This ACCV 2024 paper, provided here by the Computer Vision Foundation, is the author-created version.
The content of this paper is identical to the content of the officially published ACCV 2024
LNCS version of the paper as available on SpringerLink: https://link.springer.com/conference/accv

A Generic Autoregressive Predictive Feedback Framework for Skeleton-Based Action Recognition

Xinpeng Yin[1][0000−0002−7310−7488], Jing Hu[2], and Wenming Cao[1][0000−0002−8174−6167]

[1] The State Key Laboratory of Radio Frequency Heterogeneous Integration, Department of Electronics and Information Engineering, Shenzhen University, Shenzhen 518060, China (2110436215@email.szu.edu.cn; wmcao@szu.edu.cn)
[2] Department of Mathematics, The University of Massachusetts Boston 100 William T. Morrissey Boulevard Boston, MA 02125 (Jing.Hu@umb.edu)

Abstract. Prior works in skeleton-based action recognition have struggled with overcoming sequence order constraints while achieving comprehensive global modeling of temporal dependencies. However, most focus on enhancing the model's fitting ability across different temporal scales, overlooking the temporal non-stationary characteristics inherent in motion sequences. In this paper, we explore the adaptation of state-space modeling (SSM), typically suited for stationary sequences, to motion sequences. Addressing the challenge posed by the trendiness of motion sequences and the stability requirement of SSM, we integrate SSM into a generalized Autoregressive Predictive Feedback (APF) framework. Our approach involves segmenting motion sequences into trend and stationary components. We introduce the Non-Independent Multi-channel Processing (NiMc-P) module to capture implicit relationships among 3D coordinates and propose the Independent Multi-joint SSM (IMj-S) module to model temporal dependencies within stationary components. Throughout this process, state space matrices drive the feedback mechanism. Experiments conducted on the NTU-RGB+D 60 and NTU-RGB+D 120 datasets demonstrate the efficiency and versatility of APF.

Keywords: Action recognition · Autoregressive predictive · Long-term temporal dependence

1 Introduction

Human motion sequences are interdisciplinary spatio-temporal entangled signals, and identifying the actions they contain is the foundation of extensive signal processing applications [4,21,32,47]. Skeleton-based action recognition has garnered significant attention due to the advantages of 3D skeleton sequences, including their immunity to interference from video background [23] and high information aggregation [42]. The crux of this task lies in effectively modeling skeleton sequences to extract comprehensive and representative spatial and temporal features.

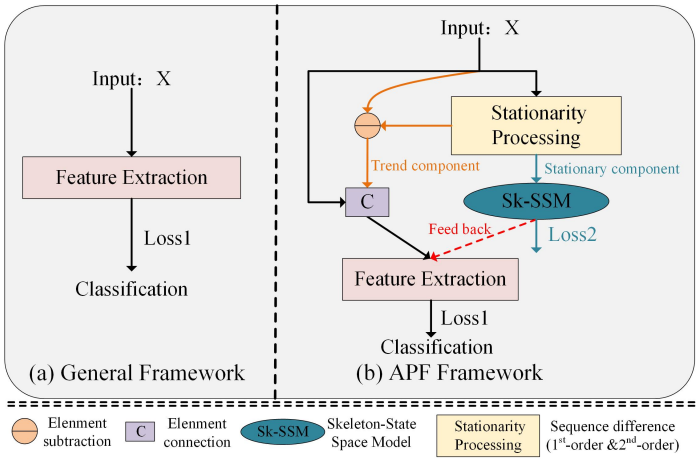

Fig. 1. (a) General framework. (b) Autoregressive predictive feedback framework (APF). The loss function of the APF is $Loss1 + \alpha * Loss2$. α is a hyperparameter.

Since Yan et al. [39] introduced the application of Graph Convolutional Network (GCN) to model spatial features and achieved remarkable results, numerous GCN-based optimization methods have been proposed [1, 3, 7, 20]. Regarding temporal modeling, previous methods can be generally categorized into two groups. The first group employs the Temporal Convolutional Network (TCN), which relies on 1D convolution to aggregate information across frames [10, 31]. While this approach is straightforward and robust, it tends to become rigid when modeling long-term temporal dependencies, as irrelevant information accumulates with the increasing convolution kernel size, k. The second utilizes attention mechanisms to overcome the limitations imposed by the sequence order, establishing connections between any two frames through the inner product of the temporal sequence [11, 12, 18, 19, 25, 27, 44]. However, due to its inherent design flaws, the attention mechanism, while relaxing the sequential constraints of the sequence, limits the global perspective of the model. The relationship between the i_{th} frame and the j_{th} frame cannot be supplemented or optimized by incorporating information from the k_{th} frame. In other words, it loses a portion of the sequential information, which plays a crucial role in skeleton sequence with clear trends.

To alleviate the limitations of the two aforementioned methods in modeling time dependencies, we enhance state space modeling (SSM) [6,43] for application to skeleton sequences. Note that SSM necessitates stationary sequences. However, motion sequences inherently exhibit deterministic trends, and the positive importance of these features in sequence recognition can not be disregarded.

