Aleš Leonardis · Elisa Ricci ·
Stefan Roth · Olga Russakovsky ·
Torsten Sattler · Gül Varol (Eds.)

LNCS 15099

Computer Vision – ECCV 2024

18th European Conference
Milan, Italy, September 29–October 4, 2024
Proceedings, Part XLI

41 Part XLI

ECCV

Springer

MOREMEDIA

Lecture Notes in Computer Science 15099

The series Lecture Notes in Computer Science (LNCS), including its subseries Lecture Notes in Artificial Intelligence (LNAI) and Lecture Notes in Bioinformatics (LNBI), has established itself as a medium for the publication of new developments in computer science and information technology research, teaching, and education.

LNCS enjoys close cooperation with the computer science R & D community, the series counts many renowned academics among its volume editors and paper authors, and collaborates with prestigious societies. Its mission is to serve this international community by providing an invaluable service, mainly focused on the publication of conference and workshop proceedings and postproceedings. LNCS commenced publication in 1973.

Aleš Leonardis · Elisa Ricci · Stefan Roth ·
Olga Russakovsky · Torsten Sattler · Gül Varol
Editors

Computer Vision – ECCV 2024

18th European Conference
Milan, Italy, September 29–October 4, 2024
Proceedings, Part XLI

Springer

Editors
Aleš Leonardis
University of Birmingham
Birmingham, UK

Elisa Ricci
University of Trento
Trento, Italy

Stefan Roth
Technical University of Darmstadt
Darmstadt, Germany

Olga Russakovsky
Princeton University
Princeton, NJ, USA

Torsten Sattler
Czech Technical University in Prague
Prague, Czech Republic

Gül Varol
École des Ponts ParisTech
Marne-la-Vallée, France

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-031-72939-3 ISBN 978-3-031-72940-9 (eBook)
https://doi.org/10.1007/978-3-031-72940-9

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Foreword

Welcome to the proceedings of the European Conference on Computer Vision (ECCV) 2024, the 18th edition of the conference, which has been held biennially since its founding in France in 1990. We are holding the event in Italy for the third time, in a convention centre where we can accommodate nearly 7000 in-person attendees.

The journey to ECCV 2024 started at CVPR 2019, where the general chairs met at a sketchy bar in Long Beach to discuss ideas on how to organize an amazing future ECCV. After that, Covid came and changed conferences probably forever, but we maintained our excitement to organize an awesome in-person conference in 2024, particularly having seen the wonderful event that was ECCV 2022 in Tel Aviv. And here we are, 5 years later, delighted that we finally get to welcome all of you to Milan!

First and foremost, ECCV is about the exchange of scientific ideas, and hence the most important organizational task is the selection of the programme. To our Program Chairs, Aleš Leonardis, Elisa Ricci, Gül Varol, Olga Russakovsky, Stefan Roth, and Torsten Sattler, our entire community owe a huge debt of gratitude. As you will read in their Preface to the proceedings, they dealt with over 8,500 submissions, and their diligence, thoroughness, and sheer effort has been inspirational. From the early stages of designing the call for papers, through recruiting and selecting area chairs and reviewers, to ensuring every paper has a fair and thorough assessment, their professionalism, commitment, and attention to detail has been exemplary. From all the choices that we made to organize ECCV, choosing this amazing team of Program Chairs has been, without a doubt, our very best decision.

The Program Chairs were advised and supported by an Ethics Review Committee, Chloé Bakalar, Kate Saenko, Remi Denton, and Yisong Yue, who provided invaluable input on papers where reviewers or Area Chairs had raised ethical concerns.

The Publication Chairs, Jovita Lukasik, Michael Möller, François Brémond, and Mahmoud Ali, did great work in assembling the camera-ready papers into these Springer volumes, dealing with numerous issues that arose promptly and professionally.

With the ever-expanding importance of workshops, tutorials, and demos at our major conferences comes a corresponding increase in the challenge of organizing over 70 workshops and 9 tutorials attached to the main conference. The Workshop and Tutorial Chairs, Alessio Del Bue, Jordi Pont-Tuset, Cristian Canton, and Tatiana Tommasi, and Demo Chairs, Hyung Chang and Marco Cristani, worked tirelessly to coordinate the very varied demands and structures of these different events, yielding an extremely rich auxiliary program which will enhance the conference experience for everyone who attends.

The Industry Chairs, Shaogang Gong and Cees Snoek, working with Limor Urfaly and Lior Gelfand, managed to attract numerous companies and organizations from all around the world to the industrial Expo, making ECCV 2024 an international event that not only showcases excellent foundational research, but also presents how such research can be transformed in real products. It is notable that together with the big companies, several start-ups chose ECCV to present their business ideas and activities.

As usual in the recent main conference, we also organized Speed Mentoring and Doctoral Consortium events. While the latter is an institutional occasion allowing senior PhDs to show their work, the former is especially important to answer to the many doubts and uncertainties young scholars have while undertaking a career in Computer Vision and Artificial Intelligence. We warmly thank our Social Activities Chairs, Raffaella Lanzarotti, Simone Bianco and Giovanni Farinella, and the Doctoral Consortium Chairs, Cigdem Beyan and Or Litany, for their dedication to organize such events, so important for our young colleagues, as well as all the mentors who were available to share their experience.

As the conference becomes larger and larger, the overall budget increases to a level where uncertainties in estimates of attendance can result in significant losses, perhaps enough to significantly damage the prospects of running the conference in the future. This risk imposed the need to require authors to commit to full conference registrations, with the undesirable consequent possibility to cause hardship to authors, particularly student authors, from organizations where funding attendance is difficult. To mitigate this effect, we allocated travel grants to students who might otherwise have had difficulty in attending. Our Diversity Chairs, David Fouhey and Rita Cucchiara, put tremendous effort into a wonderfully careful and thoughtful programme, balancing diversity in all its forms with a changing budget landscape, allowing us to support the participation of many people who might not otherwise have been able to attend.

This year's conference also marks the move to a unified, multi-year conference website, based on that used for other leading computer vision and AI conferences. This was an effort that we decided to take on to make the ECCV website more familiar to the community and support the organization of the future ECCV editions. Our thanks go to Lee Campbell of Eventhosts who managed the entire design and development of the new website and provided responsive support and updates throughout the conference organization timeline.

Our Finance Chairs, Gérard Medioni and Nicole Finn, provided invaluable advice and assistance on all aspects of the financial planning of the conference, in conjunction with the professional conference organizers AIM Group, where Lavinia Ricci led a fantastic team that managed thousands of details from space planning to food to registrations. In particular, Elder Bromley and Mara Carletti were a tremendous support to the entire conference organization. The conference centre, Mico DMC, also provided an excellent service, showing considerable flexibility in adapting the venue and processes to the particular demands of an academic-focused conference.

Finally, we cannot forget to thank our Social Media Chair, Kosta Derpanis, supported by Abby Stylianou and Jia-Bin Huang, who, with their presence on social media, spread ECCV news and responded to small and big questions in a timely manner, helping the community to promptly receive information and fix their problems.

October 2024

Andrew Fitzgibbon
Laura Leal-Taixé
Vittorio Murino

Preface

ECCV 2024 saw a remarkable 48% increase in submissions, with 8585 valid papers submitted, compared to 5804 in ECCV 2022, 5150 in 2020, and 2439 in 2018. Of these, 2387 papers (27.9%) were accepted for publication, with 200 (2.3% overall) selected for oral presentations.

A total of 173 complete submissions were desk-rejected for various reasons. Common issues included revealing author identities in the paper or supplementary material (for example: by including the author names, referring to specific grants in the acknowledgments, including links to GitHub accounts with visible author information, etc.), exceeding the page limit or otherwise significantly altering the submission template, or posting an arXiv preprint explicitly mentioning the work as an ECCV 2024 submission. The desk-rejected submissions also include 14 papers identified as dual submissions to other concurrent conferences, such as ICML, NeurIPS, and MICCAI. Seven papers were rejected among the accepted papers due to plagiarism issues, after the Program Chairs conducted a thorough plagiarism check with the iThenticate software followed by manual inspection.

The double-blind review process was managed using the CMT system, ensuring anonymity between authors and Area Chairs/reviewers. Each paper received at least three reviews, amounting to over 26,000 reviews in total. To maintain the quality and the fairness of the reviews, we recruited 469 Area Chairs (ACs) and more than 8200 reviewers. Of the recruited reviewers, 7293 ultimately participated.

Area Chairs were chosen for their technical expertise and reputation; many of them had served in similar roles at other top conferences. We also recruited additional Area Chairs through an open call for self-nominations. Among the selected ACs, around 18% were women. Geographical distribution was taken into account during the selection process, with 31% of ACs working in Europe, 38% in the Americas, 27% in Asia, and 4% in the rest of the world. Despite our best efforts to consider diversity factors in the selection process, the gender diversity and geographic diversity of ACs is still below what we were hoping for.

To ensure a smooth decision-making process, we grouped the ACs into triplets. Upon accepting their role, ACs were requested to enter into the CMT system domain conflicts and subject areas, as well as information about their DBLP, TPMS (Toronto Paper Matching System), and OpenReview profiles. These were used to determine whether different subject areas were sufficiently covered, to detect conflicts, and to automatically form AC triplets. An AC triplet was a group of three ACs who worked together throughout the process, which helped to facilitate discussion, calibrate the decision-making process, and provide support for first-time ACs. AC triplets were manually refined by Program Chairs to ensure no triplet contained more than two major time zones to enable easier coordination for virtual AC meetings. Paper assignments were made such that no AC within the triplet was conflicted with any of the papers. Each AC was responsible for overseeing 18 papers on average.

