Learning from Skeleton Data for Human Action Recognition

Tailin Chen

School of Computing

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy of the Newcastle University

2023

Abstract

Human action recognition is a fundamental task in human-centred scene understanding and has achieved much progress in computer vision and multimedia. Recently, skeleton-based action recognition has become much more popular with the help of inexpensive motion sensors and effective pose estimation algorithms. As the skeleton data typically are light-weight, view-invariant and privacy-friendly to its video counterparts, a wide range of applications, such as human-computer interactions and healthcare assistance, can therefore benefit from these features. As human body skeleton is naturally constructed as a spatial-temporal graph instead of a sequence of vector or pseudo-image, where the topology information is more informative for action recognition, the recent graph convolutional neural networks (GCNs) are then proposed for learning representations on such graph structured data and have dominated skeleton-based action recognition tasks. In this thesis we focus on human action recognition from skeleton data using GCN-based methods.

In my first work, a novel dual-head GCN model is proposed aims to jointly capture fine-grained and coarse-grained motion patterns efficiently. In this dual-head network, each head focuses on specific granularity of temporal motions and hence is more effective. In my second work, a novel long short-term feature aggregation strategy is proposed to model the varied spatial-temporal dependencies, which is also a key to recognise human actions in skeleton sequences. This novel factorised architecture can alternately perform spatial and temporal feature aggregation. The aforementioned two works focus on the problem of many-shot classification, where each class has a substantial amount of samples during training. Nevertheless, the acquisition of well-annotated skeletal sequences is labour-intensive and time-consuming. In my third work, to alleviate the data collection burden, a part-aware prototypical representation learning strategy is proposed for one-shot skeleton-based action recognition. This novel part-aware model captures skeleton motion patterns at global and part levels which is rarely investigated. Extensive experiments are conducted on public datasets and models achieve the state-of-the-art performance on all of the corresponding benchmarks.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

Human action recognition is one of the fundamental tasks in human-centred scene understanding and has achieved much progress in computer vision due to its wide range of applications, such as security surveillance, human-computer interactions, robotics, healthcare assistance and so on.

Previously, most of the works in this domain focus on using RGB videos as input [28, 62, 70, 6, 69] for action recognition. However, when using RGB images as input, it is difficult to achieve an efficient and effective human action recognition system. First, the cluttered backgrounds and huge variations of appearance, viewpoint angle, and illumination are very challenging factors for recognition. Second, RGB videos contain millions of pixel values, which requires a significant amount of computing resources to process. Finally, the 3D body-structured data of the observed action, which is not provided in the 2D image-based input, is vital important for reliable action recognition.

To address the aforementioned problems, many recent works [16, 38, 49, 22]

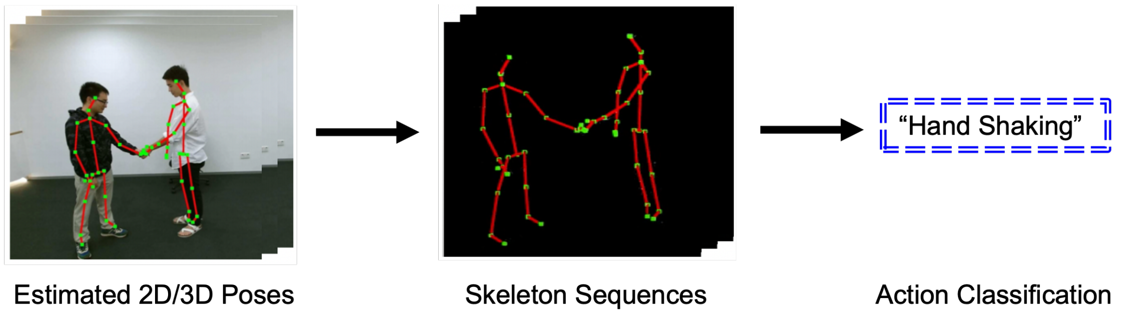


Figure 1.1: The task of skeleton-based action recognition aims to classify human actions from skeleton sequences only. Examples of the images and skeletons from the NTU RGB+D dataset [53]

began to investigate human action recognition based on a high-level compact representation: 2D/3D skeleton data, i.e., recognising actions by using the 2D/3D coordinates of the major body joints in each frame as input, as illustrated in Fig. 1.1.

In contrast to other modalities of data, such as RGB sequence and Optical Flow, skeleton sequences encode the trajectories of human body joints, which characterise informative human motions and are typically more compact and robust to environment conditions than its video counterpart. Therefore, skeleton data is also a suitable modality for human action recognition.

Besides, with the help of effective pose estimation algorithms [68, 5, 67, 21] and inexpensive depth sensors [92], such as the Microsoft Kinect [92], the skeleton data can be easily and effectively acquired. As a result, action recognition with 3D skeleton data is further accelerated and becomes more and more popular in recent years.

1.2 Challenges and Motivations

In this thesis, we mainly focus on the task of 3D skeleton-based human action recognition. We adopt GCN-based methods and investigate the challenges both in

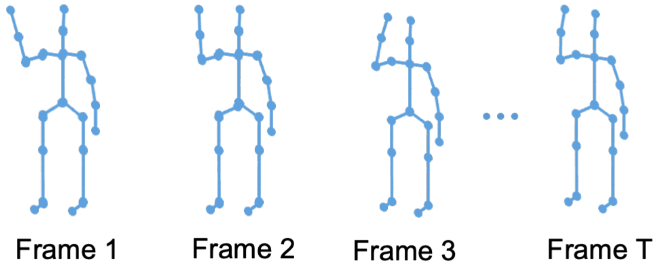


Figure 1.2: The skeleton sequence of action ”hand waving”

fully-supervised learning and few-shot learning settings. Below we introduce some of the important aspects that need to be considered when handling this task and the motivations of the proposed methods in this thesis.

1.2.1 Spatio-Temporal Dependency Modelling

As illustrated in Fig.1.2, in the sequence of 3D skeleton data, there are strong dependency relations among different frames, which reveal the temporal dynamics of the human motions. Besides, in each single frame, there is also high dependence among different body joints, which shows the spatial posture pattern. Intuitively, to understand the human behaviours in the skeleton sequence well, the analysis of the dependency information in both spatial and temporal dimensions is vitally important. Specifically, the challenges can be summarised as follows:

Inherent Dynamics and Complexity: Within the sequences of 3D skeleton data, each frame representing a human posture bears complex dynamic changes in relation to its adjacent frames. These changes not only reflect the continuity of movements but also embody the complexity of human actions. For instance, a simple gesture such as waving involves coordinated movements of multiple joints at different points in time, where the inter-dependencies among these movements are crucial for accurately recognising the action.

Long-term Dependency Problem: Identifying certain actions requires understanding not just the short-term dynamics but also capturing the long-term dependencies within action sequences. This is particularly important for comprehending complex action sequences, such as dances or athletic movements, where the significance of the actions unfolds over a longer duration.

This motivates us to develop methods to model the dependency of dynamic joints in both the spatial and temporal dimensions for human action recognition under different settings (Chapter 3, Chapter 4 and Chapter 5). The uniqueness of this work can be summarised as follows:

Innovative Model Architecture: Our research distinguishes itself from previous works through the adoption of a dual-head network structure that learns both coarse-grained and fine-grained action patterns simultaneously. This structure more effectively captures the spatio-temporal dependencies of actions because it allows the model to understand actions at different levels of granularity, thereby enhancing recognition accuracy and robustness.

Application of Cross-head Attention Mechanism: An important improvement over existing methods in our work is the design of a cross-head attention mechanism to mutually enhance features at different levels of granularity. This mechanism enables the model to focus more on key action information while leveraging contextual information from both spatial and temporal dimensions, further improving the performance of action recognition.

Long-short Term Feature Aggregation Strategy: We also propose a long-short term feature aggregation strategy to better model the long-term dependencies in action sequences. By combining action information across different temporal scales, this strategy allows the model to understand the overall structure of actions while retaining clarity in the details of key movements, which is especially crucial when dealing with complex action sequences.

1.2.2 Discover Informative Joints and Frames

Intuitively, not all joints or frames in the skeleton sequence are informative for recognising some actions. For example, the movements of the foot joints are very important for the action “kicking”, while the hand joints’ motions are irrelevant. Skeleton sequences of different actions often have different informative joints and frames. Besides, in the same sequence, the informativeness degree of a joint may also vary across different frames. Specifically, the challenges can be summarised as follows:

Diversity and Complexity of Actions: A core challenge in action recognition lies in understanding and leveraging the significance of information from different joints and frames within an action. Different actions may depend on different parts of the body, for example, jumping actions may rely more on the movement of leg joints, while clapping actions mainly depend on hand joints. Distinguishing which joints and frames are crucial for a specific action and which are secondary or negligible is vital for enhancing recognition accuracy.

Issue of Redundancy and Distraction: Not all joints or frames in skeletal action data carry useful information for action identification. Movements of certain joints or frames might be irrelevant to the action being recognised, or could even introduce noise, thereby obstructing the accuracy of action recognition.

Motivated by this observation, we investigate to discover the informative joints or frames of the skeleton sequence and emphasise their features, and meanwhile suppress the features of the irrelevant joints or frames, because the irrelevant joints and frames often contribute negatively for action recognition, and may bring in noise that corrupts the performance (Chapter 3 and Chapter 5). The uniqueness of this work can be summarised as follows:

Adaptive Mechanism for Assessing the Importance of Joints and

Frames: Diverging from traditional action recognition methodologies, we propose an adaptive mechanism for evaluating and harnessing the significance of joints and frames. Our approach, by analysing dynamic patterns in skeletal data, autonomously identifies joints and frames that contribute most to recognising specific actions, thereby optimising the recognition process and enhancing accuracy.

Feature Enhancement via Attention Mechanisms: We employ attention mechanisms to highlight features of informative joints and frames while suppressing those that are unimportant or distracting. This method allows for more precise focus on features that are decisive for action recognition, improving both the efficiency and accuracy of identification.

Fusion of Multi-granular Information: Our work goes beyond focusing on the information from individual joints or frames; it also considers how to effectively merge information from different joints and frames to capture both global and local characteristics of actions. This strategy of fusing multi-granular information enables our method to comprehend actions more comprehensively, maintaining high accuracy even in cases of subtle or complex movements.

1.2.3 Multi-Granular Action Representation Learning

As shown in Fig.1.3, different from RGB videos, skeleton data contains only the 2D or 3D coordinates of human joints, which makes fine-grained actions particularly challenging due to the lack of image-based contextual information. In order to capture discriminative spatial structure and temporal motion patterns, existing methods usually rely on a unified spatio-temporal representation of the skeleton data at the original frame rate of input sequence. While such a strategy may have the capacity to capture action patterns of multiple scales, they often suffer from inaccurate predictions on fine-grained action classes due to model complexity and

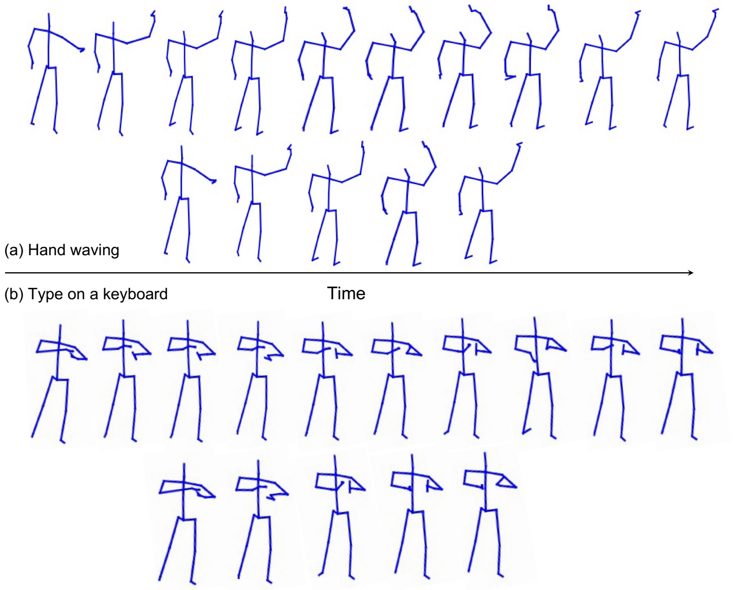


Figure 1.3: Some actions can be recognised via coarse-grained temporal motion patterns, such as 'hand waving' (top), while recognising other actions requires not only coarse-grained motion but subtle temporal movements, such as 'type on a keyboard' (bottom).

limited training data. Specifically, the challenges can be summarised as follows:

Granularity Variations in Actions: One of the pivotal challenges in action recognition is effectively capturing and distinguishing actions across different granularities. For instance, coarse-grained actions, such as walking, differ significantly from fine-grained actions, like writing, not only in the amplitude of movements but also in the subtlety of action features. Developing a model capable of representing these varied granularities within a unified framework to achieve high-accuracy recognition presents a significant challenge.

Recognition Complexity of Fine-grained Actions: The recognition of fine-grained actions is notably more challenging due to their minimal movement amplitude and subtle feature characteristics. Existing action recognition methodologies often fall short in accurately identifying fine-grained actions, struggling to capture sufficient detail or effectively differentiate between similar fine-grained movements.

This motivates us to develop an approach to efficiently capture actions of multiple scales (Chapter 3). The uniqueness of this work can be summarised as follows:

Innovative Multi-Granularity Learning Framework: We propose an innovative multi-granularity learning framework, specifically designed to capture action features from coarse to fine detail. Through a hierarchical network structure, this framework processes action information at different granularity levels concurrently, thereby enhancing the ability to recognize and differentiate actions across granularities effectively.

Enhancement Techniques for Fine-grained Features: Addressing the challenge of recognizing fine-grained actions, we have developed a set of fine-grained feature enhancement techniques. Utilizing advanced attention mechanisms and deep feature learning strategies, these techniques significantly improve the model's capability to capture subtle action details, thereby increasing the accuracy of fine-grained action recognition.

Dynamic Weight Adjustment Mechanism: Our framework includes a dynamic weight adjustment mechanism that adaptively modulates the focus on features of different granularities based on the complexity and detail of the action. This mechanism allows the model to automatically enhance the learning of critical features when dealing with complex actions or those with subtle details, ensuring efficient and accurate action recognition.