Therefore, we incorporate SSM into a general Autoregressive Predictive Feedback (APF) framework, as illustrated in Fig. 1, to alleviate the aforementioned contradictions. Specifically, we initially segment the skeleton sequence into trend

and stationary components and treat the trend components as separate features to revise the model. Parallelly, the SSM is enhanced to Sk-SSM to achieve long-term temporal dependence modeling in the stationary components. The state-space matrices generated throughout the process catalyze optimizing model feedback. In Sk-SSM, we propose a non-independent multi-channel processing (NiMc-P) module to map the implicit relationships between the 3D coordinates of a single joint, and optimize the SSM from the perspective of treating the skeleton sequence as a collection of independent sequences of multiple joints, thereby designing an independent multi-joint SSM (IMj-S) module to model the long-term temporal dependency behavior of the stationary components of the skeleton.

The main contributions of this paper can be summarized as follows: 1) SSM is incorporated into a general Autoregressive Predictive Feedback (APF) framework to complete skeleton-based recognition tasks. 2) NiMc-P and IMj-S are designed to effectively model the temporal aspects of the skeleton sequences within Sk-SSM. 3) APF shows efficiency and universality on two large-scale datasets: NTU-RGB+D 60 and NTU-RGB+D 120.

2 Related work

2.1 Action Recognition Based on GCNs

Unlike the pixel array structure of images, skeleton sequences exhibited a tree-like topology. Therefore, early attempts to apply Convolutional Neural Networks (CNNs) within the deep learning paradigm were ill-suited for handling skeleton data. Yan et al. [39] pioneered using GCN to dynamically model spatial characteristics, extending it to ST-GCN to capture spatiotemporal changes. However, its convolution kernel was limited to aggregating local features and lacked robustness in detecting associations among non-connected joints. Subsequent efforts primarily focused on improving GCNs to more effectively model the spatial features of skeletons. Notable approaches included Li et al. [15], who utilized a classic encoder-decoder framework to categorize inter-joint connections into A-links and S-links, where A-links represented dynamic non-adjacent connections between joints and S-links denoted original topological adjacency relationships. Liu et al. introduced MS-G3D [20], which successfully addressed biases in weight assignments within multi-scale adjacency matrices. They defined inter-joint connectivity across adjacent frames using single-frame adjacency matrices, thereby capturing complex spatial-temporal joint features between skeleton sequences. Chen et al. proposed CTR-GCN [1], optimizing skeleton topology from various input channels and categorizing GCN methods in motion recognition based on four types of adjacency matrices (static, dynamic, topology-shared, and topology-non-shared). Based on [1], [45] introduced contrastive learning within an unsupervised framework, developing a plug-and-play judgment module to enhance the model's ability to learn features with high inter-class similarity. Wu et al. [37] utilized GCN as the backbone for unsupervised action recognition,

achieving state-of-the-art results in the unsupervised paradigm. However, optimizing the original topology matrix inevitably rendered the model insensitive to the inherent topological structure of skeleton sequences. Recent efforts explored using inter-joint distances as a basis for joint connectivity. For instance, [40] calculated distances between joint hidden layer features across different frames to capture spatio-temporal entangled features of skeleton sequences. Similarly, [46] employed joint distances to encode skeleton connections, offering a more accurate representation of the physical topology. The emphasis of the above works was primarily on the spatial features; hence, the temporal feature-oriented APF was effectively integrated. To demonstrate the effectiveness of APF, baselines in the experimental section included [25] and [10].

2.2 Action Recognition Based on Attentions

Self-attention was first proposed in [34] for focusing on keywords in text analysis within natural language processing (NLP). Due to its powerful feature extraction capabilities and inherent ability to break sequence order, it became popular in skeleton-based human action recognition. Plizzari et al. [19] were the first to apply it in proposing the ST-TR, which more accurately reflected spatiotemporal dependencies between joints than GCN-based methods. However, it tended to overestimate relationships between certain joints. To address this, Liu et al. [27] proposed KA-AGTN, which used adjacency matrices to correct these dependencies. Recently, attention mechanisms were combined with multi-scale approaches. [13] adaptively decomposed each node into multiple sets to highlight major hierarchical edges. This was similar to the hypergraph concept in [35], where sets were hyperedges, and one joint could belong to different hyperedges, each with different weight scores. [22] improved fine-grained attention to action samples by adaptively perceiving distinguishable receptive fields based on spatiotemporal feature changes, focusing the model on the most informative joints. [26] proposed a multi-grained clip focus network (MGCF-Net), dividing the skeleton sequence into different granularities from both temporal and spatial perspectives, using multi-head attention to integrate contextual features. The Graph Weight Annealing (GWA) method proposed in [9] alleviated over-smoothing caused by multi-scale aggregation by adjusting the importance between vertices and their neighbors. To demonstrate the generalization ability of the APF proposed in this paper, we integrated APF into GCN-based baselines and the attention-based approach of [13] in the experimental section.