In each triplet, we identified a Lead AC who was an experienced member of the computer vision community willing to take on additional administrative duties. The primary responsibilities of Lead ACs included coordinating the AC triplet to ensure deadlines were met and offering guidance to less experienced ACs. They also served as the main contact person with the Program Chairs and helped in handling papers if an original AC was unable to complete their tasks.

Reviewers were invited from reviewer pools of previous conferences and also selected from the pool of authors. We asked experienced ACs to recommend more potential reviewers. We further recruited reviewers through self-nominations, where researchers volunteered to review for ECCV 2024 via an open call. The self-nominations were then filtered, selecting those reviewers with at least two papers published in related conferences such as ECCV, ICCV, CVPR, ICLR, NeurIPS, etc.

Similarly to ACs, reviewers were also asked to update their CMT, Toronto Paper Matching System (TPMS), and OpenReview profiles. Reviewers had the option to classify themselves as either senior researchers or junior researchers and students, with the number of assigned papers varying accordingly (a quota of 6 for senior researchers with 3 or more times reviewing experience, and a quota of 4 otherwise). On average, each reviewer was assigned about 4 papers.

Conflicts of interest among authors, ACs, and reviewers were managed automatically via conflict domains and co-authorship information from DBLP. Paper assignments to ACs and reviewers were determined using an algorithm that combined subject-area affinity scores from CMT, TPMS, and OpenReview, along with additional criteria such as ensuring that every paper had at least one non-student reviewer. For AC assignments, we additionally incorporated a newly built affinity score based on embedding similarity with the papers on the Google Scholar profile of the ACs. Once the assignments were made, ACs and reviewers were asked to promptly report any undetected conflicts of interest or other concerns that would preclude them from handling the assigned papers to ensure immediate reassignment, if necessary.

Given the widespread use of Large Language Models (LLMs), ECCV 2024, like previous conferences, implemented a special policy regulating their use. Authors were permitted to use any tools, including LLMs, in preparing their papers, but they remained fully responsible for any misrepresentation, factual inaccuracies, or plagiarism. Reviewers were also allowed to use LLMs to refine the wording of their reviews; however, they were held accountable for the accuracy of their content and were strictly prohibited from inputting submissions into an LLM. One challenge we ran into was that tools that detect LLM-generated content were not able to distinguish between text polished vs. text written from scratch by an LLM. Thus, it became quite difficult to enforce the policy in practice.

Overall, the challenges encountered during the review process were consistent with those faced at previous computer vision conferences. For instance, a small number of reviewers ultimately did not submit their reviews or provided brief, uninformative reviews, necessitating last-minute reassignment to emergency reviewers. As the community expands and the number of submissions rises, securing enough qualified reviewers and performing a quality check of the received reviews becomes increasingly difficult.

The review process included a rebuttal phase where authors were given a week to address any concerns raised by the reviewers. Each response was limited to a single PDF page using a predefined template. Rebuttals with formatting or anonymity violations were removed by the Program Chairs. ACs then facilitated discussions with reviewers to evaluate the merits of each submission. While the goal was to reach a consensus, the final decision was left to the ACs in the triplet to ensure fairness. As is common, the ACs within each triplet were tasked with organizing meetings to discuss papers, especially borderline cases. The entire process was conducted online, with no in-person meetings. For each paper, a primary and a secondary AC were assigned from the AC triplet. The primary AC was responsible for entering the decisions and meta-reviews into the CMT system, which were then reviewed and approved by the secondary ACs. Besides making accept/reject decisions, the ACs were also required to provide recommendations regarding oral/poster presentations and award nominations.

Reviewers and ACs were asked to flag papers with potential ethical concerns. An ethics review committee, consisting of four leading researchers from both industry and academia, was appointed. The role of the committee was to evaluate papers escalated by the AC for further investigation. In total, 31 papers were flagged to the committee. The committee requested that the authors of 15 papers address specific concerns in their camera-ready submissions. The authors all complied with the requests.

Following the recent IEEE decision to ban the use of the photograph of Lena Forsén, we automatically identified all submissions that included this image. Although the ban did not officially apply to ECCV, we nevertheless shared the context with the authors of the affected papers and suggested that they remove the photograph in the camera-ready submission. All authors complied.

To help the ACs to track the overall progress of the reviewing and decision-making processes, we created AC triplet activity dashboards for each AC triplet (via Google sheets), which were updated on a regular basis. During the review process, the activity dashboard enabled ACs to track the progress of incoming reviews for each paper and promptly identify papers that necessitated attention, e.g., papers potentially requiring the invitation of emergency reviewers, or papers with suspiciously short reviews. During the decision-making process, the dashboard was also used by the Program Chairs to identify papers with incomplete review information, e.g., for which reviewers did not enter their final recommendation, the AC did not enter their final decision or meta-review, the secondary AC did not confirm the recommendation, etc.

At the end of the decision-making progress, the Program Chairs carefully examined the very small number of cases where ACs overruled a unanimous reviewer recommendation for acceptance or rejection. In such instances, it was crucial for the primary and secondary ACs to explain in detail the motivation behind the decision. To ensure a fair evaluation, an additional Area Chair, not originally part of the decision-making triplet, was appointed to provide an independent opinion and advise the Program Chairs in determining whether to uphold or reverse the overturn.

Inevitably, after decisions were released, some authors were dissatisfied. The authors were given the possibility to appeal the final decision in case of procedural issues encountered during the review process by filling out a form to directly contact the Program Chairs. Valid reasons for appealing a decision included policy errors (e.g., reviewers or ACs enforcing a non-existent policy), clerical errors (e.g., the meta-review clearly indicated an intention to accept a paper, but it was mistakenly rejected), and significant misunderstandings by the reviewers or ACs. In total, we received 59 appeals and upheld 6 of them. One was due to a clear policy error by an AC, and the paper decision changed from reject to accept following a successful appeal. The other 5 papers were already conditionally accepted – but the AC/reviewers wrongly flagged them as papers for which the associated dataset was required to be released prior to final acceptance; this condition was removed following the appeal.

After the camera-ready deadline, additional checks were conducted, particularly for papers contributing a new dataset and for those flagged for potential ethical concerns. Specifically, during the submission phase, authors were required to specify if the primary contribution of their papers was the release of a new dataset, and for these papers, a valid dataset link had to be provided by the camera-ready deadline. A total of 435 submissions included a dataset as part of their contribution, and all complied with the requirement to release the dataset by the camera-ready deadline.

We sincerely thank all the ACs and reviewers for their invaluable contributions to the review process. Their dedication and expertise were crucial in maintaining the high standards of the conference. We would also like to thank our Technical Program Chair, Sascha Hornauer, who did tremendous work behind the scenes, and to the entire CMT team for their prompt support with any technical issues.

October 2024

Aleš Leonardis
Elisa Ricci
Stefan Roth
Olga Russakovsky
Torsten Sattler
Gül Varol

Contents – Part XLI

SkateFormer: Skeletal-Temporal Transformer for Human Action Recognition

Jeonghyeok Do and Munchurl Kim(✉)

Korea Advanced Institute of Science and Technology, Daejeon, South Korea
{ehwjdgur0913,mkimee}@kaist.ac.kr
https://kaist-viclab.github.io/SkateFormer_site/

Abstract. Skeleton-based action recognition, which classifies human actions based on the coordinates of joints and their connectivity within skeleton data, is widely utilized in various scenarios. While Graph Convolutional Networks (GCNs) have been proposed for skeleton data represented as graphs, they suffer from limited receptive fields constrained by joint connectivity. To address this limitation, recent advancements have introduced transformer-based methods. However, capturing correlations between all joints in all frames requires substantial memory resources. To alleviate this, we propose a novel approach called Skeletal-Temporal Transformer (SkateFormer) that partitions joints and frames based on different types of skeletal-temporal relation (Skate-Type) and performs skeletal-temporal self-attention (Skate-MSA) within each partition. We categorize the key skeletal-temporal relations for action recognition into a total of four distinct types. These types combine (i) two skeletal relation types based on physically neighboring and distant joints, and (ii) two temporal relation types based on neighboring and distant frames. Through this partition-specific attention strategy, our SkateFormer can selectively focus on key joints and frames crucial for action recognition in an action-adaptive manner with efficient computation. Extensive experiments on various benchmark datasets validate that our SkateFormer outperforms recent state-of-the-art methods.

Keywords: Skeleton-based Action Recognition · Transformer · Partition-Specific Attention

1 Introduction

In recent years, human action recognition (HAR) [39,51,57] has gained widespread applications in real-life scenarios, involving the classification of actions based on human movements. A diverse range of data sources such as videos captured from

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-72940-9_23.