1.2.4 Learning Meta Knowledge for One-Shot Fine-Grained Skeleton-based Action Recognition

Humans are very good at learning new concept with very little supervision. For example, a child can recognise the sport "basketball" and "football" from very few pictures in a book. However, current deep learning models still requires a large amount of examples for learning such concept. Inspired by human's fast learning ability, such "few-shot" or "one-shot" learning setting, which consists of learning a class from very few labelled examples, attracts much interests in recent years.

In real-world deployment, action recognition in certain scenarios (e.g. education, sports, healthcare) are highly demanded. However, only scarce data and annotations are available for the novel action classes that are unseen in the public datasets. The problem of one shot skeleton-based action recognition poses unique challenges in learning transferable representation and is of practical importance. Existing meta-learning frameworks typically rely on the body-level representations in spatial dimension, which limits the generalisation to capture subtle visual differences in the fine-grained label space. Specifically, the challenges can be summarised as follows:

Learning Difficulty with Limited Data: A primary challenge in one-shot learning for action recognition is developing an effective recognition model with extremely limited data availability (only one example). This challenge is particularly pronounced in fine-grained action recognition, where actions encompass more detailed and subtle differences. The model needs to capture these nuanced features from a single example and accurately classify them, demanding a sophisticated understanding and internalisation of action characteristics.

Complexity in Meta-knowledge Learning: In the one-shot learning paradigm, learning effective meta-knowledge is crucial for enhancing the model's generalisation ability across unseen actions. The complexity of meta-knowledge learning lies in enabling the model to comprehend and internalise action features extracted from limited samples and apply this knowledge to new, unseen actions effectively.

Motivated by such observations, we investigate to design a method to capture skeleton motion patterns globally as well as locally for transferable knowledge learning under one-shot skeleton-based action recognition setting (Chapter 5). The uniqueness of this work can be summarised as follows:

Innovative Meta-learning Framework: We introduce a novel meta-learning framework specifically designed for one-shot fine-grained action recognition. Unlike previous works, our framework integrates advanced feature extraction and attention mechanisms, enabling the model to learn rich action representations from a single sample and effectively apply this meta-knowledge to recognise new actions.

Efficient Feature Internalisation and Transfer Strategy: We have developed a set of strategies for efficient feature internalisation and transfer, allowing the model not only to capture the subtle differences in fine-grained actions but also to quickly transfer the knowledge learned from one action to another. This approach significantly enhances the model's accuracy and generalisation capability in fine-grained action recognition.

Multi-level Understanding of Action Representations: Our work emphasises a multi-level understanding of action representations by the model, from macro-level action features to micro-level action details. This comprehensive understanding strategy is key to differentiating our method from previous research. It ensures that, even with minimal data, the model can accurately recognise fine-grained actions.

1.3 Proposed Action Recognition Models

In this thesis, we present three models for skeleton-based action recognition, all of them are based on Graph Convolutional Networks (GCNs) and each of which addresses one or several important aspects that are discussed in Sec.1.2. Below we briefly introduce these three network models.

The first model (Chapter 3) extends the original holistic GCN-based model to a dual-head structure to learn multi-granular spatio-temporal graph representations, which has strong capability in modelling coarse- and fine-grained motion patterns simultaneously. Besides, a cross-head attention mechanism is designed to mutually enhance the feature at two different levels of granularity, which enables the model to selectively focus on key motion information with the assistance of contextual information both from spatial and temporal dimensions. This model yields state-of-the-art performance on three challenging datasets for 3D action recognition.

The second model (Chapter 4) is developed to capture both short- and long-range joint dependency in spatial domain, short-and long-term joints dynamics in temporal domain for skeleton-based action recognition. In this model, a multi-scale decentralised spatial aggregation network is used to model the dependency of distant joints, an attention-enhanced temporal pyramid aggregation network is used to model long-range temporal dynamics. Furthermore, a maximum response attention module is used for performance improvement. Our model, despite smaller model size, achieves higher or comparable results on three public datasets.

The third model (Chapter 5) is designed for one-shot skeleton-based action recognition which aims to alleviate the burden of annotation cost and is of practical importance for some rare real-world recognition scenarios, such as healthcare and physical education. To this end, a novel part-aware prototypical representation learning framework is proposed to classify skeleton sequences from only one labelled sample per class. This model first captures motion patterns at body-level and then attends to part-level for part-aware prototypes generation via meta-learning framework. Besides, we also devise a novel class-agnostic attention fusion mechanism which can highlight the importance of parts for each action class based on an contrastive learning manner. Our model achieves the new state-of-the-art on two public datasets under one-shot learning setting.

1.4 Contributions

This thesis introduces significant advancements in the field of human action recognition using 3D skeleton data under different settings, e.g., fully-supervised learning (Chapter 3, Chapter 4) and few-shot learning (Chapter 5). Our research not only addresses existing challenges but also proposes novel frameworks and methodologies, setting new benchmarks across various datasets. Herein, we articulate our pivotal contributions:

(1) Innovative Dual-head Network Architecture via Graph Convolutional Networks (GCNs): A pivotal achievement of our research is the formulation of the DualHead-Net, expounded in Chapter 3, an inventive architecture engineered to concurrently process multi-granularity spatio-temporal representations of human actions. This model uniquely integrates a cross-head attention mechanism to synergistically refine features across both macro and micro perspectives, significantly enhancing the model's capacity to discern spatiotemporal patterns within action sequences. Demonstrating unparalleled efficacy on three rigorously selected datasets, this architecture underscores our methodological superiority in decoding complex human actions from 3D skeleton data, marking a notable advancement in action recognition methodologies.

(2) Strategic Long Short-Term Feature Aggregation: Expounded in Chapter 4, the LSTA-Net represents a sophisticated approach designed to understand both proximal and distal joint dependencies within spatial and temporal dimensions. By employing a nuanced multi-scale spatial aggregation network coupled with an attention-driven temporal pyramid aggregation network, this strategy underscores our model's adeptness. Achieving superior performance on a trio of public datasets, LSTA-Net confirms its robustness and scalability, reinforcing its applicability in analysing human actions' dynamics.

(3) Revolutionary Part-aware Prototypical Network for One-shot Fine-Grained Action Recognition: To address the problem caused by scarce annotations in novel action classes, Chapter 5 proposed a state-of-the-art part-aware prototypical network. Through a meta-learning paradigm, this model excels in classifying actions with minimal examples, effectively capturing and integrating motion patterns across both holistic and part level. Furthermore, an innovative class-agnostic attention fusion mechanism is introduced to discover the relevance of specific body parts within each action class, employing a contrastive learning technique. This breakthrough model achieves leading performance under one-shot learning scenarios on two benchmark datasets, showcasing its preeminent capability to generalise and adapt to novel actions with limited data.

1.5 Organisation of Thesis

The remainder of this thesis is organised as follows:

Chapter 2 - We review the related works and relevant concepts that are necessary for next chapters.

Chapter 3 - We introduce the proposed DualHead-Net for fully-supervised skeleton-based action recognition.

Chapter 4 - We introduce the proposed LSTA-Net for fully-supervised skeleton-based action recognition.

Chapter 5 - We tackle the one-shot learning problem and introduce the part-aware graph prototypical network for one-shot skeleton-based action recognition.

Chapter 6 - We conclude this thesis and show our future research directions.

1.6 Publications

- **Tailin Chen**, Desen Zhou, Jian Wang, Shidong Wang, Yu Guan, Xuming He and Errui Ding. "Learning Multi-Granular Spatio-Temporal Graph Network for Skeleton-based Action Recognition", ACM Multimedia (**MM**), 2021.

- **Tailin Chen**, Shidong Wang, Desen Zhou and Yu Guan. "LSTA-Net: Long short-term Spatio-Temporal Aggregation Network for Skeleton-based Action Recognition", British Machine Vision Conference (**BMVC**), 2021.

- **Tailin Chen**, Desen Zhou, Jian Wang, Shidong Wang, Qian He, Chuanyang Hu, Errui Ding, Yu Guan and Xuming He. "Part-aware Prototypical Graph Network for One-shot Skeleton-based Action Recognition", International Conference on Automatic Face and Gesture Recognition (**FG**), 2023. **Best Student Paper**

Chapter 2

Literature Review

In this chapter, we first review the background knowledge of skeleton datasets. Then we review the existing methods that are relevant to our proposed models for skeleton-based action recognition. Finally, for the convenience of readers to understand the following chapters, we provide some preliminaries of the most relevant concepts.

2.1 Skeleton-based Action Recognition Datasets

In this thesis, four public skeleton datasets are used for experiments. They are NTU RGB+D 60 [53], NTU RGB+D 120 [37], NW-UCLA [77] and Kinetics-Skeleton [29]. Below, we first introduce the ways for skeleton data acquisition and then introduce the details of these datasets.

2.1.1 Skeleton Data Acquisition

Depth Sensors. With the advent of low-cost and easy-to-use depth sensors, such as Microsoft Kinect [29] and Asus Xtion, the 3D skeleton data can be easily and effectively captured in real-time [23]. As shown in Fig. 2.1, we present the commonly

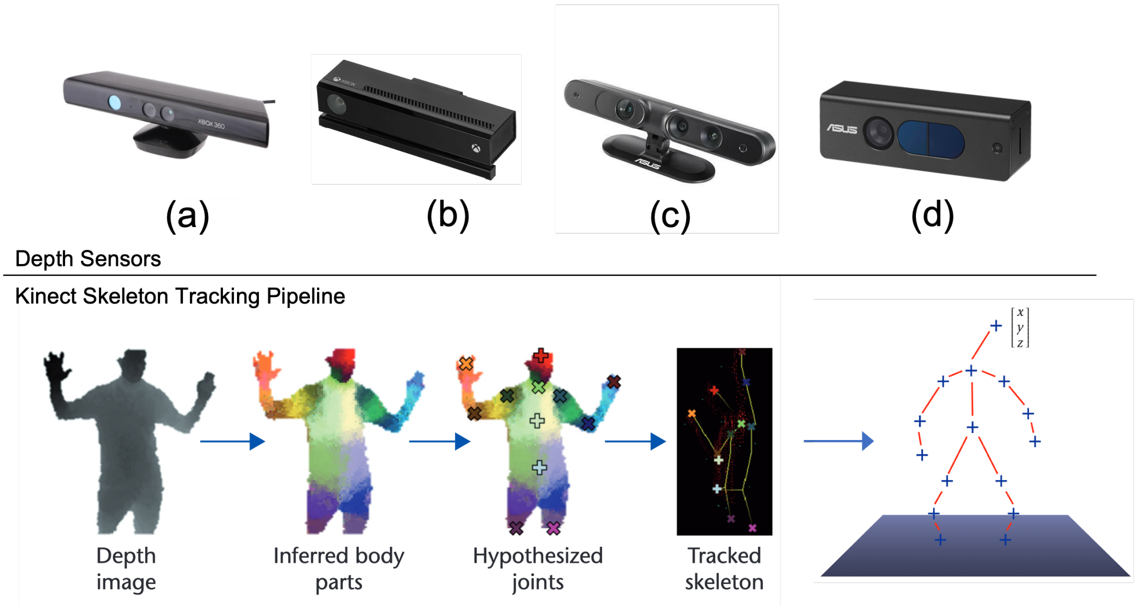


Figure 2.1: **Above:** Illustration of Depth sensors. (a) Microsoft Kinect V1. (b) Microsoft Kinect V2. (c) Asus Xtion. (d) Asus Xtion 2. **Bottom:** Illustration of Kinect Skeleton Tracking Pipeline. After performing perpixel, body-part classification, the system hypothesises the body joints by finding a global centroid of probability mass and then maps these joints to a skeleton using temporal continuity and prior knowledge [29].

adopted depth sensors for data collection and introduce the kinect skeleton tracking pipeline for skeleton data generation.

Pose Estimation Algorithms. With the development of advanced human body pose estimation algorithms [5, 45, 66], estimating human skeletons from images and RGB video is much easier in recent years.

Specifically, Moore et.al. [45] propose a new method to quickly and accurately predict 3D positions of body joints from a single depth image, without using temporal information, as shown in Fig. 2.2. In recent years, Cao et.al. propose OpenPose [5], an off-the-shelf real-time multi-person 2D pose estimation toolbox to jointly detect human body, foot, hand, and facial keypoints (in total 135 keypoints) in an image. Sun et.al. [66] propose a novel High-Resolution Net (HRNet), which is able to maintain high resolution representations for accurate pose estimation. The model architectures and output examples of OpenPose and HRNet are shown in Fig.2.3(a)

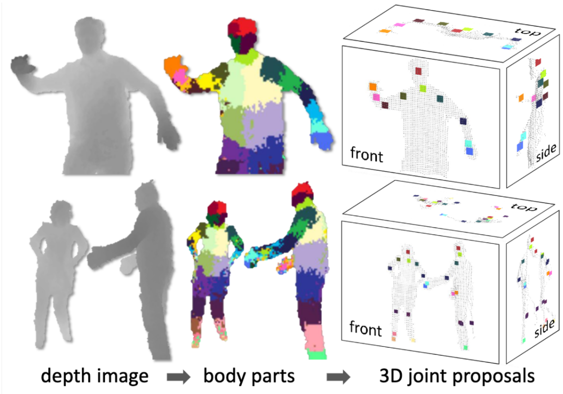


Figure 2.2: Illustration of the "Real-Time Human Pose Recognition in Parts from Single Depth Images" [45]. From an single input depth image, a per-pixel body part distribution is inferred.