2.3 Time Series Modeling and Prediction

Motion sequences, a subtype of time series data, were typically modeled using methodologies inspired by time series analysis. Initially, researchers frequently employed various variants of Recurrent Neural Networks (RNNs) [8,16]. The emergence of Transformers introduced notable advancements in long-term time series forecasting solutions, significantly enhancing prediction accuracy. Song et al. [30] integrated masked mechanisms with Transformers in the SAnD

model, successfully introducing self-attention into clinical time series analysis tasks. Tarasiou et al. [33] enhanced model discriminative power by incorporating acquisition-time-specific temporal positional encodings and multiple learnable class tokens. Informer [44] tackled Transformers' quadratic computational complexity by exploiting sparsity in correlation matrices, thereby reducing computational overhead to $\mathcal{O}(N \log N)$, where N denotes sequence length. Despite using positional encoding to retain some order information, temporal information loss remains inevitable post-self-attention [41]. While this loss is generally less concerning in semantically rich applications like NLP, it becomes significant for time series exhibiting clear trends and periodicity. Therefore, we shift our focus to autoregressive approaches.

3 Preliminaries

Notations. Skeleton sequences $\mathbf{X} \in \mathbb{R}^{C \times T \times N}$ consists of T graphs $\mathcal{G} = (\nu, \varepsilon)$, where $\nu = \{v_1, v_2, \cdots, v_N\}$ is the set of N joints and ε represents the topological relationships between them. C=3 indicates that the initial features of each joint are composed of its 3D coordinates (x, y, z).

Discrete-time state-space models. SSM maps the inputs $u_k \in \mathbb{R}^{p \times c}$ to hidden states $h_k \in \mathbb{R}^{p \times p}$, before projecting back to outputs $y_k \in \mathbb{R}^{c' \times p}$. p represents the length of the "lag" series, c and c' stand for the feature size of u_k and y_k, respectively. When $\boldsymbol{u}$ and $\boldsymbol{y}$ can be treated as copies of the same process, SSM can be formulated as

$$y_{k+1} = u_{k+1} = \boldsymbol{C}(\boldsymbol{A}h_k + \boldsymbol{B}u_k) \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{p \times p}$, $\mathbf{B} \in \mathbb{R}^{p \times c}$, and $\mathbf{C} \in \mathbb{R}^{c' \times p}$ are all state space matrices. We can predict $\tilde{y}_{k+1}$ that matches the next time-step sample $y_{k+1} = u_{k+1}$ by embedding $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ as trainable parameters into the neural network with input u_k and hidden state h_k.

Gu et al. [5] modeled the SSM as a convolution of Eq.2 to alleviate the problem of slow operating the recursive property in Eq.1.

$$y_k = \sum_{j=0}^{k-1} \boldsymbol{C}\boldsymbol{A}^{k-1-j}\boldsymbol{B}u_j \tag{2}$$

Further, Eq.2 can be transformed into matrix multiplication in Eq.3, greatly improving the calculation efficiency.

$$y_k = \left(\boldsymbol{C}\boldsymbol{A}^{\mathbf{0}}\boldsymbol{B}, \cdots, \boldsymbol{C}\boldsymbol{A}^{\boldsymbol{k-1}}\boldsymbol{B}\right)\left(u_{k-1} \cdots, u_0\right)^{\mathbf{T}} \tag{3}$$

Problem transformation. In the process of migrating SSM to skeleton-based action recognition, we identify the following issues that need to be addressed:

– **Optimization of stationarity issue:** Having stationarity is a prerequisite for sequences to be modeled by SSM. However, motion sequences have a

deterministic trend, which is non-stationary, and it can improve the model's ability to recognize actions. How to balance this contradictory relationship is the premise of this paper.
- **The transformation issue of state space matrices:** The state space matrices in Eq. 3 are designed based on a time series following the single input, single output protocol. However, each time step of the skeleton sequence is composed of V joints, and the initial features of each joint are represented by its 3D coordinates. How to transform the state space matrices to conform to the form of the skeleton sequence is the foundation of this paper.
- **The application issue of state space matrices:** After modeling and embedding the skeleton sequence into the framework, arbitrary desired state space matrices can be learned through supervised learning. How to utilize the trained $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ to feed back the recognition model is the significance of this paper.

We will analyze the above issues in Sec 4.2, Sec 4.3, and Sec 4.4, respectively.

4 Method

4.1 Overall Autoregressive Predictive Feedback Framework

We improve the general framework (Fig. 1(a)) to APF (Fig. 1(b)) to achieve a more reasonable modeling of skeleton sequences. APF divides the skeleton sequence into trend components (yellow) and stationary components (blue).

The input of the feature extraction module $F_e(\mathbf{X})$ in APF can be formulated as:

$$F_e(\mathbf{X}) = \mathrm{Cat}\left(\mathbf{X}, (\mathbf{X} - \mathrm{S_p}(\mathbf{X}))\right) \oplus F_b\left(\mathrm{SkS}\left(\mathrm{S_p}(\mathbf{X})\right), S_p(\mathbf{X})\right) \tag{4}$$

where S_p represents the stationary processing, SkS represents the modeling of stationary components, F_b represents the way feedback features are obtained. $\oplus$ represents the application of F_b, which will be reflected in Sec 4.4. Cat represents the connection function. Additionally, as the foundation for training the state-space matrices, APF computes the stationary components' loss before and after the SK-SSM operation. To aid in comprehension, we describe the complete data flow, as shown in Alg. 1.