A. Leonardis et al. (Eds.): ECCV 2024, LNCS 15099, pp. 401–420, 2025.
https://doi.org/10.1007/978-3-031-72940-9_23

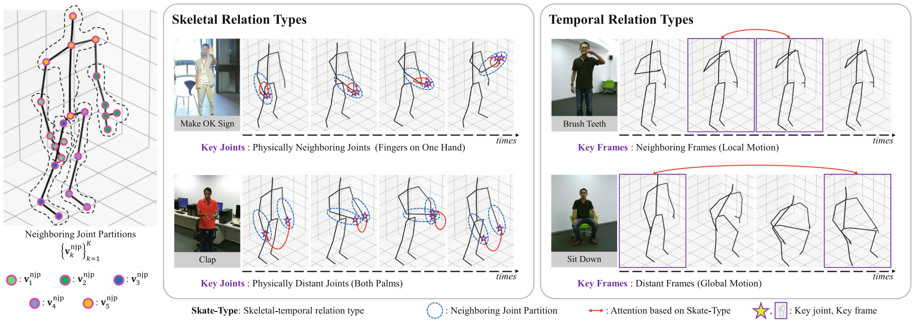

Fig. 1. SkateFormer's partition-specific attention strategy for skeleton-based action recognition, leveraging two skeletal relation types – physically neighboring and distant joints – and two temporal relation types for local and global motion.

RGB cameras, optical flow generated through post-processing, 2D/3D skeletons estimated from RGB videos [2,50], and skeletons acquired from sensors [16], contain information about human movements that can be leveraged for action recognition. However, using RGB videos as input is challenging. They are sensitive to external factors such as lighting conditions, camera distance and angles, and background variations [12]. Moreover, RGB videos require large storage capacity due to their detailed background visuals. Conversely, 3D skeletons obtained through sensors can offer a compact representation with robustness to external environmental changes, and do not require additional post-processing modules such as pose and optical flow estimations [2,15,49,50].

The joints and their connections (or bones) in skeleton input correspond to the vertices and edges, respectively, in a graph structure. Consequently, many methods based on Graph Convolutional Networks (GCNs) [18] suitable for processing graph inputs have been extensively proposed for human action recognition. Most of the GCN-based methods [4–6,10,19–22,24,36,43,46,47,61,63,67,68,75] exchange information between different joints within a single frame using graph convolutions, and capture the temporal dynamics for each joint using 1D temporal convolutions. However, they struggle to capture the relation of physically distant joints (e.g., between two hands with open arms) effectively due to the direct propagation of information between physically connected joints.

To mitigate this limitation, transformer-based methods utilizing self-attention to capture relations across all joint pairs have been proposed [40,58,60,64]. Yet, considering every joint in every frame is inefficient as specific joints in certain frames are more critical for particular action recognition. Some transformer-based methods have tried to reduce computational complexity by squeezing features along the joint or frame dimension before self-attention [11], only using either skeletal or temporal relations [13,38,44,73,76], or by tokenizing physically similar skeletal information [34,62]. Separating joints and frames during self-attention can obscure the critical skeletal-temporal relations for action recognition, and tokenization may dilute vital information, potentially degrading performance.

To overcome these issues, we propose an efficient transformer-based approach, called Skeletal-Temporal Transformer (SkateFormer) that introduces joint and frame partition strategies and partition-specific self-attention based on types of skeletal-temporal relations (Skate-Type). Figure 1 illustrates the joint and frame partition strategies and partition-specific self-attention of our SkateFormer. For example, in the 'make ok sign' action class, relations between physically neighboring joints (e.g., joints on the same hand) are crucial, whereas in the 'clap' action class, relations between physically distant joints (e.g., between the palms of both hands) are more critical. Regarding temporal relations, for repetitive local motions like in the 'brush teeth' class, neighboring frame relations are essential, whereas for actions with global motion like 'sit down', distant frame relations become critical. Additionally, the speed at which actions are performed can vary significantly depending on the actor. To make our SkateFormer computationally efficient, we introduce a novel partition-specific attention (Skate-MSA). For this, we divide the skeletal-temporal relations into four partition types: (i) neighboring joints and local motion – Skate-Type-1, (ii) distant joints and local motion – Skate-Type-2, (iii) neighboring joints and global motion – Skate-Type-3 and (iv) distant joints and global motion – Skate-Type-4. Therefore, Skate-MSA is designed to efficiently capture skeletal-temporal relations at the joint-element-level, eliminating the need for tokenization, which typically involves joint-group-level attention [34,62]. In summary, our contributions are as follows:

- We propose a Skeletal-Temporal Transformer (SkateFormer), a partition-specific attention strategy (Skate-MSA) for skeleton-based action recognition that captures skeletal-temporal relations and reduces computational complexity.
- We introduce a range of augmentation techniques and an effective positional embedding method, named Skate-Embedding, which combines skeletal and temporal features. This method significantly enhances action recognition performance by forming an outer product between learnable skeletal features and fixed temporal index features.
- Our SkateFormer sets a new state-of-the-art for action recognition performance across multiple modalities (4-ensemble condition) and single modalities (joint, bone, joint motion, bone motion), showing notable improvement over the most recent state-of-the-art methods. Additionally, it concurrently establishes a new state-of-the-art in interaction recognition, a sub-field of action recognition.

2 Related Works

2.1 RNN/CNN-Based Approaches

Recently, skeleton-based action recognition has made significant progress. Early research efforts focused on utilizing Recurrent Neural Networks (RNNs), including LSTM and GRU, to handle skeleton data due to its sequential and continuous nature [25,28,29,37,45,71,77]. Furthermore, some studies explored the conversion of skeleton data into pseudo-images to leverage Convolutional Neural Networks (CNNs) [8,17,23,59,65,66]. Very recently, PoseC3D [12] employed a 2D pose estimation network [50] on RGB video to generate image-like skeleton heatmaps as input for CNNs, and Ske2Grid [1] proposed a method to transform 2D/3D skeleton data nodes into image-format grid patches.

2.2 GCN-Based Approaches

Skeleton data is composed of joints and bones, which correspond to vertices and edges in a graph. Consequently, research in this domain has predominantly gravitated towards Graph Neural Networks (GNNs) [42,69] and Graph Convolutional Networks (GCNs) [3–6,10,19–21,24,36,43,46,47,61,67,68]. InfoGCN [7] introduced additional loss terms to form a compact latent feature space, resulting in clear decision boundaries. Similarly, FR-Head [75] incorporated contrastive learning to separate feature representations of ambiguous classes. LST [63] introduced language supervision to train GCNs effectively, while HD-GCN [22] introduced rooted trees as a data structure to create diverse connections between joints. Despite their merits, GCNs have limitations in effectively capturing topology variations depending on the input data by using the fixed-sized kernels.

2.3 Transformer-Based Approaches

Recent research explores transformer-based approaches for skeleton-based action recognition, which excel in capturing data-adaptive joint connections [40,58,60,64]. However, these methods often require high computational costs due to the usage of their large-sized attention maps. Early methods have tried to reduce attention map sizes by employing feature pooling [11] or reshaping [44] techniques. Some studies have attempted to mitigate computational load by utilizing separate skeletal and temporal attention modules in a parallel [38,73] or serial [13,44,76] configurations. Nevertheless, they face challenges in simultaneously capturing skeletal-temporal relations, which are critical aspects of skeleton-based action recognition. IGFormer [34] and ISTA-Net [62] tackle the challenge by embedding joint sets within the same partition into a unified token before attention modules, relying solely on partition strategies for tokenization without introducing partition-specific attention mechanisms. Despite incorporating skeletal-temporal attention at the joint-group-level, they encounter limitations in local action recognition due to the tokenization process leading to the loss of physically similar skeletal information. In our work, the proposed SkateFormer employs partition-specific skeletal-temporal attention modules to effectively capture skeletal-temporal relations at the joint-element-level without tokenization.

Complexity of Transformer-based Vision Tasks. Vision Transformers (ViT) [9] revolutionized the field of computer vision but face challenges due to high computational costs. To mitigate these challenges, various studies have been conducted to reduce the size of attention maps while efficiently propagating local and global information [30,54]. In the context of vision tasks, 2D feature maps are organized along horizontal and vertical spatial axes. In contrast, our task involves skeletal and temporal axes, making the nature of our data fundamentally different. However, the elements within skeleton data tend to scale mostly with the factors such as the number of recognized joints, the number of individuals performing actions, and the number of frames, necessitating the reduction of computational complexity. Inspired by the works [53,54], we propose a transformer structure that efficiently captures skeletal-temporal relations at the joint-element-level, reducing the computational cost.

3 Methodology

3.1 Overview of SkateFormer

The overall architecture of our proposed SkateFormer is depicted in Fig. 2. A skeleton sequence, corresponding to a single action, is sampled to maintain a consistent frame count of T, resulting in $\mathbf{X} \in \mathbb{R}^{T \times V \times M \times C_{\text{in}}}$, where V represents the number of joints per frame, M denotes the number of individuals involved in the action, and C_{in} is the dimensionality of data representing a single joint ($C_{\text{in}} = 3$ for 3D skeletons in our cases). Initially, we reshape $\mathbf{X}$ from (T, V, M, C_{in}) to $(T, V{\cdot}M, C_{\text{in}})$ to treat joints from different individuals separately. For simplicity, we redefine V as $V{\cdot}M$ so that we have $\mathbf{X} \in \mathbb{R}^{T \times V \times C_{\text{in}}}$ in subsequent text. Next, the three linear layers in the SkateFormer map the low-dimensional raw skeleton data into a higher-dimensional feature space. We then perform the skeletal-temporal positional embedding by adding learnable skeletal features and fixed temporal index features to this mapped feature. The embedded features pass through R SkateFormer blocks, each comprising a self-attention layer that propagates features via skeletal-temporal relations, followed by a feed-forward layer that refines the features. The final features, after R SkateFormer Blocks, undergo skeletal-temporal pooling to produce the outcome $\hat{\mathbf{y}} \in \mathbb{R}^{N_c}$, which is trained via backpropagation through a loss function by comparing $\hat{\mathbf{y}}$ with the true label $\mathbf{y}$, where N_c represents the number of classes.