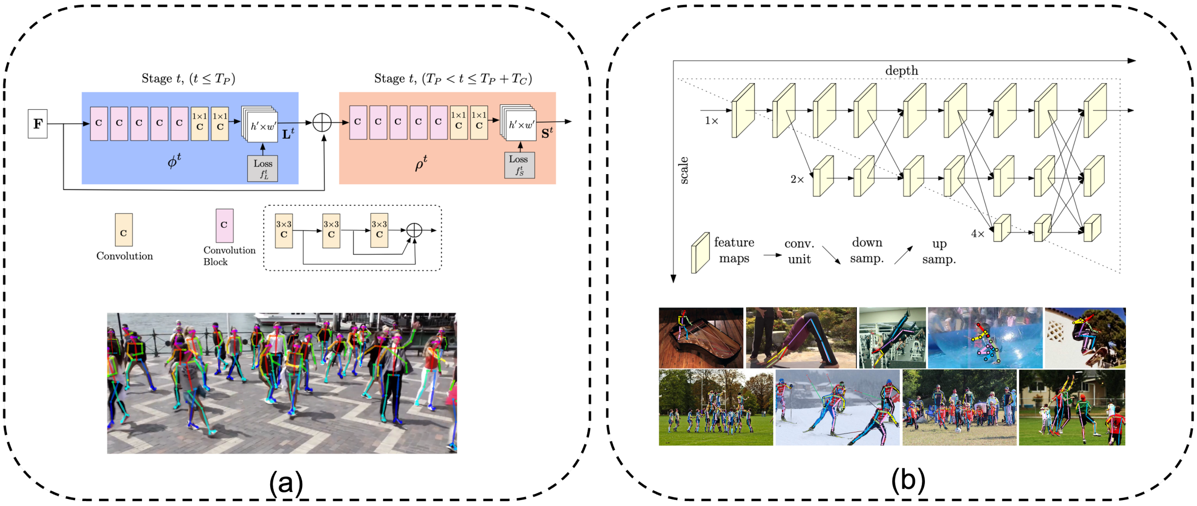


Figure 2.3: Illustration of the overall architecture of recent deep learning-based pose estimation models.**(a)** OpenPose [5] **(b)** HRNet [66].

and Fig.2.3(b), respectively. Hence, the task of skeleton-based action recognition is further accelerated with the help of massive powerful pose estimation algorithms.

2.1.2 NTU RGB+D 60

NTU RGB+D 60 [53] is a large-scale indoor-captured 3D skeleton-based action recognition dataset, containing 56,880 skeleton sequences of 60 action classes captured from 40 distinct subjects aged from 10 to 35 years, covering three different camera views($-45°$, 0, $45°$). Among these action classes, 49 are performed by single

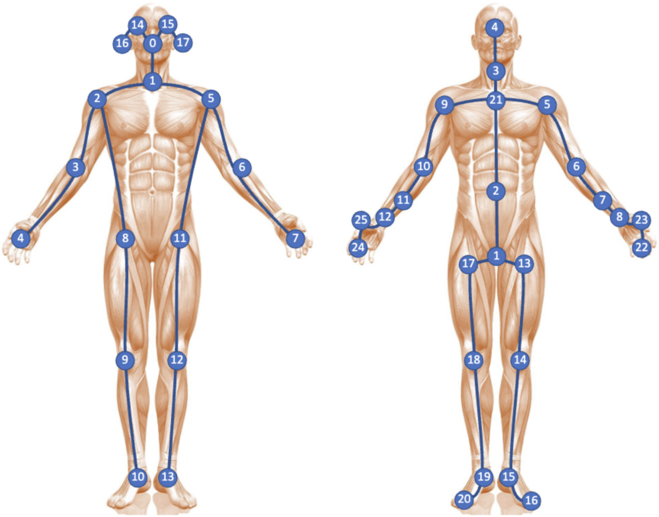


Figure 2.4: The **left sub-graph** illustrate the configuration of 18 body joints of the Kinetics-Skeleton dataset [29]. The **right sub-graph** shows the configuration of 25 body joints in NTU RGB+D 60&120 datasets[53, 37], where the labels of these joints are: (1) base of spine, (2) middle of spine, (3) neck, (4) head, (5) left shoulder, (6) left elbow, (7) left wrist, (8) left hand, (9) right shoulder, (10) right elbow, (11) right wrist, (12) right hand, (13) left hip, (14) left knee, (15) left ankle, (16) left foot, (17) right hip, (18) right knee, (19) right ankle, (20) right foot, (21) spine, (22) tip of left hand, (23) left thumb, (24) tip of right hand, (25) right thumb. Figure from [57].

persons, and 11 are interactions between two people. Each skeleton sequence contains the 3D spatial coordinates of 25 joints captured by the Microsoft Kinect v2 cameras. The configuration and the given order of joints are shown in Fig.2.4, and the action names are shown in Table 2.1

2.1.3 NTU RGB+D 120

NTU RGB+D 120 [37] is an extension of the NTU RGB+D 60 [53] in terms of the number of performers and action categories and currently is the largest 3D skeleton-based action recognition dataset containing 114,480 action samples from 120 action classes. The skeletal samples are captured by 106 subjects with three camera views. There are 32 setups in total, each representing a specific location and background. The configuration and the given order of joints are shown in Fig.2.4, and the action names are shown in Table 2.1

NTU RGB+D 60 & 120 Dataset

Daily Actions			
A1: drink water	A2: eat meal	A3: brush teeth	A4: brush hair
A5: drop	A6: pick up	A7: throw	A8: sit down
A9: stand up	A10: clapping	A11: reading	A12: writing
A13: tear up paper	A14: put on jacket	A15: take off jacket	A16: put on a shoe
A17: take off a shoe	A18: put on glasses	A19: take off glasses	A20: put on a hat/cap
A21: take off a hat/cap	A22: cheer up	A23: hand waving	A24: kicking something
A25: reach into pocket	A26: hopping	A27: jump up	A28: phone call
A29: play with phone/tablet	A30: type on a keyboard	A31: point to something	A32: taking a selfie
A33: check time (from watch)	A34: rub two hands	A35: nod head/bow	A36: shake head
A37: wipe face	A38: salute	A39: put palms together	A40: cross hands in front
A61: put on headphone	A62: take off headphone	A63: shoot at basket	A64: bounce ball
A65: tennis bat swing	A66: juggle table tennis ball	A67: hush	A68: flick hair
A69: thumb up	A70: thumb down	A71: make OK sign	A72: make victory sign
A73: staple book	A74: counting money	A75: cutting nails	A76: cutting paper
A77: snap fingers	A78: open bottle	A79: sniff/smell	A80: squat down
A81: toss a coin	A82: fold paper	A83: ball up paper	A84: play magic cube
A85: apply cream on face	A86: apply cream on hand	A87: put on bag	A88: take off bag
A89: put object into bag	A90: take object out of bag	A91: open a box	A92: move heavy objects
A93: shake fist	A94: throw up cap/hat	A95: capitulate	A96: cross arms
A97: arm circles	A98: arm swings	A99: run on the spot	A100: butt kicks
A101: cross toe touch	A102: side kick	-	-
Medical Conditions			
A41: sneeze/cough	A42: staggering	A43: falling down	A44: headache
A45: chest pain	A46: back pain	A47: neck pain	A48: nausea/vomiting
A49: fan self	A103: yawn	A104: stretch oneself	A105: blow nose
Mutual Actions			
A50: punch/slap	A51: kicking	A52: pushing	A53: pat on back
A54: point finger	A55: hugging	A56: giving object	A57: touch pocket
A58: shaking hands	A59: walking towards	A60: walking apart	A106: hit with object
A107: wield knife	A108: knock over	A109: grab stuff	A110: shoot with gun
A111: step on foot	A112: high-five	A113: cheers and drink	A114: carry object
A115: take a photo	A116: follow	A117: whisper	A118: exchange things
A119: support somebody	A120: rock-paper-scissors	-	-

Table 2.1: The actions in these two datasets are in three major categories: **Daily Actions**, **Mutual Actions** and **Medical Conditions**. Note: actions labelled from A1 to A60 are contained in“NTU RGB+D”, and actions labelled from A1 to A120 are in“NTU RGB+D 120”.

2.1.4 Kinetics-Skeleton

Kinetics dataset [29] contains approximately 300,000 video clips in 400 classes collected from the Internet. The skeleton information is not provided by the original dataset but estimated by the publicly available Open-Pose toolbox [5] . The captured skeleton information contains 18 body joints, as well as their 2D coordinates and confidence score, named Kinetics-Skeleton dataset. The joint configuration is shown in Fig.2.4.

2.1.5 NW-UCLA

NW-UCLA dataset [77] is captured by three Kinect cameras simultaneously with multiple viewpoints. It contains 1,494 video clips covering 10 action categories, as shown in Table 2.2 and each action was performed by 10 different actors. Each body has 20 skeleton joints, the configuration is similar to NTU RGB+D [53].

NW-UCLA Dataset

A1: pick up with one hand	A2: pick up with two hands	A3: drop trash	A4: walk around	A5: sit down
A6: stand up	A7: donning	A8: doffing	A9: throw	A10: carry

Table 2.2: Action Categories in NW-UCLA

2.2 Preliminary Methods

In this section, we introduce the fundamental concepts and methodologies that form the basis of this thesis. First, we discuss the Graph Convolutional Networks (GCNs), which are well-suited for modelling spatial-temporal dependencies in skeleton data. Next, we delve into Few-shot Learning, a crucial approach for handling scenarios with limited labelled data, which is particularly relevant for skeleton-based action recognition. Finally, we explain the importance of Attention Mechanisms in enhancing model performance by allowing the network to focus on the most informative parts of the data. These preliminary methods provide the foundation for understanding the subsequent contributions of this thesis.

2.2.1 Graph Convolutional Networks (GCNs)

2.2.1.1 GNNs *v.s* GCNs

GNNs are a broad class of models that can operate on arbitrary graph structures, making them applicable to a wide range of graph-based tasks. They can use various forms of neighborhood aggregation, making them adaptable to different types of graphs (e.g., social networks, knowledge graphs, etc.). They also can be designed to capture both local node features and global graph structure information through multi-hop message passing. However, GNNs often suffer from high computational complexity, especially when dealing with large graphs and multiple layers, leading to scalability issues. The representations of nodes from different classes can become indistinguishable, a phenomenon known as over-smoothing, which can hurt performance.

Currently, GCNs are particularly efficient in node classification tasks, as they combine node features with the graph structure. They use convolutional filters that aggregate information from neighboring nodes, making them highly effective at capturing local graph structures. More importantly, they are computationally efficient for sparse graphs, as they avoid the full adjacency matrix and only process local neighborhoods.

2.2.1.2 GCNs on Skeleton Data

As shown in Section. 2.1, skeleton data constitutes a sequence of frames with each frame having a set of 2D (x,y) or 3D (x,y,z) joint coordinates denoting spatial location.

Spatio-Temporal Graph Construction A graph is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices and $\mathcal{E} \subseteq (\mathcal{V} \times \mathcal{V})$ is the set of edges. The relationship of the graph, i.e. the topology of graph, is represented by the graph adjacency matrix $\mathbf{A} \in R^{V \times V}$ with entry $\mathbf{A}_{ij} = 1$, when nodes i, j are connected, and 0 otherwise; V denotes the number of vertexes. $\mathbf{A}$ is a symmetric matrix while $\mathcal{G}$ is an undirected graph. Human graph sequences contain a set of node features $\mathcal{X} = \{x_t^v | 1 \leq v \leq V, 1 \leq t \leq T; v, t \in Z\}$ that can also be represented as $\mathbf{X} \in R^{T \times V \times C}$, where C is the feature dimension (C is 2 or 3 in the beginning).

Spatial Graph Convolutional Block The spatial graph convolutional blocks are defined on each frame $\mathbf{X}_t$ with learnable weights to capture the spatial correlations between human joints. Then, on a given graph, the graph convolution can be implemented similarly to the convolution on a regular grid graph (i.e., RGB image),

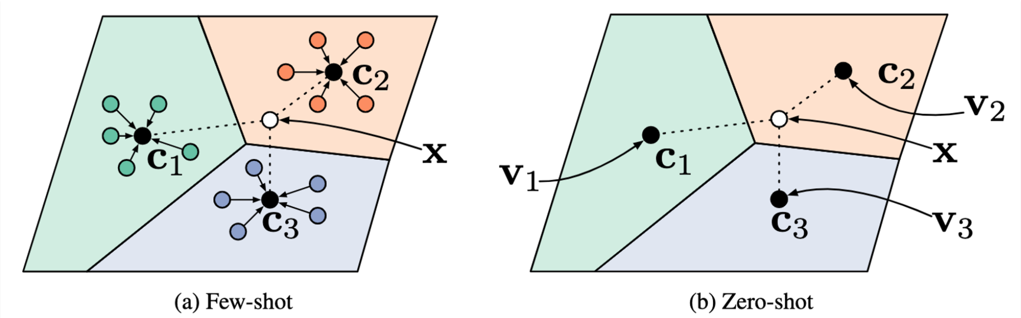
$$\mathbf{Y}_t^{\mathcal{S}} = \sigma\left(\widetilde{\mathbf{A}}\mathbf{X}_t\mathbf{W}\right), \tag{2.1}$$

where $\mathbf{X}_t \in R^{N\times C}$ and $\mathbf{Y}_t^{\mathcal{S}} \in R^{N\times C_{\text{out}}}$ denote the input and output features respectively. Here $\mathbf{W} \in R^{C\times C_{\text{out}}}$ are the graph convolution weights. $\sigma(\cdot)$ is the activation function. $\widetilde{\mathbf{A}}$ are the normalised adjacency matrices as described in [32] and can be obtained by: $\widetilde{\mathbf{A}} = \hat{\mathbf{D}}^{-\frac{1}{2}}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-\frac{1}{2}}$, where $\hat{\mathbf{A}} = \mathbf{A}+\mathbf{I}$ is the adjacency matrix including the nodes of the self-loop graph ($\mathbf{I}$ is the identity matrix) and $\hat{\mathbf{D}}$ is the degree matrix of $\mathbf{A}$. The output of entire sequence as $\mathbf{Y}^{\mathcal{S}} = [\mathbf{Y}_1^{\mathcal{S}}, \cdots, \mathbf{Y}_T^{\mathcal{S}}]$. Many graph convolution variants are based on this block.

2.2.2 Few-Shot Learning

Inspired by human's fast learning ability, few-shot learning models are developed and applied on learning novel concept representations from limited training samples. Specifically, after training a model on certain known categories with a large amount of samples, few-shot learning only needs a small number of samples to learn a model on novel categories. Most of the current works can be categorised into optimisation-learning based methods [17, 52, 50, 46], metric-learning based methods [73, 63, **?**] and graph neural network based methods [20].

Our work is inline with the metric-learning based method, ProtoNet [63], as shown in Fig. 2.5, which aims to learn discriminative feature representations of classes under cosine or euclidean distance metric. However, we further adopt this



(a) Few-shot

(b) Zero-shot

Figure 2.5: Illustration of the prototypical networks [63] in the few-shot and zero-shot scenarios. **Left:** Few-shot prototypes c_k are computed as the mean of embedded support examples for each class. **Right:** Zero-shot prototypes c_k are produced by embedding class meta-data v_k. In either case, embedded query points are classified via a softmax over distances to class prototypes.