4.2 Stationarity Processing: Optimization of stationarity issue

Certain joints exhibit deterministic trends during motion, and these perceptible but non-concrete features hidden in all frames are the trend components of the sequence, with the remaining being stationary components. Considering differencing to handling nonstationary generally [2,28], and the higher-order differential processing will bring more distortion, this paper only uses the 1st-order and 2nd-order differences of sequences as the material for SSM modeling.

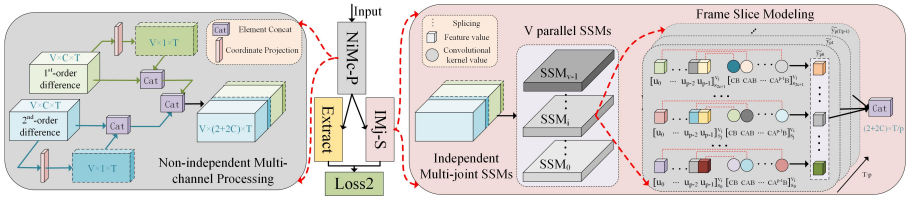

Fig. 2. The details of **Sk-SSM**, including non-independent multi-channel processing (**NiMc-P**) module and independent multi-joint SSM (**IMj-S**) module. **Extract** module realizes to filter out the next matching time step sample for loss compared with the predicted results. The orange dash lines in **IMj-S** represent a multiplication relationship.

4.3 Sk-SSM: the transformation issue of state space matrices

Human motion can be seen as an autoregressive(AR) process [24]. Consider a noiseless and unbiased AR(p) process:

$$u_k = \phi_1 u_{k-1} + \phi_2 u_{k-2} + \dots \phi_p u_{k-p} \tag{5}$$

where u_k is a linear combination of p prior samples (with coefficients $\phi_1, \cdots, \phi_p$). The work in [5] has shown that when $\mathbf{A} \in \mathbb{R}^{p\times p}$ takes the form shown in Eq.6 and setting $a_i = \mathbf{0}$, $\boldsymbol{B}=[1,0,\dots,0]^T$, $\boldsymbol{C}=[\phi_1,\dots,\phi_p]$ allows the SSM in Eq.3 to recover the AR(p) in Eq.5.

$$\boldsymbol{A}_{i,j} = \begin{cases} 1 & \text{for } i-1=j \\ a_i & \text{for } j=p-1 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

When modeling the skeleton sequences as an AR process, Eq.5 will be transformed into:

$$\begin{aligned} \left[u_{p+1}^{:v}\right] &= \sum_{i=0}^{p-1} \left(\left[\phi_i^{:v}\right] \cdot^* \left[u_i^{:v}\right]\right) \\ &= \sum_{i=0}^{p-1} \left(\begin{bmatrix} \phi_0^0 & \phi_1^0 & \cdots & \phi_{p-1}^0 \\ \phi_0^1 & \phi_1^1 & \cdots & \phi_{p-1}^1 \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0^{v-1} & \phi_1^{v-1} & \cdots & \phi_{p-1}^{v-1} \end{bmatrix} \cdot^* \begin{bmatrix} u_0^0 & u_1^0 & \cdots & u_{p-1}^0 \\ u_0^1 & u_1^1 & \cdots & u_{p-1}^1 \\ \vdots & \vdots & \ddots & \vdots \\ u_0^{v-1} & u_1^{v-1} & \cdots & u_{p-1}^{v-1} \end{bmatrix} \right) \\ &= \begin{bmatrix} \phi_0^0 u_0^0 + \phi_1^0 u_1^0 + \cdots + \phi_{p-1}^0 u_{p-1}^0 \\ \phi_0^1 u_0^1 + \phi_1^1 u_1^1 + \cdots + \phi_{p-1}^1 u_{p-1}^1 \\ \vdots \\ \phi_0^{v-1} u_0^{v-1} + \phi_1^{v-1} u_1^{v-1} + \cdots + \phi_{p-1}^{v-1} u_{p-1}^{v-1} \end{bmatrix} \in \mathbb{R}^{v\times 1} \end{aligned} \tag{7}$$

The issue is now transformed into constructing state-space matrices to be applied to Eq.7. Fig.2 shows the SSM modeling process of skeleton sequences.

Skeleton sequences can be seen as temporal changes of V joints. Therefore, we design IMj-S to construct V parallel SSMs to model it, and considering the low correlation between frames that are too far away in the skeleton sequence, we divide p frames into a group. The corresponding state space matrices are: $\mathbf{A} \in \mathbb{R}^{v\times p\times p}$, $\mathbf{B} \in \mathbb{R}^{v\times p\times 1}$, $\mathbf{C} \in \mathbb{R}^{v\times 1\times p}$.