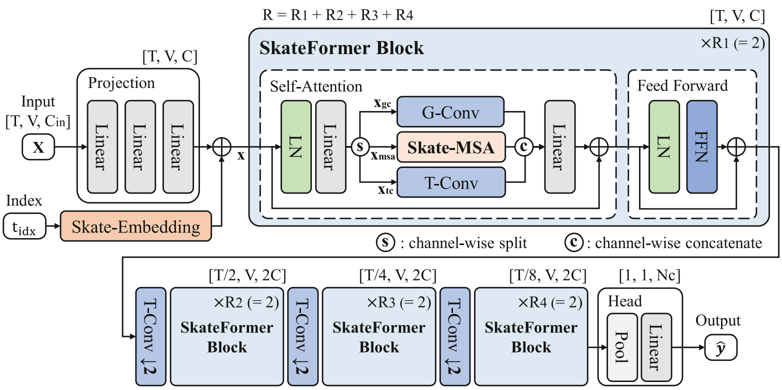

Fig. 2. The overall framework of our proposed SkateFormer.

3.2 SkateFormer Block

Each of the SkateFormer Blocks in Fig. 2 is structured similarly as the traditional transformer blocks [30,54], incorporating both a self-attention layer and a feed-forward network (FFN). The self-attention component of the SkateFormer Block can be expressed as follows:

$$\begin{aligned} &[\mathbf{x}_{\mathsf{gc}}, \mathbf{x}_{\mathsf{tc}}, \mathbf{x}_{\mathsf{msa}}] = \mathsf{Split}(\mathsf{Linear}(\mathsf{LN}(\mathbf{x}))) \\ &\mathbf{x}_{\mathsf{gc}} \leftarrow \mathsf{G\text{-}Conv}(\mathbf{x}_{\mathsf{gc}}) \\ &\mathbf{x}_{\mathsf{tc}} \leftarrow \mathsf{T\text{-}Conv}(\mathbf{x}_{\mathsf{tc}}) \\ &\mathbf{x}_{\mathsf{msa}} \leftarrow \mathsf{Skate\text{-}MSA}(\mathbf{x}_{\mathsf{msa}}) \\ &\mathbf{x} \leftarrow \mathbf{x} + \mathsf{Linear}(\mathsf{Concat}(\mathbf{x}_{\mathsf{gc}}, \mathbf{x}_{\mathsf{tc}}, \mathbf{x}_{\mathsf{msa}})), \end{aligned} \tag{1}$$

where the LN represents the Layer Normalization layer, the Linear denotes a fully-connected layer, and the Split indicates a channel splitting function. When input $\mathbf{x}$ to the self-attention component has C channels, we split $\mathbf{x}$ into $\mathbf{x}_{\mathsf{gc}}$ of $C/4$ channels, $\mathbf{x}_{\mathsf{tc}}$ of $C/4$ channels, and $\mathbf{x}_{\mathsf{msa}}$ of $C/2$ channels. To obtain inductive biases for both skeletal and temporal aspects, we utilize a one-layer GCN ($\mathsf{G\text{-}Conv}$) and a single temporal convolution layer ($\mathsf{T\text{-}Conv}$), respectively. With a total of H heads, we employ a learnable matrix of shape $(H/4, V, V)$ instead of a predefined adjacency matrix to perform $\mathsf{G\text{-}Conv}$ operations, enabling us to capture diverse connectivity patterns between joints. $\mathsf{T\text{-}Conv}$ is a 1D convolution layer with a kernel size of k, and performs 1D ($H/4$-group) convolutions for $\mathbf{x}_{\mathsf{tc}}$ to capture temporal dynamics. The FFN of the SkateFormer Block is expressed as follows: $\mathbf{x} \leftarrow \mathbf{x} + \mathsf{Linear}(\mathsf{Act}(\mathsf{Linear}(\mathsf{LN}(\mathbf{x}))))$, where Act represents the activation layer. When downsampling is needed for the input to a SkateFormer Block, a 1D convolution with a stride of 2 and Batch Normalization layer are applied.

3.3 Skate-MSA

As shown in Fig. 3, the feature map input $\mathbf{x}_{\mathsf{msa}}$ to the Skate-MSA is first split channel-wise into four equal-sized features $\mathbf{x}^1_{\mathsf{msa}}, \mathbf{x}^2_{\mathsf{msa}}, \mathbf{x}^3_{\mathsf{msa}}, \mathbf{x}^4_{\mathsf{msa}}$, each of which has $C/8$ channels. For each $\mathbf{x}^i_{\mathsf{msa}}$, self-attention operation is applied to discern the correlations between joints corresponding to specific relation types as:

$$\begin{aligned} &\mathbf{x}^i_{\mathsf{msa}} \leftarrow \boldsymbol{\mathcal{R}}_i(\mathsf{MSA}(\boldsymbol{\mathcal{P}}_i(\mathbf{x}^i_{\mathsf{msa}}))) \\ &\mathbf{x}_{\mathsf{msa}} \leftarrow \mathsf{Concat}(\mathbf{x}^1_{\mathsf{msa}}, \mathbf{x}^2_{\mathsf{msa}}, \mathbf{x}^3_{\mathsf{msa}}, \mathbf{x}^4_{\mathsf{msa}}), \end{aligned} \tag{2}$$

where $\boldsymbol{\mathcal{P}}_i$ and $\boldsymbol{\mathcal{R}}_i$ represent the i-th Skate-Type partition and reverse operations, respectively. This approach enables the Skate-MSA to effectively analyze and model various joint relations in a specialized manner, contributing to its overall performance improvement.

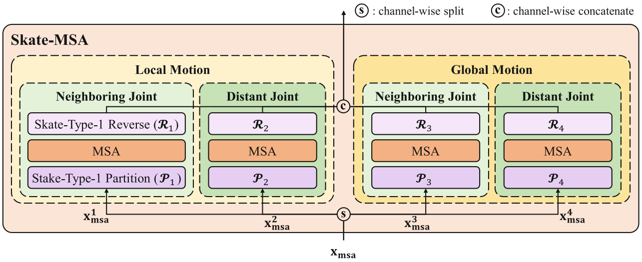

Fig. 3. The Skate-MSA of our proposed SkateFormer.

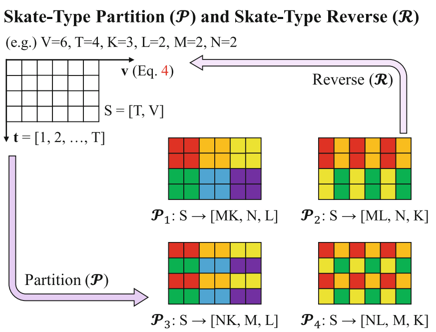

Fig. 4. Skate-Type partition and reverse.

Partition and Reverse. Let a frame be $\mathbf{f}^t \in \mathbb{R}^{V \times C_{\text{in}}}$ at time t. So, we have $\mathbf{X} = \{\mathbf{f}^t\}_{t=1}^{T} \in \mathbb{R}^{T \times V \times C_{\text{in}}}$ where $\mathbf{f}^t$ consists of the 3D ($C_{\text{in}} = 3$) positions of V joints at time t. The frames of $\mathbf{X}$ are often temporally highly correlated for most of the action types. However, the joint positions of skeletons tend to be less correlated among some joints, depending on different action types. Nevertheless, some other joints such as the joints of the head, neck, shoulders, abdomen and pelvises are relatively not highly variant among themselves for various action types. That is, they may move together in a group. So, it is efficient to partition the whole set of joints into smaller-sized joint partitions that can be useful to distinguish different actions. Furthermore, it is also worthwhile to consider categorizing the movements of various actions into two partitions, such as local and global motions: local motion, which only changes the positions of a small number of joints with partial movements such as teeth brushing and clapping, and global motion, which appears in the entire frames and includes actions like sitting down and standing up. Motivated from this, we partition the action into four skeletal-temporal relation types. For this, we first partition the entire set of joints into total non-overlapping K subsets as neighboring joint partitions $\mathbf{v}_k^{\text{njp}}$. For example, when $K = 5$, we may have $\mathbf{v}_1^{\text{njp}}$, $\mathbf{v}_2^{\text{njp}}$, $\mathbf{v}_3^{\text{njp}}$, $\mathbf{v}_4^{\text{njp}}$ and $\mathbf{v}_5^{\text{njp}}$ as the right arm, left arm, right leg, left leg and torso, respectively. The elements within each subset are ordered in a manner that extends outward from the body's central region (e.g., for $k = 3$, $v_{3,1}$, $v_{3,2}$ and $v_{3,3}$ are the 3D coordinates of the pelvis, right knee and right foot positions, respectively):

$$\mathbf{v}_k^{\text{njp}} = [v_{k,1}, v_{k,2}, ..., v_{k,L}], \tag{3}$$

where $k = 1, 2, ..., K$, L represents the total number of its elements, and $\mathbf{v}_i^{\mathsf{njp}} \cap \mathbf{v}_j^{\mathsf{njp}} = \varnothing$ for $i \neq j$. We stack $\mathbf{v}_k^{\mathsf{njp}}$'s, thus creating the skeletal axis as:

$$\mathbf{v} = \left[\mathbf{v}_1^{\mathsf{njp}} | \mathbf{v}_2^{\mathsf{njp}} | \cdots | \mathbf{v}_K^{\mathsf{njp}}\right], \tag{4}$$