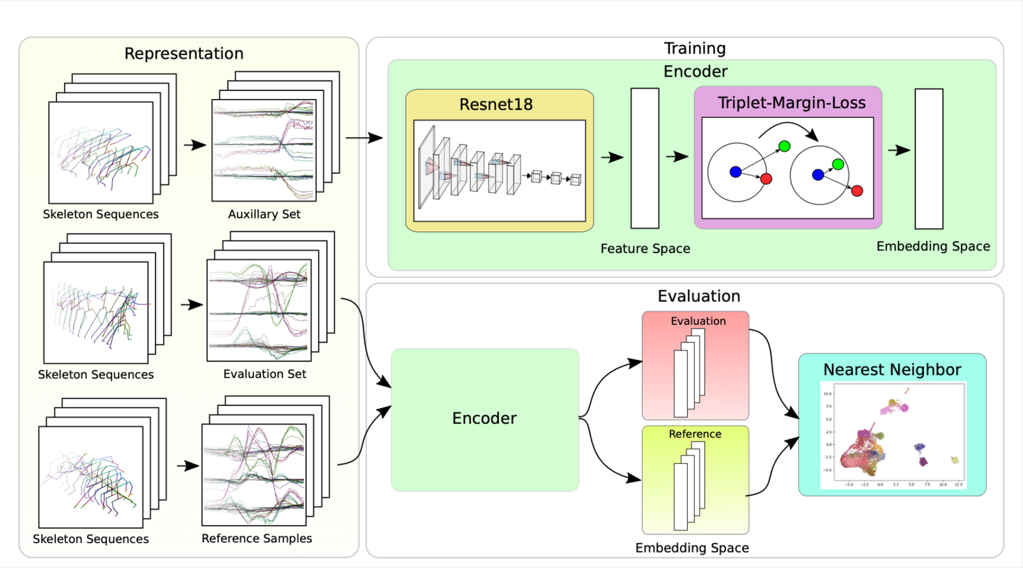


Figure 2.6: Illustration of the Single-Level Deep Metric Learning Method [44].

idea in a challenging skeleton-based action recognition task and focusing on designing a simple yet effective framework for measuring the similarity of skeleton sequences.

2.2.2.1 Terminology in Few-shot Learning

Support Set: The support set consists of the few labelled samples per novel category of data, which a pre-trained model will use to generalise on these new classes.

Query Set: The query set consists of the samples from the new and old categories of data on which the model needs to generalise using previous knowledge and information gained from the support set.

N-way K-shot learning scheme: This is a common phrase used in the FSL literature, which essentially describes the few-shot problem statement that a model will be dealing with. “N-way” indicates that there are “N” numbers of novel categories on which a pre-trained model needs to generalise over. A higher “N” value means a more difficult task. “K”-shot defines the number of labelled samples available in the support set for each of the “N” novel classes. The few-shot task becomes more difficult (that is, lower accuracy) with lower values of “K” because less supporting information is available to draw an inference. When K=1, tasks are given the name “One-Shot Learning” (discussed in Chapter 5).

2.2.3 Attention Mechanism

Attention mechanisms, which select relatively critical information from all inputs, have been recently presented to improve the performance of categorising skeleton sequences. Early works applied attention mechanisms at multiple levels of the network to distinguish actions based on joints dependencies [64]. Then, Xie et al. [83] proposed a temporal-then-spatial re-calibration scheme and further boosted it through the memory attention network. Similarly, [39] aimed to discover informative joints in each frame of each skeleton sequence by using a global context memory cell. Then, Spatial-Temporal Graph Routing (STGR) network [33] added additional edges to skeleton graphs with frame-wise attention and global self-attention mechanisms. Analogously, Two-stream adaptive graph convolutional networks (2s-AGCN) [59] introduced adaptative graphs with non-local self-attention [71] and free-learning graphs as residual masks. Besides, Si et al. [60] employed attention-based graph convolutional LSTM to capture co-occurrence relationships in Spatio-temporal domain. More recently, [13] investigated the effectiveness of modelling spatial-temporal dependencies and capturing long-range correlations by applying self-attention mechanism to the skeleton sequences.

2.3 Related Methods

We first review the traditional methods and the deep learning methods for supervised skeleton-based action recognition, especially the recent GCN-based method in details. Next, we briefly review the few-/one-shot learning for the convenience of readers understanding our one-shot skeleton-based action recognition model. Finally, we review some important attention mechanisms commonly used in image/video understanding.

2.3.1 Hand-Crafted Features for Skeleton-based Action Recognition

As mentioned in previous section, with the advent of cheap and easy-to-use depth sensors, such as Microsoft Kinect [92], and advanced human pose estimation algorithms [5, 14, 66], skeleton-based action recognition becomes very popular [1, 51]. Early approaches typically adopt hand-crafted features to capture the human body motion from the skeleton sequence [72, 75, 76, 82].

Xia et al. [82] introduced a compact representation, namely Histograms of 3D Joint Locations (HOJ3D) features, shown in Fig. 2.7, by assigning 3D joint positions into cone bins. They then proposed to model these features with the Hidden Markov models (HMMs) for action classification. Vemulapalli et al. [72] represented the human skeletons as points and curves in the Lie group, where the 3D geometric transformations between various body-parts are computed, and a support vector machine (SVM) classifier was used to classify the actions. Wang et al. [75, 76]

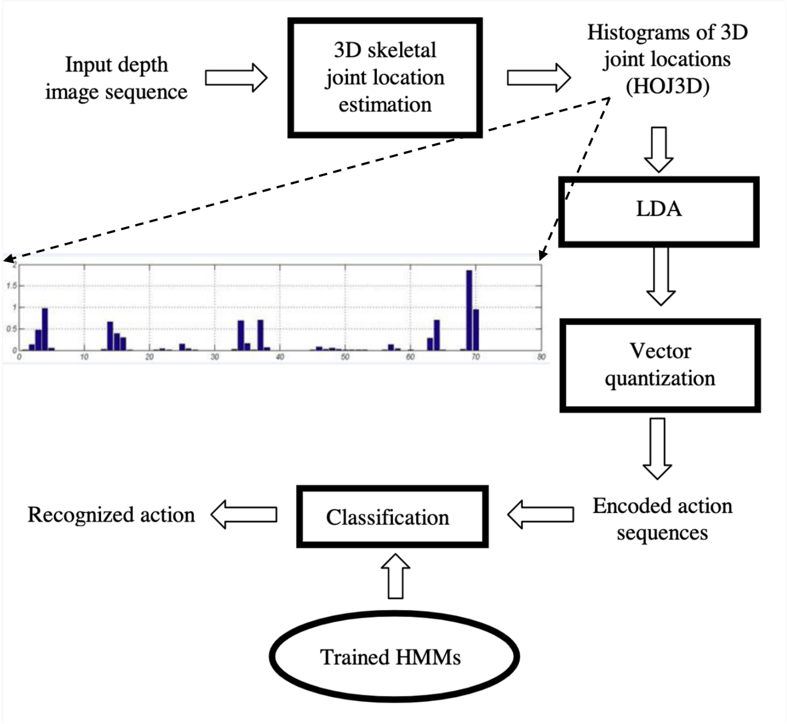


Figure 2.7: Overview of HOJ3D method [82].

presented an actionlet ensemble model to capture the motion patterns meanwhile learning the intra-class variances by computing the pairwise relative positions of each joint with other joints.

Although these methods have shown their effectiveness in terms of computation time and accuracy, they mainly rely on exploiting the relative 3D rotations and translations between joints, and hence suffer from complicated feature design and sub-optimal performance.

2.3.2 Deep Learning Methods for Skeleton-based Action Recognition

In recent years, with the availability of large-scale skeleton datasets and the development of deep learning, data-driven methods have been proven more effective on learning multi-level features from raw data, than those methods based on hand-crafted features and achieved significant progress in skeleton-based action recognition. These methods can be categorised into three groups according to their network architectures: i.e., Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs) and the latest Graph Convolutional Networks (GCNs).

RNN-based Methods. As skeleton sequences are natural time series of joint positions in space-time dimension, therefore the RNN itself is suitable for processing such time series data. Furthermore, some variants of RNN-based methods, such as Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU), have also been adopted for skeleton-based action recognition.

For exampel, Du et al. [16] proposed an end-to-end Hierarchical RNN network to obtain the global representation of human skeletons and perform action classification, as illustrated in Fig. 2.8. Zhu et al. [93], introduced a novel deep LSTM networks to learn the co-occurrence features of skeleton sequences and proposed a novel LSTM dropout algorithm to facilitate the automatic learning. Li et al. [36] constructed an adaptive tree-structured RNN where actions in skeletal representations are recognised via a hierarchical inference process. Zhang et al. [88] designed a view adaptation scheme for modelling actions. Liu et al. [38] proposed an tree-structured spatial-temporal lstm network for skeleton classification, as illustrated in Fig. 2.9, in the spatial direction, body joints in a frame are fed in a sequence, in the temporal direction, the locations of the corresponding joints are fed over time. Besides, in [39], they further employ a global context-aware attention lstm network to mine global informative joints. Wang et al. [74] proposed a two-stream RNN network to model the spatial and temporal features simultaneously. Si et al. [61] adopt LSTM networks to learn stacked temporal dynamics.

CNN-based Methods. Although the RNN-based methods can effectively model the temporal dependencies of skeleton sequences, they still lack the ability in spatial modelling. CNN-based methods, regarded as a complementary of RNN-based meth-

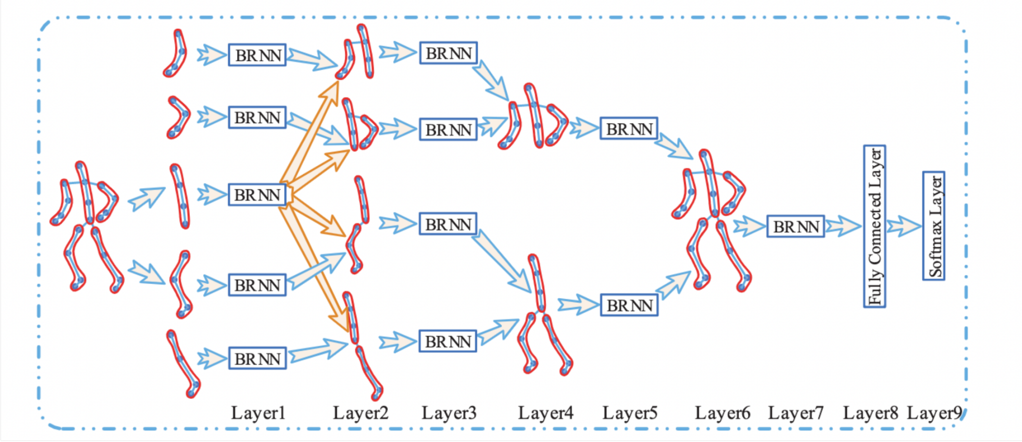


Figure 2.8: Illustration of the Hierarchical RNN network proposed in [16].

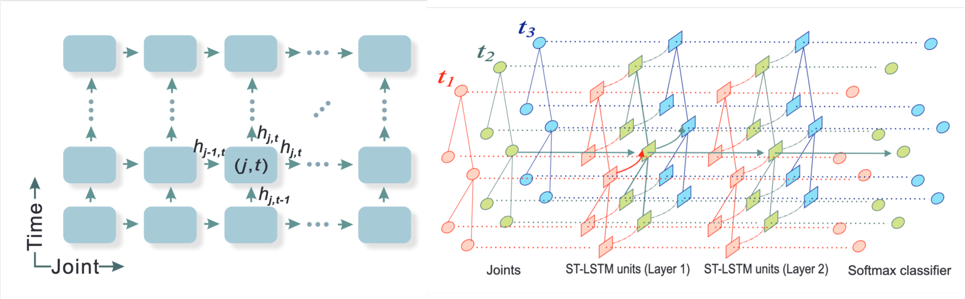


Figure 2.9: Illustration of the Spatial-Temporal LSTM network proposed in [38]. j, t and h stand for the joint and time-step and mid-level hidden states

ods, are proposed to learn spatial high-level semantic cues for skeleton-based action recognition. These CNN-based methods normally transform the skeleton sequence into a pseudo-image and then employ a popular network, such as ResNet [25] to explore the spatial and temporal dynamics for action recognition.

In particular, Ke et al. [30] convert the skeleton sequence into clips, which are represented by the relative position between four selected reference joints, as illustrated in Fig.2.10, the generated clips are then fed into a deep CNN model to extract CNN features which are used in a multitask learning network for action recognition. Li et al. [34] designed a co-occurrence feature learning framework. Kim et al. [31] used an one-dimensional residual CNN to identify skeleton sequences based on directly-concatenated joint coordinates. Liu et al. [40] proposed 10 types of spatio-temporal images for skeleton encoding, which are enhanced by visual and

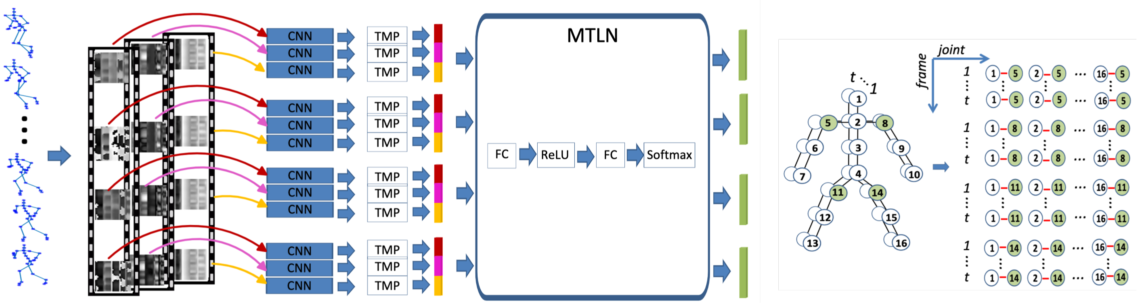


Figure 2.10: Illustration of the overall architecture of Clip-CNN+MTLN model proposed in [30].

motion features. The success of CNNs is attributed to their strong capability in feature representation. However, both the RNNs and CNNs have difficulty in capturing the skeleton topology which are naturally of a graph structure.