Meanwhile, since the initial features of each joint in the skeleton sequences are represented by its 3D coordinates, naturally, we add another dimension to each spatial state matrix: $\mathbf{A} \in \mathbb{R}^{v\times s\times p\times p}$, $\mathbf{B} \in \mathbb{R}^{v\times s\times p\times 1}$, $\mathbf{C} \in \mathbb{R}^{v\times s\times 1\times p}$. However, it is unreasonable that this approach is equivalent to the fact that the 3D coordinates of the default joint are independent. We design NiMc-P to add a new row based on the s-dimension, with the s+1 slice representing the SSM that projects the 3D coordinates onto one dimension. The corresponding loss function can be formulated as:

$$\text{Loss2} = \sum_{j=0}^{T/p-1} \sum_{i=0}^{V-1} \text{MSE}\left(\mathbf{u}_{pj}^{i}, \tilde{\mathbf{y}}_{pj}^{i}\right) \tag{8}$$

where $\tilde{y}_{pj}^{i}$ and u_{pj}^{i} represent the prediction and the original results of the j_{th} group, which both belong to the i_{th} joint. MSE represents the mean square error loss function.

4.4 Feedback: the application issue of state space matrices

After Sec 4.3, T/p sets of trainable convolution kernels $\mathbf{K} \in \mathbb{R}^{s\times 1\times p}$ for each joint are obtained. We multiply $\mathbf{K}$ with the corresponding original information and concatenate the obtained features with the features in Sec 4.2 into the feature extraction module. The right side of Eq.4 is specifically:

$$\begin{aligned} &a \oplus b = \text{Cat}(a, b) \\ &b = \text{Cat}\left(b(i)\right), i = 1, 2, \cdots, T/p \\ &b(i) = \sigma\left(\left(\mathbf{K_{(i)}} + \mathbf{K}_{\text{res(i)}}\right) S_p(\mathbf{X_{(i)}})\Theta_{(i)} + r_{\text{res(i)}}\right) \end{aligned} \tag{9}$$

where $\mathbf{K}_{\text{res(i)}}$ is a trainable matrix with the same dimension as $\mathbf{K}$. $\Theta_{(i)}$ denotes a learnable weight matrix in group i, $r_{res(i)}$ represents residual connection, and $\sigma(\cdot)$ is an activation function.

5 EXPERIMENTS

5.1 Datasets

NTU-RGB+D 60 [29] contains 60 types of actions, and all actions are performed by 40 individuals. It uses two division criteria when dividing the training and test sets. 1) X-Sub: The training and test sets are divided according to the person ID. 2) X-View: The training and test sets are divided according to the camera.

Algorithm 1 APF Data Flow

Step 1: Stationarity Processing
$X, X'\&X'' \leftarrow$ Input,$1^{st}\&2^{nd}$-order difference
$S_c \leftarrow concat[X', X'']$ (Stationary components)
$T_c \leftarrow concat[X - X', X - X'']$ (Trend components)

Step 2: Feature Mapping(NiMc-P)
$S_c \leftarrow concat[X', map(X'), X'', map(X'')]$
$2 + 2C = 8$ (Left part of Fig. 2)

Step 3: State Space Modeling (SSM)
$S_c \leftarrow S_{c0}, S_{c1}, ..., S_{c(T/p-1)}$
# T frames divided into non-overlapping T/p groups.
for each group, each joint v in the first p-1 frames **do**
  $SSM(v) \leftarrow$ Predict 1 frame based on p-1 frames
  # v are modeled independently.
  # Each v includes 8 feature flows in parallel.
end for (Right part of Fig. 2)

Step 4: Extract & Loss2
Extract$(S_c) \leftarrow$ extract the p frame in each group
Loss2 $\leftarrow$ MSE[$SSM(v)$, Extract(S_c)]

Step 5: Feedback
$K, K_{res} \leftarrow$ state space matrix, trainable matrix
Predict frame $\leftarrow (K + K_{res}) \times S_c\Theta + res$
Just minor changes to $SSM(v)$
Final input $\leftarrow concat[X, T_c,$ Predict frame]
Fill the remaining T-T/p frames with 0.

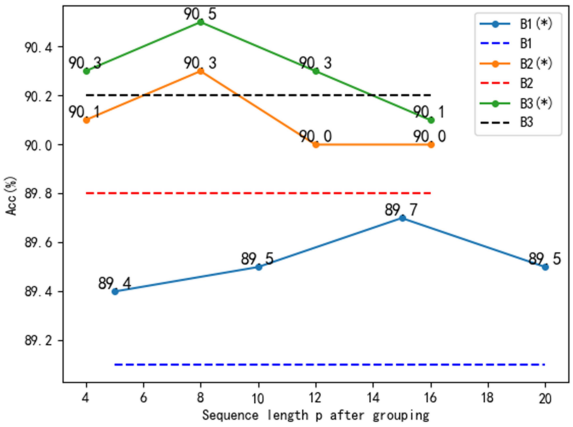

Fig. 3. Comparing models' accuracy on different lengths of skeleton sequences. Bi(*) represents inserting the APF in Bi.

Table 1. Ablation recognition accuracy(%) on the X-Sub of NTU-RGB+D 60 dataset. Tc-res & Tc-cat represent residual and concatenation operations on the trend components. * indicates NiMc-P module.