where $|\mathbf{v}| = KL = V$. Also, to consider the relations between the joints that can be physically separated in a distance each other, we group the same-positioned elements of $\mathbf{v}_k^{\mathsf{njp}}$ to create possibly distant joint partitions $\mathbf{v}_l^{\mathsf{djp}}$ as:

$$\mathbf{v}_l^{\mathsf{djp}} = [v_{1,l}, v_{2,l}, ..., v_{K,l}], \tag{5}$$

where $l = 1, 2, ..., L$. Similarly, we define $\mathbf{t} = [1, 2, ..., T]$ as a time axis. We further define two semantic time axes as $\mathbf{t}_m^{\mathsf{local}}$ for local motion comprehension and $\mathbf{t}_n^{\mathsf{global}}$ for global motion understanding as:

$$\begin{aligned} \mathbf{t}_m^{\mathsf{local}} &= [(m-1)N + 1, (m-1)N + 2, ..., mN] \\ \mathbf{t}_n^{\mathsf{global}} &= [n, n + N, ..., n + (M-1)N], \end{aligned} \tag{6}$$

where $m = 1, 2, ..., M$ with the total M number of $\mathbf{t}_n^{\mathsf{global}}$ elements, and $n = 1, 2, ..., N$ with the total N number of $\mathbf{t}_m^{\mathsf{local}}$ elements. Note that $\mathbf{t}_m^{\mathsf{local}}$ is a segment of consecutive time indices to represent a local motion within the time segment while $\mathbf{t}_n^{\mathsf{global}}$ is an N-strided sparse time axis to capture the global motion over $\mathbf{t}$. So, we have $\mathbf{t} = \{\mathbf{t}_m^{\mathsf{local}}\}_{m=1}^{M} = \{\mathbf{t}_n^{\mathsf{global}}\}_{n=1}^{N}$ and $|\mathbf{t}| = MN = T$. Based on these time axes, our skeletal-temporal partitions of joints and frames are explained in the followings.

We partition the joints and frames together into four types in the context of skeletal-temporal relation for the Skate-MSA – Skate-Type-1, -2, -3 and -4: (i) The Skate-Type-1 partition, denoted as $\boldsymbol{\mathcal{P}}_1$, pertains to a self-attention branch targeting neighboring joints and local motion, based on $\mathbf{v}_k^{\mathsf{njp}}$ and $\mathbf{t}_m^{\mathsf{local}}$; (ii) the Skate-Type-2 partition ($\boldsymbol{\mathcal{P}}_2$) represents a branch for distant joints and local motion, based on $\mathbf{v}_l^{\mathsf{djp}}$ and $\mathbf{t}_m^{\mathsf{local}}$; (iii) the Skate-Type-3 partition ($\boldsymbol{\mathcal{P}}_3$) signifies a branch for neighboring joints and global motion, based on $\mathbf{v}_k^{\mathsf{njp}}$ and $\mathbf{t}_n^{\mathsf{global}}$; (iv) Lastly, the Skate-Type-4 partition ($\boldsymbol{\mathcal{P}}_4$) corresponds to a branch targeting distant joints and global motion, based on $\mathbf{v}_l^{\mathsf{djp}}$ and $\mathbf{t}_n^{\mathsf{global}}$. The Skate-Type partition operations transform the shape ($S = (T, V, c)$) of $\mathbf{x}_{\mathsf{msa}}^i$ into:

$$\begin{aligned} \boldsymbol{\mathcal{P}}_1 &: S \rightarrow (MK, N, L, c) \quad & \boldsymbol{\mathcal{P}}_2 &: S \rightarrow (ML, N, K, c) \\ \boldsymbol{\mathcal{P}}_3 &: S \rightarrow (NK, M, L, c) \quad & \boldsymbol{\mathcal{P}}_4 &: S \rightarrow (NL, M, K, c), \end{aligned} \tag{7}$$

where $c = C/8$. The partitioned feature map $\mathbf{x}_{\mathsf{msa}}^{i,\boldsymbol{\mathcal{P}}} = \boldsymbol{\mathcal{P}}_i(\mathbf{x}_{\mathsf{msa}}^i)$ undergoes multi-head self-attention (MSA) and is then reshaped back to its original size of (T, V, c) through a Skate-Type reverse operation $\boldsymbol{\mathcal{R}}_i$ according to Eq. 2.

Multi-head self-attention. In order to collectively consider the skeletal-temporal relation types, we can generalize feature maps $\mathbf{x}_{\mathsf{msa}}^{i,\boldsymbol{\mathcal{P}}}$ to have a shape of (B, T', V', c). The feature maps are first reshaped into $(B, T' \cdot V', c)$. Through linear mappings of $\mathbf{x}_{\mathsf{msa}}^{i,\boldsymbol{\mathcal{P}}}$ of shape (c, c) by $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$, we obtain the query ($\mathbf{Q}$), key ($\mathbf{K}$), and value ($\mathbf{V}$) tensors as $[\mathbf{Q}, \mathbf{K}, \mathbf{V}] = \mathbf{x}_{\mathsf{msa}}^{i,\boldsymbol{\mathcal{P}}} \cdot [\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V]$.

Since we utilized half of the total H heads for $\mathsf{G\text{-}Conv}$ and $\mathsf{T\text{-}Conv}$ operations, the remaining $H/2$ heads are divided into quarters with $H' = H/8$. They are then assigned differently with the four Skate-Types, each of which has a MSA. We reshaped $\mathbf{Q}$, $\mathbf{K}$, $\mathbf{V}$ to have a shape of $(B, H', T' \cdot V', c/H')$. This refers to the self-attention (SA) for each individual head h, given as:

$$\begin{aligned} \mathsf{SA}_h(\mathbf{x}_{\mathsf{msa}}^{i,\boldsymbol{\mathcal{P}}}) &= \mathsf{SoftMax}(\mathbf{Q}_h \mathbf{K}_h^{\mathsf{T}} / \sqrt{c/H'} + \mathbf{B}_h)\mathbf{V}_h \\ \mathsf{MSA}(\mathbf{x}_{\mathsf{msa}}^{i,\boldsymbol{\mathcal{P}}}) &= \mathsf{Concat}(\mathsf{SA}_1(\mathbf{x}_{\mathsf{msa}}^{i,\boldsymbol{\mathcal{P}}}), ..., \mathsf{SA}_{H'}(\mathbf{x}_{\mathsf{msa}}^{i,\boldsymbol{\mathcal{P}}})), \end{aligned} \tag{8}$$

where $\mathbf{Q}_h \mathbf{K}_h^{\mathsf{T}}$ is a data-dependent term that varies based on the input, and $\mathbf{B}_h$ represents the skeletal-temporal positional bias.

Complexity analysis. The computational complexity of a naive self-attention layer [9,55] for feature map $\mathbf{x}_{\mathsf{msa}}$ with a shape of $(T, V, C/2)$ is $2(VT)^2(C/2)$. In contrast, the computation complexity of our Skate-MSA is:

$$2(VT)^2 \frac{C}{2} \left[\frac{1}{4} \left(\frac{1}{MK} + \frac{1}{ML} + \frac{1}{NK} + \frac{1}{NL} \right) \right]. \tag{9}$$

In our experimental settings, this results in approximately a $48\times$ reduction in computational complexity compared to the naive self-attention layer.

3.4 Skeletal-Temporal Positional Embedding

Our novel skeletal-temporal positional embedding method, called Skate-Embedding, is tightly associated with temporal data augmentation. So, we first explain data augmentation with our contribution and then describe the Skate-Embedding.

Intra-instance augmentation. Previous skeleton-based action recognition methods often used data augmentation by temporally sampling input frames with fixed strides or randomly for the whole input, which we refer to as *temporal augmentation*. Also, as *skeletal augmentation*, various transformations such as the actor order permutation, random shear, random rotation, random scaling, random coordinate drop, and random joint dropout have been applied [14,26].

We propose trimmed-uniform random sampling of frames with p portion. This sampling cuts out the first and last parts of the total input sequence and performs uniform random sampling of frames. From our trimmed-uniform random sampling of frames with a p portion, the masking effect of skeleton sequences in their front and back portions, as well as a more dense sampling effect in the middle, are expected. This can lead to a concentration in the middle portion, resulting in better data augmentation.

Inter-instance augmentation. We define inter-instance augmentation as data augmentation by exchanging the bone lengths of different subjects across different frame sequences (not within each frame sequence). By doing so, the resulting data augmentation can provide the diversity of subjects with different body sizes, which can help generalization learning.

Skate-Embedding. We propose a novel skeletal-temporal positional embedding method, called Skate-Embedding, that utilizes fixed (not learnable) temporal index features and learnable (not fixed) skeletal features. The temporal index features are suitable for conveying to the first SkateFormer Block the temporal positional information of the sampled frames from sequences of various lengths. The sampled temporal indices are designated as $t_{\mathsf{idx}} = [t_1, t_2, ..., t_T]$. These temporal indices are then normalized to the range $[-1, 1]$, and are used for the fixed temporal index features as done in the temporal positional embedding. The fixed temporal index features, denoted as $\mathsf{TE} \in \mathbb{R}^{T \times C}$, are constructed for t_{idx} by using the sinusoidal positional embedding [55]. On the other hand, as skeletal joint positional (which are not the 3D coordinates of joints but their indices) embeddings, the learnable skeletal features, denoted as $\mathsf{SE} \in \mathbb{R}^{V \times C}$, are learned within the Skate-Embedding as shown in Fig. 2. Finally, the skeletal-temporal positional embedding $\mathsf{STE} \in \mathbb{R}^{T \times V \times C}$ is done by taking the outer product of SE and TE as $\mathsf{STE}[i, j, d] = \mathsf{SE}[j, d] \cdot \mathsf{TE}[i, d]$ at the i-th time, the j-th joint and the d-th channel.