GCN-based Methods. Graph Convolutional Networks (GCNs) have become a powerful tool for skeleton-based action recognition due to their ability to naturally represent the human body as a spatial-temporal graph, where each joint is a node and the connections between joints form edges. This structured representation allows GCNs to capture both spatial (within a frame) and temporal (across frames) dependencies in human motion. Over the years, several directions and advancements have emerged in applying GCNs to skeleton-based tasks. Below, I will introduce some key findings in GCN-based methods for skeleton-based action recognition:

Particularly, Yan et al. [84] proposed a milestone work, spatial temporal graph convolutional networks (ST-GCN) to model the graph-structured data. As illustrated in Fig. 2.11, their method defines a sparse connected spatial-temporal graph that both considers natural human body structure and temporal motion dependencies in space-time domain. Multiple layers of ST-GCN are stacked and gradually generate higher-level feature maps on the graph. The final output features can be classified by a standard Softmax classifier to the corresponding action category.

After this, a large body of ST-GCN variants were proposed in the past a few years, tackling specific limitations existing in the original implementation. Shi et

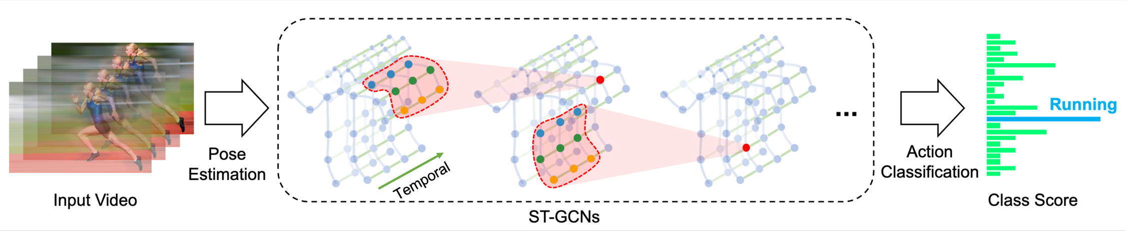


Figure 2.11: Illustration of the architecture of Spatial Temporal Graph Convolutional Networks (ST-GCN) proposed in [84]

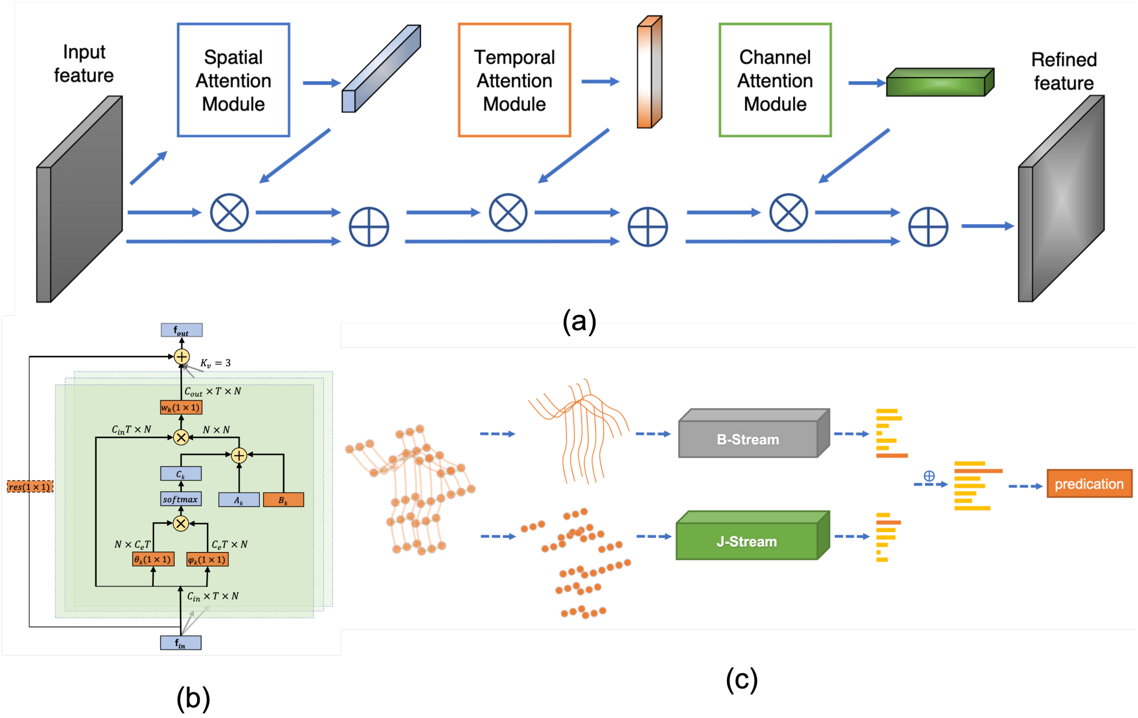


Figure 2.12: Adaptive Graph Convolutional Networks (AGCN) [58, 57]. **(a)** Illustration of the Spatial-Temporal-Channel Attention (STC-Attention) module. **(b)** Illustration of the adaptive graph convolutional layer (AGCL). **(c)** Illustration of the overall architecture of the 2s-AGCN. N, C, T represent the number of joint nodes, channel dimensions and temporal dimensions.

al. [57, 58], based on basic ST-GCN [84], proposed an adaptive attention module , as illustrated in Fig. 2.12 (b), to learn non-local dependencies of joints in spatial dimension, and further introduced SE-like [26] spatial, temporal and channel attentions in the GCNs for feature enhancement, as shown in Fig. 2.12 (a). More importantly, they devise a multi-stream fusion framework which utilise the high-order information of original Joint data, such as Bone data, which effectively improved the recognition

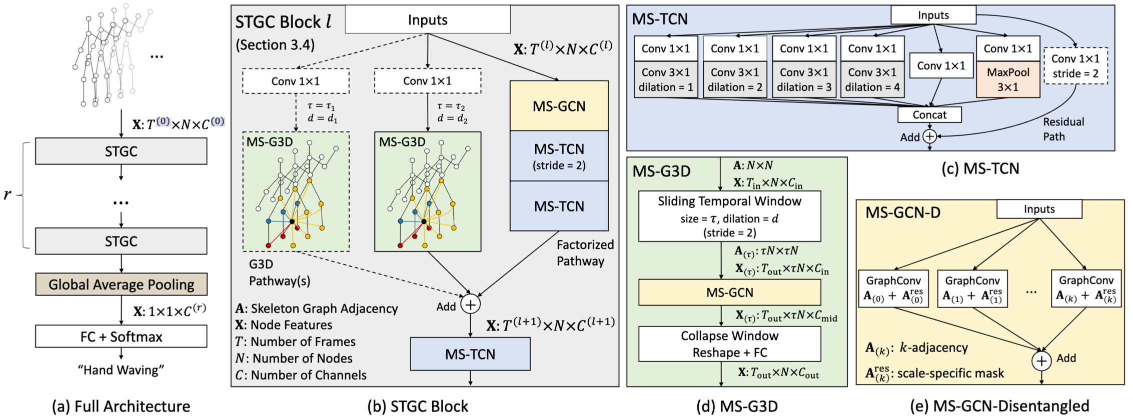

...
$\mathbf{X}: T^{(0)}\times N\times C^{(0)}$
STGC
r
...
STGC
Global Average Pooling
$\mathbf{X}: 1\times1\times C^{(r)}$
FC + Softmax
"Hand Waving"
(a) Full Architecture
STGC Block l
(Section 3.4)
Inputs
$\mathbf{X}: T^{(l)}\times N\times C^{(l)}$
Conv 1×1
Conv 1×1
$\tau = \tau_1$
$d = d_1$
$\tau = \tau_2$
$d = d_2$
MS-G3D
MS-G3D
MS-GCN
MS-TCN
(stride = 2)
MS-TCN
G3D
Pathway(s)
Factorized
Pathway
Add
$\mathbf{X}: T^{(l+1)}\times N\times C^{(l+1)}$
MS-TCN
$\mathbf{A}$: Skeleton Graph Adjacency
$\mathbf{X}$: Node Features
T: Number of Frames
N: Number of Nodes
C: Number of Channels
(b) STGC Block
MS-TCN
Inputs
Conv 1×1
Conv 3×1
dilation = 1
Conv 1×1
Conv 3×1
dilation = 2
Conv 1×1
Conv 3×1
dilation = 3
Conv 1×1
Conv 3×1
dilation = 4
Conv 1×1
Conv 1×1
MaxPool
3×1
Conv 1×1
stride = 2
Concat
Add
Residual
Path
(c) MS-TCN
MS-G3D
$\mathbf{A}: N\times N$
$\mathbf{X}: T_{\text{in}}\times N\times C_{\text{in}}$
Sliding Temporal Window
size = τ, dilation = d
(stride = 2)
$\mathbf{A}_{(\tau)}: \tau N\times\tau N$
$\mathbf{X}_{(\tau)}: T_{\text{out}}\times\tau N\times C_{\text{in}}$
MS-GCN
$\mathbf{X}_{(\tau)}: T_{\text{out}}\times\tau N\times C_{\text{mid}}$
Collapse Window
Reshape + FC
$\mathbf{X}: T_{\text{out}}\times N\times C_{\text{out}}$
(d) MS-G3D
MS-GCN-D
Inputs
GraphConv
$\mathbf{A}_{(0)} + \mathbf{A}^{\text{res}}_{(0)}$
GraphConv
$\mathbf{A}_{(1)} + \mathbf{A}^{\text{res}}_{(1)}$
...
GraphConv
$\mathbf{A}_{(k)} + \mathbf{A}^{\text{res}}_{(k)}$
$\mathbf{A}_{(k)}$: k-adjacency
$\mathbf{A}^{\text{res}}_{(k)}$: scale-specific mask
Add
(e) MS-GCN-Disentangled

Figure 2.13: Illustration of the overall architecture of MS-G3D [41]. "TCN", "GCN", prefix"MS-", and suffix"-D" denotes temporal and graph convolutional blocks, and multi-scale and disentangled aggregation, respectively. Each of the r STGC blocks (b) deploys a multi-pathway design to capture long-range and regional spatial-temporal dependencies simultaneously. Dotted modules, including extra G3D pathway, 1×1 conv, and strided temporal convolutions, are situational for model performance/complexity trade-off.

performance, as shown in Fig. 2.12 (c), a two-stream architecture where scores of two streams are added to obtain the final prediction.

Other works, for example, Zhang et al. [91] explored contextual information between joints. Cheng et al. [11] introduced the shift operation from CNNs to GCNs and intended to address the computational complexity of the original GCNs network. Song et al. [65] utilised multiple data modality and adopt a residual-like structure in a single model. Ye et al. [9] proposed a multi-scale spatial temporal graph for long range modelling. Particularly, Liu et al. [41] proposed a disentangled multi-scale aggregation model (MS-G3D) to effectively aggregate the spatial and temporal features. As illustrated in Fig. 2.13, this aggregation scheme can disentangles the importance of nodes in different neighbourhoods for effective long-range modelling, and the G3D module can leverages dense cross-spacetime edges as skip connections for direct information propagation across the spatial-temporal graph.

Few-shot Skeleton-based Action Recognition There are few works focus on few(one)-shot 3D skeleton-based action recognition [37, 44, 43, 78]. Action-Part Semantic Relevance-aware (APSR) framework [37] adopts the semantic relevance between each body part and each action class at the distributed word embedding space. Single Level Deep Metric Learning[44] and Skeleton-DML [43] convert the original skeletons into images, extract features using CNNs and apply multi-similarity miner losses, as shown in Fig. 2.6. The most recent work JEANIE [78] encodes the 3D body joints into temporal blocks with GNNs and then simultaneously perform temporal and view-point alignment of query-support in the meta-learning regime.

However, these networks inevitably pool/fuse features across different frames and parts to holistically extract a single feature vector representing the whole skeleton sequences and fail to maintain local discriminative features. Such feature is vital important for some fine-grained classes classification, and especially in this data-insufficient one-shot setting. In contrast, our method adopts the meta-learning regime and constructs multiple part sub-graphs during representation learning stage aiming to extract multiple discriminative local descriptors which can help the later query-support matching stage. Furthermore, an attention-based selection module is proposed for sub-graphs selection. ResGCN [65] proposes a part attention block to enhance part features. However, it generates part representations based on the holistic body level modelling, while ours exploit the zoom-in module for further part level modelling.

Chapter 3

Learning Multi-Granular Spatio-Temporal Graph Network for Skeleton-based Action Recognition

In our first work, we focus on skeleton-based action recognition, and a new GCN-based network model (DualHead-Net), is introduced for this task. Our method jointly models the coarse- and fine-grained skeleton motion patterns which enables us to extract features at two spatio-temporal resolutions from the skeleton sequences in an effective and efficient manner.

The need for effective feature extraction and the need for precise temporal sequence analyse motivate us to utilise two branches of interleaved graph networks to extract features at two different temporal resolutions. Such a dual-head net design allows each "head" of the network to specialise in distinct aspects of the action recognition process, such as coarse-grained and fine-grained features. This architecture also offers flexibility in adapting to various complexities and variations in action data. It can be scaled or modified more readily than a monolithic system because each head can be independently adjusted or improved without necessitating a complete overhaul of the entire network. By having separate modules for feature extraction and temporal analysis, the network can reduce the risk of over-fitting and lead to improvements in both accuracy and computational efficiency.

3.1 Introduction

Action recognition is a fundamental task in human-centred scene understanding and has achieved much progress in computer vision and multimedia. Recently, skeleton-based action recognition has attracted increasing attention to the community due to the advent of inexpensive motion sensors [5, 53] and effective human pose estimation algorithms [84, 58, 41]. The skeleton data typically are more compact and robust to environment conditions than its video counterpart, and accurate action recognition from skeletons can greatly benefit a wide range of applications, such as human-computer interactions, healthcare assistance and physical education.