Methods					**Acc**(%)		
Tc-res	Tc-cat	w(*)	wo(*)	Fb	B1	B2	B3
-	-	-	-	-	89.1	89.8	90.2
√	-	-	-	-	89.4	90.1	90.2
-	√	-	-	-	89.7	89.6	90.5
-	-	√	-	-	89.7	90.3	90.5
-	-	-	√	-	89.5	90.0	90.3
-	√	√	-	-	90.0	90.0	90.6
√	-	√	-	-	89.8	90.4	90.3
√	-	-	√	-	89.6	90.2	90.0
-	√	-	√	-	89.8	89.9	90.5
-	-	√	-	√	90.0	90.5	90.6
-	-	-	√	√	89.7	90.2	90.4
-	√	√	-	√	**90.2**$^{\uparrow 1.1}$	90.4	90.6
√	-	√	-	√	**90.2**$^{\uparrow 1.1}$	**90.6**$^{\uparrow 0.8}$	**90.8**$^{\uparrow 0.6}$

NTU-RGB+D 120 [17] extends 60 categories based on NTU-RGB+D 60, and the effective number of samples is 113945. The author recommends two subsets similar to NTU-RGB+D 60: X-Sub and X-Set.
Northwestern-UCLA(N-UCLA) [36] contains 1494 samples of 10 classes. The dataset follows the same evaluation protocol as in a previous study, in which the training data comes from the first two cameras, and testing data comes from the other camera.

5.2 Experimental details

To demonstrate the excellent generalization ability of APF, all parameters related to the model are the same as Baselines. Where SGD had momentum of 0.9 and batch size of 64. The training epoch is set to 50/65 and the learning rate is set to 0.1 and decay with a factor of 0.1 at epochs (30, 40)/(35, 55) in baseline1/baseline2. In baseline3, the learning rate decays from 0.02 to 0.0001. The experiments are conducted on two Nvidia RTX (3090, 24GB) GPUs.

Table 2. Comparison of Params and Flops before and after Embedding APF. B1,B2,B3 represent baseline1,baseline2 and baselien3, respectively. $^{+}$ represents inserting APF.

Methods	B1	B1^{+}	B2	B2^{+}	B3	B3^{+}
Params (M)	1.44	1.49	3.21	3.27	1.66	1.71
Flops (G)	29.95	30.21	1.97	2.15	3.73	3.86
Inference Time(H)	19.1	19.8	8.9	9.4	12.4	13.0

5.3 Ablation Studies

APF consists of three separate and unified parts. We gradually embedded each part into the baseline1(MS-G3D [20]), baseline2(CTR-GCN [1]) and baseline3(HD-GCN [13]) on the X-Sub benchmark of the NTU-RGB+D 60 dataset to demonstrate the effectiveness of each part in both independent and joint states, and the results are shown in Tab 1. It can be seen that:

- Incorporating trend components into the model as residual terms has a positive effect; Baseline2 uses nonlinear changes in distance between different channels as the basis for classifying actions, while the distance calculated after concatenating trend information and raw data is meaningless, resulting in reduced model performance.
- The NiMc-P module is indispensable while modeling the stationary components, and incorporating a feedback module after modeling can further fine-tune the model's performance. The accuracy of the three baselines increased by 1.1%, 0.8%, and 0.6%, respectively, after introducing APF.

We slice the skeleton sequence to mitigate the error accumulation effect in autoregressive models. Fig.3 demonstrates the ability of Sk-SSM to model sequences of different lengths. p is taken as (5, 10, 15, 20) and (4, 8, 12, 16) according to original baselines. To address the issue of model bloating and parameter redundancy resulting from backpropagation of the parameters of the 300/P groups in baseline1, we randomly chose 6 groups for training and optimization, leaving the remaining groups with a value of 0. The results show that Sk-SSM modeling has superior performance in long-term temporal dependence.

Table 3. Comparisons with state-of-the-art methods for NTU-RGB+D 60, NTU-RGB+D 120 datasets and N-UCLA.

Methods	**NTU-RGB+D 60**		**NTU-RGB+D 120**		**N-UCLA**
	X-Sub (%)	X-View (%)	X-Sub (%)	X-Set (%)	
Ta-CNN+ [38]	90.7	95.7	85.7	87.3	97.2
MS-G3D [20]	91.5	96.2	86.9	88.4	95.8
CTR-GCN [1]	92.4	96.8	88.7	90.1	96.5
KA-AGTN [19]	90.4	96.1	86.1	88.0	-
LKA [27]	90.7	96.1	86.3	87.8	-
Ishift-GCN [14]	91.7	96.8	-	-	95.0
AM-GCN [10]	90.3	95.2	86.4	88.2	-
HD-GCN [13]	92.4	96.6	89.1	90.6	96.6
SPINet [40]	92.8	96.8	89.2	90.4	-
MSS-GCN [9]	92.7	96.9	88.9	90.6	96.6
APF+MS-G3D	**92.4**$^{\uparrow 0.9}$	**96.6**$^{\uparrow 0.4}$	**87.5**$^{\uparrow 0.6}$	**88.6**$^{\uparrow 0.2}$	**96.3**$^{\uparrow 0.5}$
APF+CTR-GCN	**92.9**$^{\uparrow 0.5}$	**96.9**$^{\uparrow 0.1}$	**89.3**$^{\uparrow 0.6}$	**90.8**$^{\uparrow 0.7}$	**97.2**$^{\uparrow 0.7}$
APF+HD-GCN	**93.0**$^{\uparrow 0.4}$	**96.9**$^{\uparrow 0.3}$	**89.5**$^{\uparrow 0.6}$	**91.1**$^{\uparrow 0.5}$	**97.0**$^{\uparrow 0.4}$