4 Experimental Results

4.1 Datasets

NTU RGB+D. This dataset [41] offers 60 action classes and includes diverse activities like drinking water, eating, brushing teeth, dropping objects, and more. It comprises total 56,880 videos captured from 40 subjects across 155 camera viewpoints. The dataset utilizes Kinect v2, encompassing RGB, IR, depth, and 3D skeleton data, and supports cross-subject (*X-Sub60*) and cross-view (*X-View60*) evaluation. Out of the 60 action classes, only 11 are related to human interaction, specifically when two individuals are present. We denote this subset as *NTU-Inter* [11,34].

NTU RGB+D 120. This dataset [27] extends the NTU RGB+D dataset [41] with 120 action classes, covering actions such as putting on headphones, basketball shooting, juggling table tennis balls, and more. It contains 114,480 videos from 106 subjects across 155 camera viewpoints. It enables cross-subject (*X-Sub120*) and cross-setup (*X-Set120*) evaluation based on different subject and camera setups. Out of the 120 action classes, only 26 are related to human interaction. We denote this subset as *NTU-Inter 120* [11,34,62].

NW-UCLA. This dataset [56] includes 10 action classes, such as picking up objects, dropping trash, walking, sitting, standing, and more. It comprises total 1,475 videos from 10 subjects across three camera views, collected using Kinect v1 [74] and featuring RGB, IR, depth, and 3D skeleton data. Evaluation follows the cross-view approach, utilizing two training views and one test view.

4.2 Experiment Details

All experiments were performed using the PyTorch framework [35], running on a single NVIDIA DGX A100 GPU. Each model was trained with a total of 500 epochs. We employed a linear warm-up strategy for the learning rate, gradually increasing it from 10^{-7} to 10^{-3} during the first 25 epochs. Subsequently, a cosine-annealing scheduler [31] was used to update the learning rate at each iteration for the remaining epochs. We employed the AdamW optimizer [32] with a betas of (0.9, 0.999), a weight decay of 5×10^{-4}. Additionally, gradient clipping [70] was applied for the loss values with gradients exceeding 1. The batch size was set to 128, the random seed was fixed at 1, and we adopted the label-smoothed cross-entropy loss [33,52] as our loss function with parameter $\alpha = 0.1$.

From various experiments, we found the empirical values as follows. For our experiments with the NTU RGB+D and NTU RGB+D 120 datasets, we used the following configuration: $V = 48$ (excluding the center of the body, resulting in 24 joints for each of the two individuals), $T = 64$, $L = 4$, $K = 12$, $M = 8$, $N = 8$, and $C = 96$. For the experiments with the NW-UCLA dataset, $V = 20$, $T = 64$, $L = 4$, $K = 5$, $M = 8$, $N = 8$, and $C = 96$.

Table 1. Top-1 accuracy of different skeleton-based action recognition methods on the NTU RGB+D, NTU RGB+D 120 and NW-UCLA datasets. The methods of RNN, CNN, GCN, and Transformer types are evaluated based on the number of input frames and the use of ensemble strategies. The **best performances** are highlighted in **bold**.

Types	Methods	Frames	NTU RGB+D (%)						NTU RGB+D 120 (%)						NW-UCLA (%)
			X-Sub60			X-View60			X-Sub120			X-Set120			
			$\mathbb{E}_1$	$\mathbb{E}_2$	$\mathbb{E}_4$	$\mathbb{E}_1$	$\mathbb{E}_2$	$\mathbb{E}_4$	$\mathbb{E}_1$	$\mathbb{E}_2$	$\mathbb{E}_4$	$\mathbb{E}_1$	$\mathbb{E}_2$	$\mathbb{E}_4$	
RNN	AGC-LSTM [45]	100	87.5	89.2	-	93.5	95.0	-	-	-	-	-	-	-	93.3
CNN	TA-CNN [65]	64	88.8	-	90.4	93.6	-	94.8	82.4	-	85.4	84.0	-	86.8	96.1
	Ske2Grid [1]	100	88.3	-	-	95.7	-	-	82.7	-	-	85.1	-	-	-
GCN	SGN [72]	20	-	89.0	-	-	94.5	-	-	79.2	-	-	81.5	-	-
	CTR-GCN [3]	64	89.9	-	92.4	-	-	96.8	84.9	88.7	88.9	-	90.1	90.6	96.5
	ST-GCN++ [10]	100	89.3	91.4	92.1	95.6	96.7	97.0	83.2	87.0	87.5	85.6	87.5	89.8	-
	InfoGCN [7]	64	-	-	92.7	-	-	96.9	85.1	88.5	89.4	86.3	89.7	90.7	96.6
	FR-Head [75]	64	90.3	92.3	92.8	95.3	96.4	96.8	85.5	-	89.5	87.3	-	90.9	96.8
	Koopman [61]	64	90.2	-	92.9	95.2	-	96.8	85.7	-	**90.0**	87.4	-	91.3	97.0
	LST [63]	64	90.2	-	92.9	95.6	-	97.0	85.5	-	89.9	87.0	-	91.1	97.2
	HD-GCN [22]	64	90.6	92.4	93.0	95.7	96.6	97.0	85.7	89.1	89.8	87.3	90.6	91.2	96.9
	STC-Net [21]	64	-	92.5	93.0	-	96.7	97.1	-	89.3	89.9	-	90.7	91.3	97.2
Trans-former	DSTA-Net [44]	128	-	-	91.5	-	-	96.4	-	-	86.6	-	-	89.0	-
	STST [73]	128	-	-	91.9	-	-	96.8	-	-	-	-	-	-	-
	FG-STFormer [13]	128	-	-	92.6	-	-	96.7	-	-	89.0	-	-	90.6	97.0
	Hyperformer [76]	64	90.7	-	92.9	95.1	-	96.5	86.6	-	89.9	88.0	-	91.3	96.7
	SkateFormer	64	**92.6**	**93.0**	**93.5**	**97.0**	**97.4**	**97.8**	**87.7**	**89.4**	89.8	**89.3**	**91.0**	**91.4**	**98.3**

4.3 Performance Comparison

We present a comprehensive performance comparison for our SkateFormer against recent state-of-the-art skeleton-based action recognition methods. The performance comparison is made for three ensembles of different modalities ($\mathbb{E}_1$,

$\mathbb{E}_2$, $\mathbb{E}_4$): (i) $\mathbb{E}_1$ - *joint* modality only; (ii) $\mathbb{E}_2$ - *joint* + *bone* modalities; and (iii) $\mathbb{E}_4$ - *joint* + *bone* + *joint motion* + *bone motion* modalities. As in [3,7,75,76], we train separate networks for each modality and ensemble their outputs. Table 1 shows the overall performance comparisons of various methods for the NTU RGB+D, NTU RGB+D 120 and NW-UCLA datasets under the three ensembles. Notably, several works address human interaction recognition [11,34,62], a sub-part of skeleton-based action recognition, specifically focusing on scenarios where two or more individuals coexist within a single action. Accordingly, we additionally present the performance of human interaction recognition methods on the NTU Inter and NTU Inter 120 datasets in Table 2.

$\mathbb{E}_1$, $\mathbb{E}_2$, $\mathbb{E}_4$ **performance.** As shown in Tables 1, 2, the extensive experiments demonstrate that our SkateFormer outperforms all the state-of-the-art (SoTA) methods except the $\mathbb{E}_4$ of X-Sub120. It should be noted that the performance of our SkateFormer is relatively higher for $\mathbb{E}_1$ and $\mathbb{E}_2$ than $\mathbb{E}_4$. This is indicative of the model's efficient attention strategy, which is based on partition-specific processing, allowing for the simultaneous handling of joint (input), bone (skeletal relations), joint motion (temporal relations), and bone motion (skeletal-temporal relations) inputs within a singular network framework. As the ensemble modality increases, the potential for information redundancy also rises, which may diminish the ensemble synergy. Therefore, our SkateFormer is already somewhat a strong learner in the form of a single modality compared to others.

Ensemble methods improve performance, but their effectiveness depends on the computational complexity, scaling proportionally with the number of models in the ensemble. Also, optimizing ensemble coefficients is dataset and model-specific [22], posing challenges in real-world applications. Consequently, it is crucial to leverage diverse modalities inherently within a single model to achieve both generalization and efficiency. Therefore, highlighting the importance of $\mathbb{E}_1$ performance is essential as a key indicator of the model's ability to generalize across diverse inputs.

Table 2. Comparison with human interaction recognition methods on the NTU-Inter and NTU-Inter 120 datasets. The **best performances** are highlighted in **bold**.