Different from RGB videos, skeleton data contain only the 2D or 3D coordinates of human joints, which makes skeleton-based action recognition particularly challenging due to the lack of image-based contextual information. In order to capture discriminative spatial structure and temporal motion patterns, existing methods [57, 84, 41] usually rely on a shared spatio-temporal representation for the skeleton data at the original frame rate of input sequences. While such a strategy may have the capacity to capture action patterns of multiple scales, they often suffer from inaccurate predictions on fine-grained action classes due to high model complexity and limited training data. To tackle this problem, we argue that a more effective solution is to explicitly model the motion patterns of skeleton sequences at multiple temporal granularity. For example, actions such as ‘hand waving’ or ‘stand up’ can

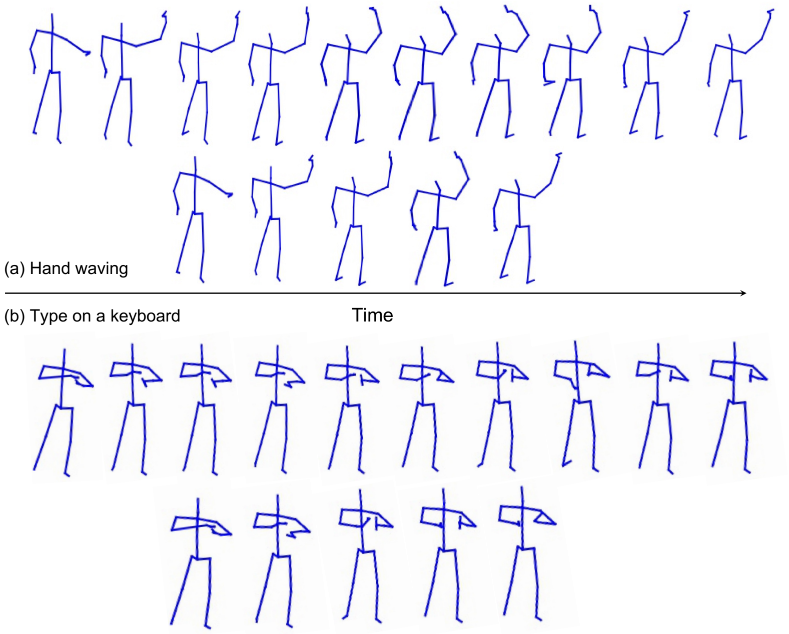


Figure 3.1: Some actions can be recognised via coarse-grained temporal motion patterns, such as 'hand waving' (top), while recognising other actions requires not only coarse-grained motion but subtle temporal movements, such as 'type on a keyboard' (bottom).

be distinguished based on coarse-grained motion patterns, while recognising actions like 'type on a keyboard' or 'writing' requires understanding not only coarse-grained motions but also subtle temporal movements of pivotal joints, as shown in Figure.3.1. To the best of our knowledge, such multiple granularity of temporal motion information remains implicit in the recent deep graph network-based approaches, which is less effective in practice.

In this chapter, we propose a dual-head graph neural network framework for skeleton-based action recognition in order to capture both coarse-grained and fine-grained motion patterns. Our key idea is to utilise two branches of interleaved graph networks to extract features at two different temporal resolutions. The branch with lower temporal resolution captures motion patterns at a coarse level, while the branch with higher temporal resolution is able to encode more subtle temporal movements. Such coarse-level and fine-level feature extraction are processed in parallel and finally the outputs of both branches are fused to perform dual-granular action classification.

Specifically, we first exact a base feature representation of the input skeleton sequence by feeding it into a backbone Graph Convolution Network (GCN). Then we perform two different operations to the resulting feature maps: the first operation subsamples the feature maps at the temporal dimension with a fixed downsampling rate, which removes the detailed motion information and hence produces a coarse-grained representation; in the second operation, we keep the original temporal resolution and utilise an embedding function to generate a fine-grained representation.

Subsequently, we develop two types of GCN modules to process the resulting coarse- and fine-grained representations, which are referred as *fine head* and *coarse head*. Each head consists of two sequential GCN blocks, which extract features within respective granularity. In particular, our coarse head captures the correlations between joints at a lower temporal resolution, hence infers actions in a more holistic manner. To facilitate such coarse-level inference, we estimate a temporal attention from the fine-grained features in the fine head, indicating the importance of each frame. The temporal attention is used to re-weight the features at the coarse head. The intuition behind such cross head attention is as follows: here we pass the fine-grained motion contexts encoded in the attention to the coarse head in order to remedy the lack of fine level information in the coarse head. Similarly, we utilise the coarse-grained features to estimate a spatial attention indicating the importance of joints. The spatial attention highlights the pivotal joint nodes in the fine head. Our fine GCN blocks are able to focus on the subtle temporal movements of the pivotal joints and hence extract the fine-grained information effectively. Finally, each head predicts an action score, and the final prediction is given by fusion of two scores.

The remainder of this chapter is organised as follows. In Section 3.2, we introduce our end-to-end trainable dual-head graph neural network (DualHead-Net) for skeleton-based action recognition. The experiments are presented in Section 3.3.

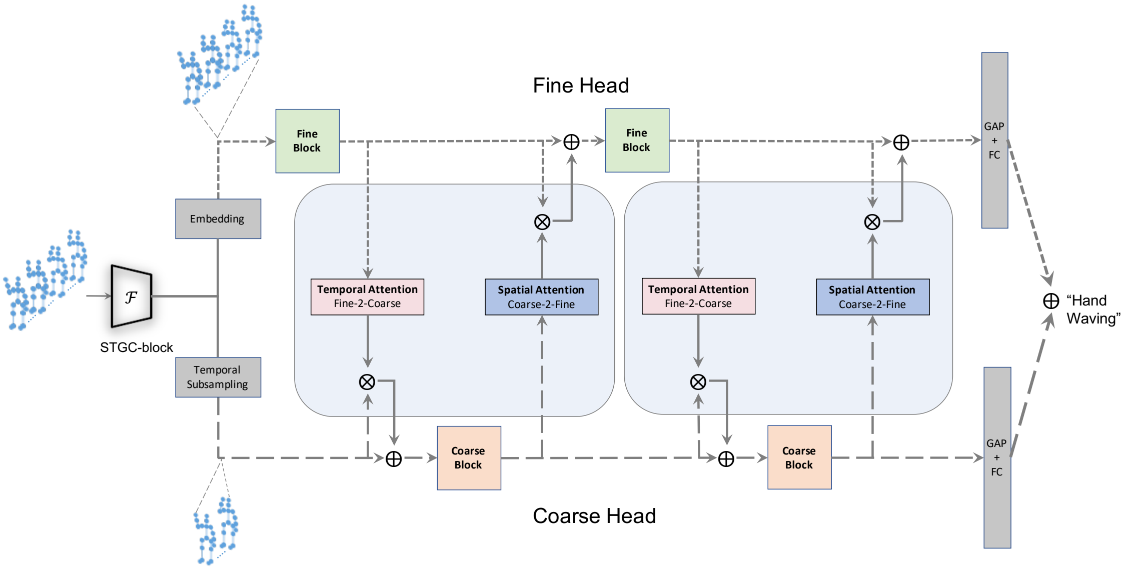


Figure 3.2: Overview of our proposed framework. We first utilise a STGC block to generate backbone features of skeleton data, and then use a coarse head to capture coarse-grained motion patterns, and a fine head to encode fine-grained subtle temporal movements. Cross-head spatial and temporal attentions are exploited to mutually enhance feature representations. Finally each head generates a probabilistic prediction of actions and the ultimate estimation is given by fusion of both predictions.

Finally, the chapter is concluded in Section 3.4.

3.2 DualHead-Net

3.2.1 Overview

Our goal is to jointly capture the multi-granular spatio-temporal dependencies in skeleton data and learn discriminative representations for action recognition. To this end, we develop a novel dual-head network that explicitly captures motion patterns at two different spatio-temporal granular levels. Each head of our network adopts a different temporal resolution and hence focuses on extracting a specific type of motion features. In particular, a fine head maintains the original temporal resolutions as the input so that it can model fine-grained local motion patterns, while a coarse head uses a lower temporal resolution via temporal sub-sampling so that it can focus more on coarse-level temporal contexts. This design allows the model to capture features at different temporal scales, where one head focuses on extracting rough patterns of the overall action (such as the general flow of the movement), while the other concentrates on the details of the movement (such as minor motions of hands and feet). This dual-head structure provides a more comprehensive understanding and analysis of action data than a single-head approach. Moreover, we further introduce a cross-head attention module to ensure that the extracted information from different heads can be communicated in a mutually reinforcing way. The cross-attention mechanism facilitates interaction between the two heads, enhancing the model's ability to understand and abstract action features. Through this method, each head not only processes the information it directly learns but also utilises features from the other head to optimise its own representations, thereby improving overall recognition accuracy and robustness. While a single branch can focus on feature extraction and attention at a specific temporal resolution, it cannot simultaneously account for both the overall and detailed aspects of an action. This structure, complemented by cross-attention at overall and detailed level, allowing for a fuller understanding of complex actions. Furthermore, in dealing with complex or diverse actions, key information about some actions may exist across different temporal scales. Such cross-attention mechanism captures this information more flexibly, adapting to various complex scenarios effectively. Finally, the dual-head network generates its final prediction by fusing the output scores of both heads.

The details of the proposed method are organised as follows. Firstly, we introduce the GCNs (Section 3.2.2) and backbone module (Section 3.2.3) for skeletal feature extraction. Secondly, we depict the dual-head module (Section 3.2.4), the cross-communication attention module (Section 3.2.5) and the fusion module (Section 3.2.6). Finally, we describe the details of the multi-modality ensemble strategy (Section 3.2.7).

3.2.2 GCNs on Skeleton Data

The proposed framework adopts graph convolutional networks to effectively capture the dependencies between dynamic skeleton joints. Below we introduce three basic network modules used in our method, including MS-GCN, MS-TCN and MS-G3D.

Formally, given a skeleton of N joints, we define a skeleton graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, ..., v_N\}$ is the set of N joints and $\mathcal{E}$ is the collection of edges. The graph connectivity can be represented by the adjacency matrix $\mathbf{A} \in R^{N \times N}$, where its element value takes 1 or 0 indicating whether the positions of v_i and v_j are adjacent. Specifically, the adjacency matrix A is a square matrix used to represent graph connectivity, where each element $A_{i,j}$ is either 1 or 0, indicating whether the vertices v_i and v_j are adjacent (connected directly by an edge) or not. Given a skeleton sequence, we first compute a set of features $\mathcal{X} = \{x_{t,n} | 1 \leq t \leq T, 1 \leq n \leq N; n, t \in Z\}$ that can be represented as a feature tensor $\mathbf{X} \in R^{C \times T \times N}$, where $x_{t,n} = \mathbf{X}_{t,n}$ denotes the C dimensional features of the node v_n at time t.

3.2.2.1 MS-GCN

The spatial graph convolutional blocks(GCN) aims to capture the spatial correlations between human joints within each frame $\mathbf{X}_t$. We adopt a multi-scale GCN(MS-GCN) to jointly capture multi-scale spatial patterns in one operator:

$$\mathbf{Y}_t^{\mathcal{S}} = \sigma\left(\sum_{k=0}^{K} \widetilde{\mathbf{A}}_k \mathbf{X}_t \mathbf{W}_k\right), \tag{3.1}$$

where $\mathbf{X}_t \in R^{N \times C}$ and $\mathbf{Y}_t^{\mathcal{S}} \in R^{N \times C_{\text{out}}}$ denote the input and output features respectively. Here $\mathbf{W}_k \in R^{C \times C_{\text{out}}}$ are the graph convolution weights, and K indicates the number of scales of the graph to be aggregated. $\sigma(\cdot)$ is the activation function. $\widetilde{\mathbf{A}}_k$ are the normalised adjacency matrices as in [32, 35] and can be obtained by:

$\widetilde{\mathbf{A}}_k = \hat{\mathbf{D}}_k^{-\frac{1}{2}} \hat{\mathbf{A}}_k \hat{\mathbf{D}}_k^{-\frac{1}{2}}$, where $\hat{\mathbf{A}}_k = \mathbf{A}_k + \mathbf{I}$ is the adjacency matrix including the nodes of the self-loop graph ($\mathbf{I}$ is the identity matrix, which is a square matrix with ones on the diagonal and zeros elsewhere, serving as the multiplicative identity in matrix operations.) and $\hat{\mathbf{D}}_k$ is the degree matrix of $\mathbf{A}_k$, which is a diagonal matrix where each diagonal element $D_k(i,i)$ represents the degree (i.e., the number of connections or edges) of the i-th vertex in the adjacency matrix A_k. We denote the output of entire sequence as $\mathbf{Y}^{\mathcal{S}} = [\mathbf{Y}_1^{\mathcal{S}}, \cdots, \mathbf{Y}_T^{\mathcal{S}}]$.

3.2.2.2 MS-TCN

The temporal convolution(TCN) is formulated as a classical convolution operation on each joint node across frames. We adopt multiple TCNs with different dilation rates to capture temporal patterns more effectively. A N-scale MS-TCN with kernel size of $K_t \times 1$ can be expressed as,

$$\mathbf{Y}_{\mathcal{T}} = \sum_{i}^{N} Conv2D[K_t \times 1; D^i](\mathbf{X}). \tag{3.2}$$

where D^i denotes the dilatation rate of i^{th} convolution.

3.2.2.3 MS-G3D

To jointly capture spatio-temporal patterns, a unified graph operation (G3D) on space and time dimension is used. As shown in Literature Review section Figure. 2.13, a multi-scale G3D is also adopted in our model.

3.2.3 Backbone Module

We first describe the backbone module of our network, which computes the base features of the input skeleton sequence. In this work, we adopt the multi-scale

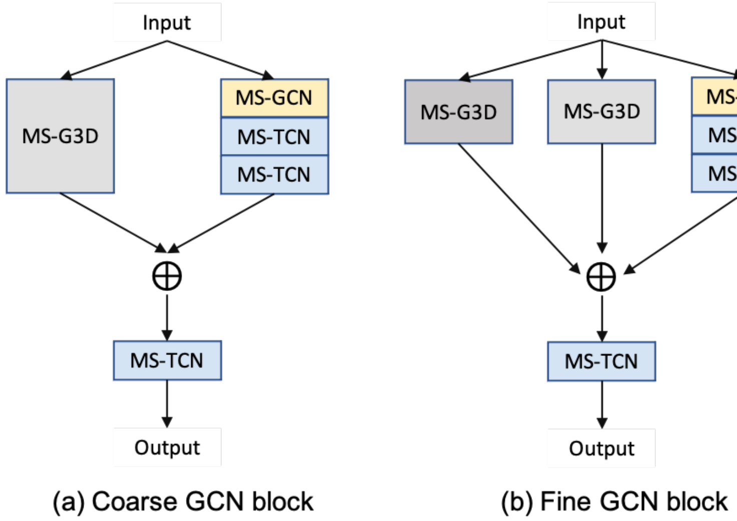


Figure 3.3: Basic blocks in coarse head(left) and fine head(right). We simplify the GCN blocks in two directions. In coarse block, we reduce the G3D component with the largest window size to reduce temporal modelling; in fine block, we reduce the convolution kernels though channel dimensions to 1/2 of coarse block.

spatial-temporal graph convolution block (STGC-block) [41], which has proven effective in representing long-range spatial and temporal context of the skeletal data.