5.4 Parameter and FLOPS Analysis

The parameters of APF mainly include the elements in the T/p groups' state-space matrices and the necessary fully connected layers when connecting various parts. Tab 2 indicates a slight increase in the parameters and FLOPS count of the baselines after the addition of APF, suggesting that APF falls into the category of lightweight networks.

5.5 Comparison With State-of-The-Art Methods

We compare the performance of the proposed method with other state-of-the-art skeleton-based action recognition methods on NTU-RGB+D 60 and NTU-RGB+D 120. The results are reported in Tab 3.

As a plug-and-play framework, to demonstrate the effectiveness of APF, we embedded it into the baselines with accuracy at the first rank. Note that the stationary components in APF are physically equivalent to motion information, a dual-stream fusion (joint and bone) instead of a four-stream (extra joint motion and bone motion) architecture has been adopted for a fair comparison. The results indicate that the APF framework can help models already in a dominant position achieve further improvements.

6 Conclusion

This work designs a generic autoregressive predictive feedback (APF) framework to model sequences in skeleton-based action recognition. Note that APF is also suitable for other tasks that require modeling deterministic trend sequences and we will try to migrate APF to them for further research.

Acknowledgments. This work is supported by the National Natural Science Foundation of China under grant 61771322 and the Shenzhen Science and Technology Program under Grant JCYJ20220531100814033.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. 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. pp. 13359–13368 (2021)
2. Ensafi, Y., Amin, S.H., Zhang, G., Shah, B.: Time-series forecasting of seasonal items sales using machine learning–a comparative analysis. International Journal of Information Management Data Insights **2**(1), 100058 (2022)
3. Gao, J., He, T., Zhou, X., Ge, S.: Skeleton-based action recognition with focusing-diffusion graph convolutional networks. IEEE Signal Processing Letters **28**, 2058–2062 (2021)

4. Ghorbel, E., Demisse, G., Aouada, D., Ottersten, B.: Fast adaptive reparametrization (far) with application to human action recognition. IEEE Signal Processing Letters **27**, 580–584 (2020)
5. Gu, A., Goel, K., Re, C.: Efficiently modeling long sequences with structured state spaces. In: International Conference on Learning Representations (2021)
6. Gu, A., Johnson, I., Goel, K., Saab, K., Dao, T., Rudra, A., Ré, C.: Combining recurrent, convolutional, and continuous-time models with linear state space layers. Advances in neural information processing systems **34**, 572–585 (2021)
7. He, C., Zhang, J., Yao, J., Zhuo, L., Tian, Q.: Meta-learning paradigm and cosattn for streamer action recognition in live video. IEEE Signal Processing Letters **29**, 1097–1101 (2022)
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
9. Jang, S., Lee, H., Kim, W.J., Lee, J., Woo, S., Lee, S.: Multi-scale structural graph convolutional network for skeleton-based action recognition. IEEE Transactions on Circuits and Systems for Video Technology (2024)
10. Kang, M.S., Kang, D., Kim, H.: Efficient skeleton-based action recognition via joint-mapping strategies. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 3403–3412 (2023)
11. Kim, S., Ji, W., Deng, S., Ma, Y., Rackauckas, C.: Stiff neural ordinary differential equations. Chaos: An Interdisciplinary Journal of Nonlinear Science **31**(9), 093122 (2021)
12. Kong, J., Bian, Y., Jiang, M.: Mtt: Multi-scale temporal transformer for skeleton-based action recognition. IEEE Signal Processing Letters **29**, 528–532 (2022)
13. Lee, J., Lee, M., Lee, D., Lee, S.: Hierarchically decomposed graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10444–10453 (2023)
14. Li, C., Li, S., Gao, Y., Guo, L., Li, W.: Improved shift graph convolutional network for action recognition with skeleton. IEEE Signal Processing Letters **30**, 438–442 (2023)
15. Li, M., Chen, S., Chen, X., Zhang, Y., Wang, Y., Tian, Q.: Actional-structural graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3595–3603 (2019)
16. Li, Y., Yu, R., Shahabi, C., Liu, Y.: Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. arXiv preprint arXiv:1707.01926 (2017)
17. 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 (2019)
18. Liu, S., Yu, H., Liao, C., Li, J., Lin, W., Liu, A.X., Dustdar, S.: Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In: International conference on learning representations (2021)
19. Liu, Y., Zhang, H., Xu, D., He, K.: Graph transformer network with temporal kernel attention for skeleton-based action recognition. Knowledge-Based Systems **240**, 108146 (2022)
20. 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)