Types	Methods	NTU-Inter ($\mathbb{E}_1$, %)		NTU-Inter 120 ($\mathbb{E}_1$, %)		Params. (M)	FLOPs (G)	Time (ms)
		X-Sub60	X-View60	X-Sub120	X-Set120			
Transformer	IGFormer [34]	93.6	96.5	85.4	86.5	-	-	-
	SkeleTR [11]	94.9	97.7	87.8	88.3	3.82	7.30	-
	ISTA-Net [62]	-	-	90.6	91.7	6.22	68.18	21.71
	SkateFormer	**97.1**	**99.3**	**92.3**	**93.2**	**2.02**	**3.62**	**11.25**

Computational Complexity Analysis. Table 3 shows the complexity comparisons [48] for various skeleton-based action recognition methods. Our SkateFormer exhibits a competitive balance between model complexity and computational efficiency, compared to other methods. It maintains a comparable number of parameters and FLOPs with the GCN-based methods, while substantially

Table 3. Comparative analysis of SkateFormer with other methods by parameters, FLOPs, inference time, and average top-1 accuracy for joint modality.

Types	Methods	Params. (M)	FLOPs (G)	Time (ms)	NTU RGB+D ($\mathbb{E}_1$, %)	NTU RGB+D 120 ($\mathbb{E}_1$, %)
GCN	InfoGCN [7]	1.56	**3.34**	12.97	-	85.7
	FR-Head [75]	**1.45**	3.60	18.49	92.8	86.4
	Koopman [61]	5.38	8.76	17.86	92.7	86.6
	LST [63]	2.10	3.60	18.85	92.9	86.3
	HD-GCN [22]	1.66	3.44	72.81	93.2	86.5
Transformer	DSTA-Net [44]	3.45	16.18	13.80	-	-
	Hyperformer [76]	2.71	9.64	18.07	92.9	87.3
	SkateFormer	2.03	3.62	**11.46**	**94.8**	**88.5**

reducing these metrics in comparison to the transformer-based counterparts. Table 3 has been compiled using publicly available official code.

4.4 Ablation Studies

To see the efficacy of the key components in our SkateFormer, the ablation experiments were carried for the NTU RGB+D dataset. A detailed analysis on the effectiveness of the components is provided in Tables 4, 5, 6 to quantify the impact of different design choices on the model's performance in terms of accuracy (%) across two benchmarks: X-Sub60 and X-View60.

Skate-Types of Skate-MSA. Table 4 presents the impact of different Skate-Types on accuracy. As shown, incorporating the skeletal relation types only improves the action classification performance over the baseline, and so does the temporal relation types only. The full model that utilizes the Skate-Types (skeletal-temporal relation types) achieves the highest accuracy, indicating that both skeletal and temporal splits are crucial in distinguishing complex actions.

Table 4. Ablation study on the influence of skeletal and temporal relation types (Skate-Types) of Skate-MSA in our SkateFormer. (*: The batch size is set to 8 due to limited GPU memory.)

Attention Types	NTU RGB+D (%)		Params. (M)	FLOPs (G)	Time (ms)	Memory (MB)
	X-Sub60	X-View60				
Baseline (no attention)	90.7	95.7	2.03	3.59	10.68	80.6
+ Naive self-attention*	90.6 (↓0.1)	95.0 (↓0.7)	2.03	4.39	17.55	1612.1
$+ \mathbf{v}_k^{\mathsf{njp}} + \mathbf{v}_l^{\mathsf{djp}}$	91.8 (↑1.1)	96.4 (↑0.7)	2.03	3.59	11.10	86.9
$+ \mathbf{t}_m^{\mathsf{local}} + \mathbf{t}_n^{\mathsf{global}}$	91.9 (↑1.2)	96.6 (↑0.9)	2.03	3.59	11.09	85.5
$+ \mathbf{v}_k^{\mathsf{njp}} + \mathbf{v}_l^{\mathsf{djp}} + \mathbf{t}_m^{\mathsf{local}} + \mathbf{t}_n^{\mathsf{global}}$	**92.6** (↑**1.9**)	**97.0** (↑**1.3**)	2.03	3.62	11.46	137.6

Table 5. Ablation study results on the Skate-Embedding to assess skeletal and temporal embedding impacts.

Embedding Methods		NTU RGB+D (%)	
Skeletal	Temporal	X-Sub60	X-View60
✗	✗	91.9 (↓0.7)	96.6 (↓0.4)
	Learnable	91.8 (↓0.8)	96.5 (↓0.5)
	Fixed (TE)	91.6 (↓1.0)	96.6 (↓0.4)
Learnable (SE)	✗	92.1 (↓0.5)	96.5 (↓0.5)
	Learnable	91.4 (↓1.2)	96.2 (↓0.8)
	Fixed (TE)	**92.6**	**97.0**

Table 6. Ablation study results on the effect of intra-instance and inter-instance augmentations.

Intra-instance		Inter-instance	NTU RGB+D (%)	
Temporal	Skeletal		X-Sub60	X-View60
Trimmed			89.8	94.6
Trimmed		✓	90.3 (↑0.5)	94.3 (↓0.3)
Trimmed	✓		92.2 (↑2.4)	96.6 (↑2.0)
Fixed	✓	✓	91.4 (↑1.6)	96.4 (↑1.8)
Uniform	✓	✓	91.6 (↑1.8)	96.8 (↑2.2)
Trimmed	✓	✓	**92.6 (↑2.8)**	**97.0 (↑2.4)**

Exploration of Skate-Embedding. Table 5 shows the ablation study on the STE to assess the impact of different skeletal and temporal embedding methods. We explore various combinations of skeletal and temporal embedding methods. The learnable skeletal embedding (SE) paired with fixed temporal embedding (TE) achieves superior performance, suggesting an optimal balance between adaptability and stability in embeddings. The Skate-Embedding allows for a more tailored feature representation, leading to improved recognition accuracy.

Evaluation of frame sampling methods. In Table 6, we compare three frame sampling strategies with (i) fix-strided (*Fixed*), (ii) uniform random (*Uniform*) and (iii) our trimmed-uniform random (*Trimmed*) sampling methods. Due to the masking effect of skeleton sequences in their front and the back portions as well and more dense sampling effect in the middles during training, the trimmed-uniform random sampling is more effective in perspectives of generalization learning and data augmentation. It surpasses traditional fix-strided and uniform random sampling approaches with 1.2 (0.6)% and 1.0 (0.2)% margins for the X-Sub60 (X-View60), respectively.

Impact of data augmentations. Table 6 presents the effectiveness of intra-instance (traditional) and inter-instance (additional) data augmentations. The intra-instance augmentation alone improves the performance while performance drop is observed with the inter-instance augmentation alone for the X-View60. This performance drop with 0.3%-point in accuracy is due to the inherent characteristics of the cross-view setting such that the variability in bone lengths across different subjects within the X-View60 is limited. This implies that the model may already encapsulate a comprehensive representation of these attributes. Consequently, the introduction of inter-instance data augmentation does not contribute additional discriminative information and may instead introduce redundancy, leading to a performance decline. However, the amalgamation of both intra- and inter-instance augmentations provides a synergistic enhancement, especially by random scaling and shearing on bone lengths in the skeletal augmentation, achieving peak accuracy of 92.6% on X-Sub60 and 97.0% on X-View60.

5 Conclusions

In this paper, we presented SkateFormer – a novel skeletal-temporal transformer tailored for action recognition tasks. For this, we propose an effective partition-specific attention strategy in skeleton-based action recognition, both to capture essential features and to reduce computational complexity. For efficient training, our novel Skate-Embedding that combines skeletal and temporal features is presented, significantly enhancing action recognition performance by forming an outer product between learnable skeletal features and fixed temporal index features. Our SkateFormer sets a new state-of-the-art for action recognition performance across multiple modalities (4-ensemble condition) and single modalities (joints, bones, joint motions, bone motions), showing notable improvement over the most recent state-of-the-art methods.

Acknowledgements. This work was supported by IITP grant funded by the Korea government (MSIT) (No. RS2022-00144444, Deep Learning Based Visual Representational Learning and Rendering of Static and Dynamic Scenes).

References

1. Cai, D., Kang, Y., Yao, A., Chen, Y.: Ske2Grid: skeleton-to-grid representation learning for action recognition. In: International Conference on Machine Learning (2023)
2. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7291–7299 (2017)
3. 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)
4. Chen, Z., Li, S., Yang, B., Li, Q., Liu, H.: Multi-scale spatial temporal graph convolutional network for skeleton-based action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 1113–1122 (2021)
5. Cheng, K., Zhang, Y., Cao, C., Shi, L., Cheng, J., Lu, H.: Decoupling GCN with DropGraph module for skeleton-based action recognition. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12369, pp. 536–553. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58586-0_32
6. Cheng, K., Zhang, Y., He, X., Chen, W., Cheng, J., Lu, H.: Skeleton-based action recognition with shift graph convolutional network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 183–192 (2020)
7. Chi, H.G., Ha, M.H., Chi, S., Lee, S.W., Huang, Q., Ramani, K.: InfoGCN: representation learning for human skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20186–20196 (2022)
8. Ding, Z., Wang, P., Ogunbona, P.O., Li, W.: Investigation of different skeleton features for CNN-based 3D action recognition. In: 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), pp. 617–622. IEEE (2017)