Specifically, given an input sequence $\{z_{t,n} \in R^d | 1 \leq t \leq T, 1 \leq n \leq N; t, n \in Z\}$, where $d \in \{2, 3\}$ indicates the dimension of joint locations, the output of the backbone module can be defined as:

$$\mathcal{X}_{back} = \{x_{t,n}^{(back)} \in R^{\mathcal{C}_{back}} | 1 \leq t \leq T, 1 \leq n \leq N; t, n \in Z\}, \tag{3.3}$$

where $\mathcal{C}_{back}$ is the channel dimension of the output feature, and $x_{t,n}^{(back)}$ indicates the representation of a specific joint n at frame t.

3.2.4 Dual-head Module

To capture the motion patterns with inherently variable temporal resolutions, we develop a multi-granular spatio-temporal graph network. In contrast to prior work relying on shared representations, we adopt a dual-head network to simultaneously extract motion patterns from coarse and fine levels. Given the backbone features $\mathcal{X}_{back}$, below we introduce the detailed structure of our coarse head and fine head.

3.2.4.1 Coarse Head

Our coarse head extracts features at a low temporal resolution, aiming to capture coarse grained motion contexts. In the coarse head, a subsampling layer is first adopted to downsample the feature map at the temporal dimension. Concretely, given the backbone features $\mathcal{X}_{back}$ with T frames, we uniformly sample T/α nodes in the temporal dimension:

$$\mathcal{X}_{coar} = \mathcal{F}_{subs}(\mathcal{X}_{back}), \tag{3.4}$$

where $\mathcal{F}_{subs}$ and $\alpha \in Z$ denote the subsampling function and subsampling rate respectively. $\mathcal{X}_{coar} = \{x_{t,n}^{(coar)} \in R^{\mathcal{C}_{coar}} | 1 \leq t \leq T/\alpha, 1 \leq n \leq N; t, n \in Z\}$ represents the initial feature maps of coarse head.

Subsequently, we introduce a coarse GCN block, denoted by $\mathcal{G}_{coar}$, which consists of two parallel paths formed by a MS-G3D and stacking of multiple MS-GCN and MS-TCN respectively, followed by a MS-TCN fusion block. The detailed structures of coarse GCN block is shown in Figure 3.3 (a). The coarse GCN block is used to compute the final coarse feature representation as follows:

$$\widetilde{\mathcal{X}_{coar}} = \mathcal{G}_{coar}(\mathcal{X}_{coar}) \tag{3.5}$$

where $\widetilde{\mathcal{X}_{coar}}$ is the output feature set.

3.2.4.2 Fine Head

Our fine head extracts features at a high temporal resolution and encode more fine-grained motion contexts. In the fine head, an embedding function $\mathcal{F}_{embed}$ (i.e., 1 × 1 convolution layer) will be applied at the beginning to reduce the feature dimensions. In this way, the output features of the backbone module $\mathcal{X}_{back}$ will be projected into the new feature space $\mathcal{X}_{fine}$:

$$\mathcal{X}_{fine} = \mathcal{F}_{embed}(\mathcal{X}_{back}), \tag{3.6}$$

where $\mathcal{X}_{fine} = \{x_{t,n}^{(fine)} \in R^{\mathcal{C}_{fine}} | 1 \leq t \leq T, 1 \leq n \leq N; t, n \in Z\}$. Then we introduce a fine GCN block, denoted as $\mathcal{G}_{fine}$, to extract fine-grained temporal features as below,

$$\widetilde{\mathcal{X}_{fine}} = \mathcal{G}_{fine}(\mathcal{X}_{fine}). \tag{3.7}$$

The fine GCN block consists of three parallel branches formed by two MS-G3D and stacking of multiple MS-GCN and MS-TCN respectively, followed by a MS-TCN fusion block.The detailed structures of fine GCN block is shown in Figure 3.3 (b).

3.2.4.3 Block Simplification

The proposed dual-head network, a novel structure based on the divide-and-conquer strategy, can effectively extract motion patterns of different levels of granularity in the temporal domain. Note that, due to the downsampling operation, the temporal receptive field has already been expanded. As a result, GCN blocks in such structure naturally can be simplified, such as using smaller temporal convolution kernels. Specifically, in the coarse head, we remove G3D component with the largest window size in the STGC-block to reduce temporal modeling. In the fine head, as the original temporal resolution data already maintains rich temporal contexts, we then reduce the channel dimensions for efficient modelling. The detailed structures of coarse GCN block and fine GCN block are shown in Figure 3.3.

3.2.5 Cross-head Attention Module

To better fuse the representations at different temporal resolutions, we introduce a cross-head attention module to enable communication between two head branches. Such communication can mutually enhance the representations encoded by both head branches.

3.2.5.1 Communication Strategy

The detailed workflow of the proposed communication strategy can be found in Figure 3.2. Since the unsampled frames intrinsically contain more elaborate motion patterns than the downsampled frames, the granular temporal information will be initially transmitted from the fine head to the coarse head. Taking the first temporal attention block as an example, the output of the Fine block will go through an attention block similar to the SE network [26]. The generated attention serves as an interactive message, and then the correlation with the coarse temporal feature is obtained through element-wise matrix multiplication. The correlated feature will be finally fused with the coarse temporal feature and fed into the coarse block for the upcoming propagation. The spatial attention holds a similar structure as the temporal attention, but the order of input features is different. The detailed calculation of these two attention will be introduced below and the structure is shown in Figure 3.4.

3.2.5.2 Temporal Attention

The temporal attention block takes the output of the fine block as the input and can extract the temporal patterns with high similarity from the distant frames to

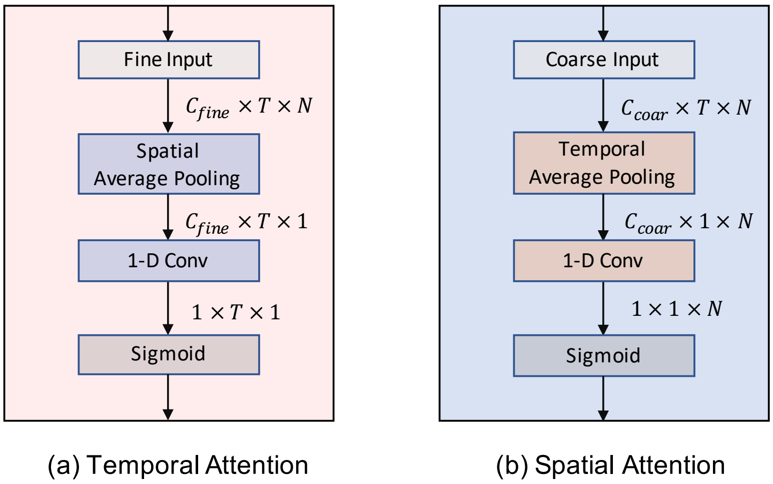


Figure 3.4: An illustration of our spatial and temporal attention block.

the greatest extent because it retains the complete frame rate of the input sequence. The features learned from this can fully reflect the importance of the frame level and lead the coarse-level reasoning in an efficient way of communication. Formally, it can be denoted as:

$$\theta_{te} = \sigma(\mathbf{W_{te}}(AvgPool_{sp}(\mathcal{X}_{fine}))), \tag{3.8}$$

where $\mathcal{X}_{fine}$ is the feature map in fine head, $AvgPool_{sp}$ denotes the average pool in spatial dimension, $\mathbf{W_{te}}$ indicates a 1D convolution with a large kernel size to capture large temporal receptive field. σ indicates sigmoid activation function, $\theta_{te} \in R^{1\times T\times 1}$ is the estimated temporal attention. The attention is used to re-weight coarse features $\mathcal{X}_{coar}$ in a residual manner:

$$\hat{\mathcal{X}}_{coar} = \theta_{te} \cdot \mathcal{X}_{coar} + \mathcal{X}_{coar}, \tag{3.9}$$

where $\cdot$ indicates the element-wise matrix multiplication with shape alignment.

3.2.5.3 Spatial Attention

Our fine head aims to extract motion patterns from subtle temporal movements. To promote such fine-grained representation learning, we then utilise the spatial attention to highlight important joints across frames. Compared with fine-head itself, our coarse head extracts features in lower temporal resolution and can easily learn high-level abstractions in a holistic view. We therefore take the advantage of such holistic representation from coarse head to estimate the spatial attention. Formally,

$$\theta_{sp} = \sigma(\mathbf{W_{sp}}(AvgPool_{te}(\mathcal{X}_{coar}))), \tag{3.10}$$

where $AvgPool_{te}$ indicates the average pooling at temporal dimension, $\mathbf{W_{sp}}$ indicates a 1D convolution layer, $\theta_{sp} \in R^{1\times1\times N}$ is the joint level attention, which is used to re-weight fine features in a residual manner:

$$\hat{\mathcal{X}}_{fine} = \theta_{sp} \cdot \mathcal{X}_{fine} + \mathcal{X}_{fine}, \tag{3.11}$$

As shown in Figure 3.2, our temporal attention and spatial attention are alternately predicted to enhance temporal and spatial features of both heads.

3.2.6 Fusion Module

The proposed network has two heads, responsible for the different granularities of temporal reasoning. We utilise the score level fusion to combine the information of both head and facilitate the final prediction. For simplicity, we denote the outputs of coarse head and fine head as $\widetilde{\mathcal{X}_{coar}}$ and $\widetilde{\mathcal{X}_{fine}}$, and then each head is attached with a global average pooling (GAP) layer, a fully connected layer combined with a SoftMax function to predict a classification score s_{coar} and s_{fine}:

$$s_{coar} = SoftMax(\mathbf{W_{coar}}(AvgPool(\widetilde{\mathcal{X}_{coar}}))), \tag{3.12}$$

$$s_{fine} = SoftMax(\mathbf{W_{fine}}(AvgPool(\widetilde{\mathcal{X}_{fine}}))). \tag{3.13}$$

where $s_{coar}, s_{fine} \in R^a$ indicate the estimated probability of a classes in the dataset, $AvgPool$ is a GAP operation conducted on the spatial and temporal dimensions, $\mathbf{W_{coar}}$ and $\mathbf{W_{fine}}$ are fully connected layeres. Then, the final prediction can be achieved by fusion of these two scores:

$$s = \mu \cdot s_{coar} + (1 - \mu) \cdot s_{fine}, \tag{3.14}$$

where μ is a hyper-parameter used to examine the importance between two heads. During our implementation, we empirically set $\mu = 0.5$. More configurations can be explored in future work. During training, we also supervise s_{coar} and s_{fine} with two cross entropy losses and sums them with the same weight μ.

3.2.7 Multi-modality Ensemble

Following prior works [58, 57, 11, 9], we generate four modalities for each skeleton sequence, they are the joint, bone, joint motion and bone motion. Specifically, the joint modality is derived from the raw position of each joints. The bone modality is produced by the offset between two adjacent joints in a predefined human structure. The joint motion modality and bone motion modality are generated by the offsets of the joint data and bone data in two consecutive frames. Our final model is a 4-stream network that sums the softmax scores of each modality to make predictions.

3.3 Experiments

We evaluate our proposed method[1] on the NTU RGB+D 60 [53], NTU RGB+D 120[37] and Kinetics-Skeleton [29] datasets. Below, we first introduce the implementation details of our method. Next, we introduce the evaluation protocols and results respectively. Finally, in the ablation study section, we conduct extensive experiments to verify the effectiveness of the proposed modules. Besides, we also show some qualitative results for analysis.

3.3.1 Implementation Details

All experiments are conducted using PyTorch. The cross-entropy loss is used as the loss function, and Stochastic Gradient Descent (SGD) with Nesterov Momentum (0.9) is used for optimization. The downsample ratio of coarse head is set to $\beta = 2$. The fusion weight μ of the two heads is set to 0.5.

The preprocessing of the NTU-RGB+D 60&120 dataset is in line with previous work [41]. Specifically, the second body is padded with 0 if the number of bodies in the sample is less than 2. The maximum number of frames in each sample is 300. For samples with less than 300 frames, the existing frames will be repeatedly stacked until the final number of frames reaches 300. During training, the batch size is set to 64 and the weight decay is set to 0.0005. The initial learning rate is set to 0.1, and then divided by 10 in the 40^{th} epoch and 60^{th} epoch. The training process ends in the 80^{th} epoch. The experimental setup of the Kinetics-Skeleton dataset is consistent with previous works [84, 58]. We set the batch size to 128 and weight decay to 0.0001. The learning rate is set to 0.1 and is divided by 10 in the 45^{th} epoch and the 55^{th} epoch. The model is trained for a total of 80 epochs.

[1] Code available at https://github.com/tailin1009/DualHead-Network

3.3.2 Experiments on the NTU RGB+D 60 Dataset

The NTU RGB+D dataset has two standard evaluation protocols [53]. (1) Cross-Subject (X-sub): half of the 40 subjects are used for training, and the rest are kept for testing. (2) Cross-View (X-view): 2/3 of the viewpoints are used for training, and 1/3 unseen viewpoints are left out for testing. Following the previous works [58, 57, 11, 9], we generate four modalities data (joint, bone, joint motion and bone motion) and report the results of joint stream (Js), bone stream (Bs), joint-bone two-stream fusion (2s), and four-stream fusion (4s). Experimental results are shown in Table 3.1

Specifically, as shown in Table 3.1, our method achieves state-of-the-art performance **91.7** on **NTU RGB+D 60** X-sub setting with only joint-bone two-stream fusion. The final four-stream model further improves the performance to **92.0**.

3.3.3 Experiments on the NTU RGB+D 120 Dataset

There are two evaluation protocols are provided in NTU RGB+D 120 [37], (1) Cross-Subject (X-sub) that splits 106 subjects into training and test sets, where each set contains 53 subjects; (2) Cross-setup (X-set) that splits the collected samples by the setup IDs (i.e., even setup IDs for training and odds setup IDs for testing). Similarly, we generate four modalities data and experimental results are shown in Table 3.2

It is worth noting that on X-sub and X-view setting, our model achieve the best performance which demonstrates the effectiveness of our proposed dual-head graph network design.