21. Lv, S., Lu, Y., Dong, M., Wang, X., Dou, Y., Zhuang, W.: Qualitative action recognition by wireless radio signals in human–machine systems. IEEE Transactions on Human-Machine Systems **47**(6), 789–800 (2017)
22. Myung, W., Su, N., Xue, J.H., Wang, G.: Degcn: Deformable graph convolutional networks for skeleton-based action recognition. IEEE Transactions on Image Processing **33**, 2477–2490 (2024)
23. Peng, W., Shi, J., Zhao, G.: Spatial temporal graph deconvolutional network for skeleton-based human action recognition. IEEE signal processing letters **28**, 244–248 (2021)
24. Qazani, M.R.C., Asadi, H., Lim, C.P., Mohamed, S., Nahavandi, S.: Prediction of motion simulator signals using time-series neural networks. IEEE Transactions on Aerospace and Electronic Systems **57**(5), 3383–3392 (2021)
25. Qiu, H., Wu, Y., Duan, M., Jin, C.: Glta-gcn: Global-local temporal attention graph convolutional network for unsupervised skeleton-based action recognition. In: 2022 IEEE International Conference on Multimedia and Expo (ICME). pp. 1–6. IEEE (2022)
26. Qiu, H., Hou, B.: Multi-grained clip focus for skeleton-based action recognition. Pattern Recognition **148**, 110188 (2024)
27. Qiu, H., Hou, B., Ren, B., Zhang, X.: Spatio-temporal segments attention for skeleton-based action recognition. Neurocomputing **518**, 30–38 (2023)
28. Salles, R., Belloze, K., Porto, F., Gonzalez, P.H., Ogasawara, E.: Nonstationary time series transformation methods: An experimental review. Knowledge-Based Systems **164**, 274–291 (2019)
29. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1010–1019 (2016)
30. Song, H., Rajan, D., Thiagarajan, J., Spanias, A.: Attend and diagnose: Clinical time series analysis using attention models. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
31. Song, Y.F., Zhang, Z., Shan, C., Wang, L.: Stronger, faster and more explainable: A graph convolutional baseline for skeleton-based action recognition. In: proceedings of the 28th ACM international conference on multimedia. pp. 1625–1633 (2020)
32. Sun, Z., Ke, Q., Rahmani, H., Bennamoun, M., Wang, G., Liu, J.: Human action recognition from various data modalities: A review. IEEE transactions on pattern analysis and machine intelligence (2022)
33. Tarasiou, M., Chavez, E., Zafeiriou, S.: Vits for sits: Vision transformers for satellite image time series. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10418–10428 (June 2023)
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
35. Wang, G., Liu, M., Liu, H., Guo, P., Wang, T., Guo, J., Fan, R.: Augmented skeleton sequences with hypergraph network for self-supervised group activity recognition. Pattern Recognition **152**, 110478 (2024)
36. Wang, J., Nie, X., Xia, Y., Wu, Y., Zhu, S.C.: Cross-view action modeling, learning and recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2649–2656 (2014)
37. Wu, C., Wu, X.J., Kittler, J., Xu, T., Ahmed, S., Awais, M., Feng, Z.: Scd-net: Spatiotemporal clues disentanglement network for self-supervised skeleton-based action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 5949–5957 (2024)

38. Xu, K., Ye, F., Zhong, Q., Xie, D.: Topology-aware convolutional neural network for efficient skeleton-based action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 2866–2874 (2022)
39. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
40. Yin, X., Zhong, J., Lian, D., Cao, W.: Spatiotemporal progressive inward-outward aggregation network for skeleton-based action recognition. Pattern Recognition **150**, 110262 (2024)
41. Zeng, A., Chen, M., Zhang, L., Xu, Q.: Are transformers effective for time series forecasting? In: Proceedings of the AAAI conference on artificial intelligence. vol. 37, pp. 11121–11128 (2023)
42. Zhang, L., Lim, C.P., Yu, Y.: Intelligent human action recognition using an ensemble model of evolving deep networks with swarm-based optimization. Knowledge-based systems **220**, 106918 (2021)
43. Zhang, M., Saab, K.K., Poli, M., Dao, T., Goel, K., Re, C.: Effectively modeling time series with simple discrete state spaces. In: The Eleventh International Conference on Learning Representations (2023)
44. Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W.: Informer: Beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 11106–11115 (2021)
45. Zhou, H., Liu, Q., Wang, Y.: Learning discriminative representations for skeleton based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10608–10617 (2023)
46. Zhou, Y., Yan, X., Cheng, Z.Q., Yan, Y., Dai, Q., Hua, X.S.: Blockgcn: Redefine topology awareness for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2049–2058 (2024)
47. Zhu, J., Zou, W., Zhu, Z., Xu, L., Huang, G.: Action machine: Toward person-centric action recognition in videos. IEEE Signal Processing Letters **26**(11), 1633–1637 (2019)