree in electrical engineering from the University of Moratuwa, Sri Lanka, in 2009, and the master's and Ph.D. degrees in computer engineering and electrical and computer engineering from Syracuse University, Syracuse, NY, USA, in 2013 and 2018, respectively. From 2019 to 2020, he was a Data Scientist with Corning Inc., Corning, NY, USA. Currently, he is an Assistant Professor of computer science and engineering with Fairfield University, Fairfield, CT, USA. His current research interests include applied machine learning, bioinformatics, human–computer interaction, and computational social science.