9. Dosovitskiy, A., et al.: An image is worth 16×16 words: transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
10. Duan, H., Wang, J., Chen, K., Lin, D.: PYSKL: towards good practices for skeleton action recognition. In: Proceedings of the 30th ACM International Conference on Multimedia, pp. 7351–7354 (2022)
11. Duan, H., et al.: SkeleTR: towards skeleton-based action recognition in the wild. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 13634–13644 (2023)
12. Duan, H., Zhao, Y., Chen, K., Lin, D., Dai, B.: Revisiting skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2969–2978 (2022)
13. Gao, Z., et al.: Focal and global spatial-temporal transformer for skeleton-based action recognition. In: Proceedings of the Asian Conference on Computer Vision, pp. 382–398 (2022)
14. Guo, T., Liu, H., Chen, Z., Liu, M., Wang, T., Ding, R.: Contrastive learning from extremely augmented skeleton sequences for self-supervised action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 762–770 (2022)
15. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: FlowNet 2.0: evolution of optical flow estimation with deep networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2462–2470 (2017)
16. Kay, W., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
17. Ke, Q., Bennamoun, M., An, S., Sohel, F., Boussaid, F.: Learning clip representations for skeleton-based 3D action recognition. IEEE Trans. Image Process. **27**(6), 2842–2855 (2018)
18. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
19. Korban, M., Li, X.: DDGCN: a dynamic directed graph convolutional network for action recognition. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12365, pp. 761–776. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58565-5_45
20. Kwon, T., Tekin, B., Stühmer, J., Bogo, F., Pollefeys, M.: H2O: two hands manipulating objects for first person interaction recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10138–10148 (2021)
21. Lee, J., Lee, M., Cho, S., Woo, S., Jang, S., Lee, S.: Leveraging spatio-temporal dependency for skeleton-based action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10255–10264 (2023)
22. 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)
23. Li, B., Dai, Y., Cheng, X., Chen, H., Lin, Y., He, M.: Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep CNN. In: 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), pp. 601–604. IEEE (2017)
24. 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)

25. Li, S., Li, W., Cook, C., Zhu, C., Gao, Y.: Independently recurrent neural network (IndRNN): Building a longer and deeper RNN. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5457–5466 (2018)
26. Lin, L., Zhang, J., Liu, J.: Actionlet-dependent contrastive learning for unsupervised skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2363–2372 (2023)
27. 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 Trans. Pattern Anal. Mach. Intell. **42**(10), 2684–2701 (2019)
28. Liu, J., Shahroudy, A., Xu, D., Wang, G.: Spatio-temporal LSTM with trust gates for 3D human action recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 816–833. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_50
29. Liu, J., Wang, G., Duan, L.Y., Abdiyeva, K., Kot, A.C.: Skeleton-based human action recognition with global context-aware attention LSTM networks. IEEE Trans. Image Process. **27**(4), 1586–1599 (2017)
30. Liu, Z., et al.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022 (2021)
31. Loshchilov, I., Hutter, F.: SGDR: stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (2016)
32. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
33. Müller, R., Kornblith, S., Hinton, G.E.: When does label smoothing help? In: Advances in Neural Information Processing Systems, vol. 32 (2019)
34. Pang, Y., Ke, Q., Rahmani, H., Bailey, J., Liu, J.: IGFormer: interaction graph transformer for skeleton-based human interaction recognition. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXV, pp. 605–622. Springer Nature Switzerland, Cham (2022). https://doi.org/10.1007/978-3-031-19806-9_35
35. Paszke, A., et al.: Automatic differentiation in PyTorch (2017)
36. Peng, W., Hong, X., Chen, H., Zhao, G.: Learning graph convolutional network for skeleton-based human action recognition by neural searching. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 2669–2676 (2020)
37. Perez, M., Liu, J., Kot, A.C.: Interaction relational network for mutual action recognition. IEEE Trans. Multimedia **24**, 366–376 (2021)
38. Plizzari, C., Cannici, M., Matteucci, M.: Skeleton-based action recognition via spatial and temporal transformer networks. Comput. Vis. Image Underst. **208**, 103219 (2021)
39. Presti, L.L., La Cascia, M.: 3D skeleton-based human action classification: a survey. Pattern Recogn. **53**, 130–147 (2016)
40. Qiu, H., Hou, B., Ren, B., Zhang, X.: Spatio-temporal segments attention for skeleton-based action recognition. Neurocomputing **518**, 30–38 (2023)
41. 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)
42. Shi, L., Zhang, Y., Cheng, J., Lu, H.: Skeleton-based action recognition with directed graph neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7912–7921 (2019)

43. Shi, L., Zhang, Y., Cheng, J., Lu, H.: Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12026–12035 (2019)
44. Shi, L., Zhang, Y., Cheng, J., Lu, H.: Decoupled spatial-temporal attention network for skeleton-based action-gesture recognition. In: Proceedings of the Asian Conference on Computer Vision (2020)
45. Si, C., Chen, W., Wang, W., Wang, L., Tan, T.: An attention enhanced graph convolutional LSTM network for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1227–1236 (2019)
46. 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)
47. Song, Y.F., Zhang, Z., Shan, C., Wang, L.: Constructing stronger and faster baselines for skeleton-based action recognition. IEEE Trans. Pattern Anal. Mach. Intell. **45**(2), 1474–1488 (2022)
48. Sovrasov, V.: ptflops: a flops counting tool for neural networks in PyTorch framework (2018–2023). https://github.com/sovrasov/flops-counter.pytorch
49. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: PWC-NET: CNNs for optical flow using pyramid, warping, and cost volume. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8934–8943 (2018)
50. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5693–5703 (2019)
51. Sun, Z., Ke, Q., Rahmani, H., Bennamoun, M., Wang, G., Liu, J.: Human action recognition from various data modalities: a review. IEEE Trans. Pattern Anal. Mach. Intell. (2022)
52. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
53. Tu, Z., Talebi, H., Zhang, H., Yang, F., Milanfar, P., Bovik, A., Li, Y.: MAXIM: multi-axis MLP for image processing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5769–5780 (2022)
54. Tu, Z., et al.: MaxViT: multi-axis vision transformer. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIV, pp. 459–479. Springer Nature Switzerland, Cham (2022). https://doi.org/10.1007/978-3-031-20053-3_27
55. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
56. 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)
57. Wang, L., Huynh, D.Q., Koniusz, P.: A comparative review of recent kinect-based action recognition algorithms. IEEE Trans. Image Process. **29**, 15–28 (2019)
58. Wang, L., Koniusz, P.: 3Mformer: multi-order multi-mode transformer for skeletal action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5620–5631 (2023)
59. Wang, P., Li, Z., Hou, Y., Li, W.: Action recognition based on joint trajectory maps using convolutional neural networks. In: Proceedings of the 24th ACM International Conference on Multimedia, pp. 102–106 (2016)

60. Wang, Q., Shi, S., He, J., Peng, J., Liu, T., Weng, R.: IIP-transformer: intra-inter-part transformer for skeleton-based action recognition. In: 2023 IEEE International Conference on Big Data (BigData), pp. 936–945. IEEE (2023)
61. Wang, X., Xu, X., Mu, Y.: Neural koopman pooling: control-inspired temporal dynamics encoding for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10597–10607 (2023)
62. Wen, Y., Tang, Z., Pang, Y., Ding, B., Liu, M.: Interactive spatiotemporal token attention network for skeleton-based general interactive action recognition. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7886–7892. IEEE (2023)
63. Xiang, W., Li, C., Zhou, Y., Wang, B., Zhang, L.: Generative action description prompts for skeleton-based action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10276–10285 (2023)
64. Xin, W., Miao, Q., Liu, Y., Liu, R., Pun, C.M., Shi, C.: Skeleton MixFormer: multivariate topology representation for skeleton-based action recognition. In: Proceedings of the 31st ACM International Conference on Multimedia, pp. 2211–2220 (2023)
65. 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)
66. Xu, Y., Cheng, J., Wang, L., Xia, H., Liu, F., Tao, D.: Ensemble one-dimensional convolution neural networks for skeleton-based action recognition. IEEE Signal Process. Lett. **25**(7), 1044–1048 (2018)
67. 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)
68. Ye, F., Pu, S., Zhong, Q., Li, C., Xie, D., Tang, H.: Dynamic GCN: context-enriched topology learning for skeleton-based action recognition. In: Proceedings of the 28th ACM International Conference on Multimedia, pp. 55–63 (2020)
69. Zeng, A., Sun, X., Yang, L., Zhao, N., Liu, M., Xu, Q.: Learning skeletal graph neural networks for hard 3D pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 11436–11445 (2021)
70. Zhang, J., He, T., Sra, S., Jadbabaie, A.: Why gradient clipping accelerates training: a theoretical justification for adaptivity. arXiv preprint arXiv:1905.11881 (2019)
71. Zhang, P., Lan, C., Xing, J., Zeng, W., Xue, J., Zheng, N.: View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2117–2126 (2017)
72. Zhang, P., Lan, C., Zeng, W., Xing, J., Xue, J., Zheng, N.: Semantics-guided neural networks for efficient skeleton-based human action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1112–1121 (2020)
73. Zhang, Y., Wu, B., Li, W., Duan, L., Gan, C.: STST: spatial-temporal specialized transformer for skeleton-based action recognition. In: Proceedings of the 29th ACM International Conference on Multimedia, pp. 3229–3237 (2021)
74. Zhang, Z.: Microsoft kinect sensor and its effect. IEEE Multimedia **19**(2), 4–10 (2012)

75. 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)
76. Zhou, Y., Li, C., Cheng, Z.Q., Geng, Y., Xie, X., Keuper, M.: Hypergraph transformer for skeleton-based action recognition. arXiv preprint arXiv:2211.09590 (2022)
77. Zhu, W., et al.: Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 30 (2016)

Author Index

A. Leonardis et al. (Eds.): ECCV 2024, LNCS 15099, pp. 499–501, 2025.
https://doi.org/10.1007/978-3-031-72940-9

Z