Methods	NTU RGB+D 60	
	X-sub	X-view
GCA-LSTM [39]	74.4	82.8
VA-LSTM [88]	79.4	87.6
TCN [31]	74.3	83.1
Clips+CNN+MTLN [30]	79.6	84.8
ST-GCN [84]	81.5	88.3
SR-TSL [61]	84.8	92.4
STGR-GCN [33]	86.9	92.3
AS-GCN [35]	86.8	94.2
AGCN [58]	88.5	95.1
DGNN [55]	89.9	96.1
GR-GCN [19]	87.5	94.3
SGN [89]	89.0	94.5
MS-AAGCN [57]	90.0	96.2
NAS-GCN [48]	89.4	95.7
Decouple-GCN [10]	90.8	**96.6**
Shift-GCN [11]	89.7	**96.6**
STIGCN [27]	90.1	96.1
ResGCN [65]	90.9	96.0
Dynamic-GCN [86]	91.5	96.0
MS-G3D [41]	91.5	96.2
MST-GCN [9]	91.5	**96.6**
DualHead-Net (Ours)	**92.0**	**96.6**

Table 3.1: Comparison of the Top-1 accuracy (%) with the state-of-the-art methods on the NTU RGB+D 60 dataset.

Methods	NTU RGB+D 120	
	X-sub	X-set
ST-LSTM [38]	55.7	57.9
Clips+CNN+MTLN [30]	62.2	61.8
SkeMotion[4]	67.7	66.9
TSRJI [3]	67.9	62.8
ST-GCN [84]	70.7	73.2
AGCN [58]	82.5	84.2
Decouple-GCN [10]	86.5	88.1
Shift-GCN [11]	85.9	87.6
MS-G3D [41]	86.9	88.4
ResGCN [65]	87.3	88.3
Dynamic-GCN [86]	87.3	88.6
MST-GCN [9]	87.5	88.8
DualHead-Net (Ours)	**88.2**	**89.3**

Table 3.2: Comparison of the Top-1 accuracy (%) with the state-of-the-art methods on the NTU RGB+D 120 dataset.

Methods	Kinetics-Skeleton	
	Top-1	Top-5
PA-LSTM [53]	16.4	35.3
TCN [31]	20.3	40.0
ST-GCN [84]	30.7	52.8
AS-GCN [35]	34.8	56.5
AGCN [58]	36.1	58.7
DGNN [55]	36.9	59.6
NAS-GCN [48]	37.1	60.1
MS-G3D [41]	38.0	60.9
STIGCN [27]	37.9	60.8
Dynamic-GCN [86]	37.9	**61.3**
MST-GCN [9]	38.1	60.8
DualHead-Net (Ours)	**38.4**	**61.3**

Table 3.3: Comparison of the Top-1 accuracy (%) and Top-5 accuracy (%) with the state-of-the-art methods on the Kinetics Skeleton dataset.

3.3.4 Experiments on the Kinetics-Skeleton Dataset

We follow the same experimental protocol provided by [85, 58]. There are 240,436 samples for training and 19,794 samples for testing. We also generate four modalities data and the results are reported in Table 3.3

On this largest dataset, our four-stream model outperforms prior work[9] by **0.3** in terms of the top-1 accuracy.

3.3.5 Ablation Studies

In this subsection, we perform ablation studies to evaluate the effectiveness of our proposed modules and attention mechanism. Except for the experiments in **Incremental ablation study**, all the following experiments are performed by modifying the target component based on the full model. Unless stated, all the experiments are conducted under X-sub setting of NTU-RGBD 60 dataset, using only joint stream.

Method	Params	Dual head	TA	SA	Acc
Baseline	3.2M	-	-	-	89.4
Ours	3.0M	✓	-	-	89.9
	3.0M	✓	✓	-	90.2
	3.0M	✓	✓	✓	**90.3**

Table 3.4: Ablation study of different modules on NTU RGB+D 60 X-sub setting, evaluated with only joint stream. 'TA' indicates cross head temporal attention, 'SA' indicates cross head spatial attention. Note that both attention block introduces parameters fewer than 0.1M.

Channel reduction	Params	Acc
Baseline (MS-G3D[41])	3.2M	89.4
No reduction	4.9M	90.5
Reduce channels to 1/2 (proposed)	3.0M	90.3
Reduce channels to 1/4	2.5M	89.7

Table 3.5: Different reduction rate of feature channels in fine head.

3.3.5.1 Incremental ablation study

We first evaluate the proposed dual head module, temporal and spatial attention mechanism in an incremental manner. We start from our baseline network, MS-G3D[41]. We add our proposed modules one-by-one. The results are shown in Table 3.4. Our **dual head structure** improves the performance from 89.4 to 89.9, which demonstrates the effectiveness such divide-and-conquer structure. Note that our dual head structure keeps less parameters than baseline network due to block simplification. Adding attentions further improves the performance.

3.3.5.2 Model simplification

Due to the robust modelling ability of dual head structure, we argue that the GCN blocks in both heads can be simplified for balancing the model performance and complexity. The simplification strategies are investigated and discussed below.

Simplification of fine head. We reduce the channel dimensions of feature maps in fine head due to the rich temporal information contained. In Table 3.5, we can

G3D pathways	Params	Acc
w/o G3D(factorized)	2.4M	90.0
1 G3D	3.0M	90.3
2 G3D	3.9M	90.0

Table 3.6: Comparison of different number of G3D pathways in coarse block.

observe that, without channel reduction, the model achieves an accuracy of 90.5, but with 4.9M parameters. By reducing the channels to 1/2, the accuracy only drops to 90.3, while the parameters are significantly reduced to 3.0M, which is smaller than the baseline network(3.2M). However, as we further reduce the parameters to 2.5M, the accuracy will drop to 87.4. To balance the model complexity and performance, we choose a reduction rate of 2(reduce to 1/2) in our final model.

Simplification of coarse head. We report the ablation study of different G3D pathways in our coarse block in Table 3.6. We can observe that utilising one G3D component is able to sufficiently capture the coarse grained motion contexts. Increasing G3D components to 2 will drop the performance a little, we believe it's because the coarse grained motion contexts are easy to capture and large models will turn to over-fitting.

3.3.5.3 Temporal subsample rate of coarse head.

We model the coarse grained temporal information in our coarse head, which is generated by subsampling the features in temporal dimension. We hence perform an ablation study of temporal subsampling rate of coarse head, shown in Table 3.7. Since proposed method utilises a subsampling rate of 2 (reduce to 1/2). We can observe that, without subsampling, the coarse head also takes fine grained features, and the performance will drop from 90.3 to 89.8, demonstrating the importance of coarse grained temporal context. However, as we further enlarge the subsample rate to 4 (reduce to 1/4), the performance will drop to 90.1. This implies that over subsampling will lose the important frames and hence drop the performance.

Temporal subsample	Acc
subsample all frames	89.8
subsample 1/2 frames	90.3
subsample 1/4 frames	90.1

Table 3.7: Ablation study of temporal subsample rate in coarse head.

Attention type	Mechanism	Acc
Temporal attention	cross head attention	90.3
	self learned attention	90.0
Spatial attention	cross head attention	90.3
	self learned attention	90.1

Table 3.8: Comparison of cross head attention and self learned attention, which is estimated by the features of their own heads.

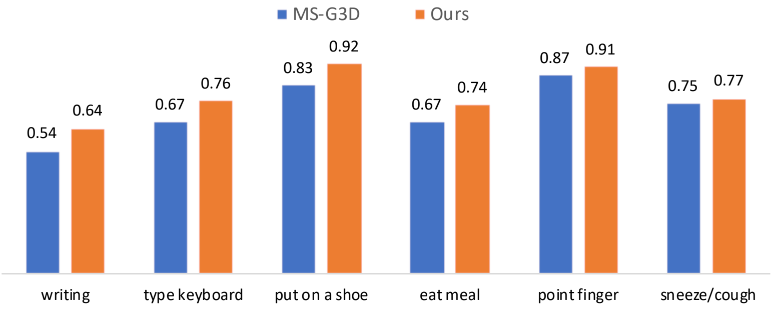


Figure 3.5: Comparison of classification accuracy of 6 difficult action classes on NTU RGB+D X-Sub Setting.

3.3.5.4 Cross head attention

We also perform ablation studies on our proposed cross head temporal attention and spatial attention. Our proposed cross head temporal attention passes fine grained temporal context from fine head to coarse head and re-weight the coarse features. As shown in Table 3.8, such cross head temporal attention mechanism outperforms the attention estimated by coarse features themselves. Similar in Table 3.8, the cross head spatial attention outperforms the spatial attention estimated by fine features, denoted by ‘self learned attention’.

3.3.6 Visualisation

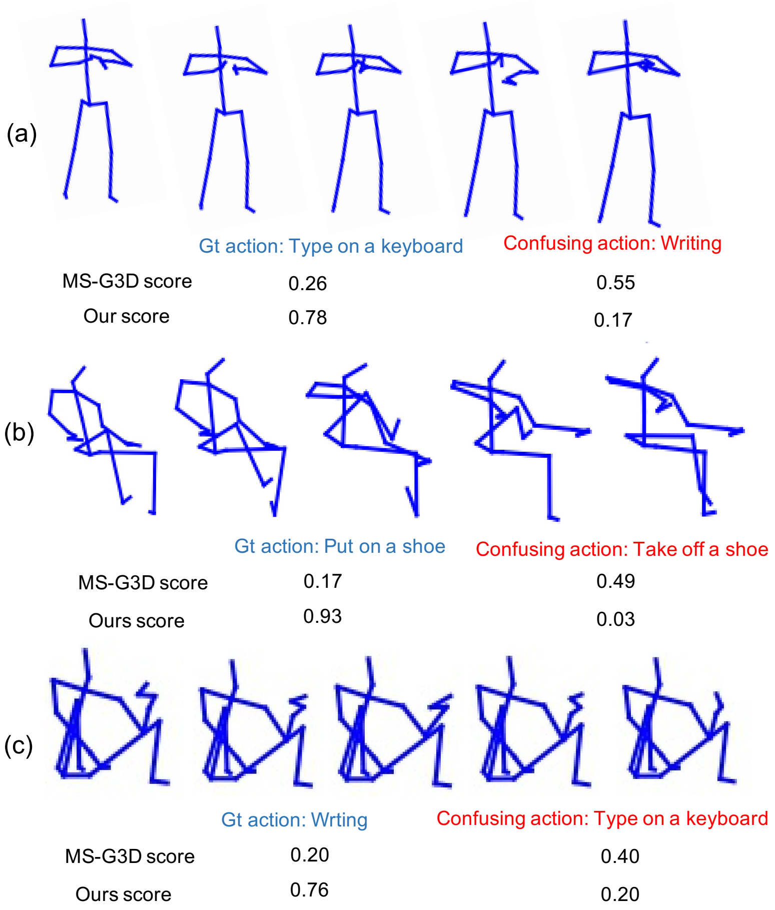


Figure 3.6: Skeleton samples and the prediction scores of MS-G3D and our method. GT action and confusing action are shown in blue and red colour. Our method improves the prediction scores of those fine-grained actions.

We show some quantitative results in Figure.3.5 and some qualitative results in Figure.3.6. We can observe that our method improves those action classes that are in fine-grained label space, which requires both coarse grained and fine grained motion information to be recognised.

3.4 Chapter Summary

In this chapter, we propose a novel multi-granular spatio-temporal graph network for skeleton-based action recognition, which aims to jointly capture coarse- and fine-grained motion patterns in an efficient way. To achieve this, we design a dual-head graph network structure to extract features at two spatio-temporal resolutions with two interleaved branches. We introduce a compact architecture for the coarse head and fine head to effectively capture spatio-temporal patterns in different granularities. Furthermore, we propose a cross attention mechanism to facilitate multi-granular information communication in two heads. As a result, our network is able to achieve new state-of-the-art on three public benchmarks,namely NTU RGB+D 60, NTU RGB+D 120 and Kinetics-Skeleton, demonstrating the effectiveness of our proposed dual-head graph network.

Chapter 4

Long Short-term Spatio-temporal Aggregation Network for Skeleton-based Action Recognition

In the previous chapter, a dual-head structure is introduced for skeleton-based action recognition from the perspective of motion granularity. However, modelling varied spatio-temporal dependencies is also the key to recognising human actions in skeleton sequences. In this chapter, from the perspective of distant joints dependency modelling, we propose a novel long short-term feature aggregation strategy which can effectively and efficiently model both long-and-short range dependencies in the space and time domain.

4.1 Introduction

Skeleton-based action recognition is a challenging task due to the lack of context information compared with RGB video based action recognition. In particular, modelling long range dependencies in spatial and temporal dimensions are difficult due to the flexible configuration between different semantic parts, as well as complex movement patterns in time domain.

Recently, substantial efforts have been devoted to extracting features from skeletal data. Despite the significant improvements in performance, there still exists some limitations. Specifically, both recurrent and convolutional operations are based on local neighbours in the space or time domain. The capture of long-range dependencies only can be achieved by repeatedly performing these operations and gradually propagating signals through the data and hence is inefficient. Directly modelling the distant joints relations and long-range temporal information is essential for distinguishing various actions. Recent work MS-G3D [42] proposed a disentangled and unified spatial-temporal graph convolution strategy to model the long range dependencies in a multi-scale manner. However, the proposed G3D module highly relies on constructing multiple pathways and hence leads to a complex model architecture.

In this chapter, to model both short-range and long-range joints relations in spatial domain, short-term and long-term joints dynamics in temporal domain, we propose a novel long short-term spatio-temporal aggregation network (*LSTA-Net*) to model such various dependencies. Specifically, we propose a multi-scale decentralised aggregation (MSDA) module which is capable of extracting multi-scale spatial features. For a given graph node in spatial dimension, The MSDA is used to do multi-scale disentangling, decentralised normalisation and then allows multiple information exchanges between neighbouring nodes and distant nodes. For long-term temporal modelling, inspired by [18, 87], we propose an attention-enhanced temporal pyramid aggregation (ATPA) module to model long-range temporal dynamics by deforming the ordinary convolution layer into a set of sub-convolutions and formulating them with a pyramid-like/hierarchical structure allowing residual-like connections between adjacent subsets. Besides, our attention module, namely maximum-response attention (MA) module is proposed for improving the local cross-channel interaction via multiple high efficient 1-D convolutions as inspired by [80]. In our framework architecture, these modules are complementary and stacked together which endows the network with both long&short spatial and temporal modelling abilities. As shown in Figure 4.1 our model is based on hierarchical convolutional blocks, unlike LSTMs, which process data over time using a series of gates (input, output and forget) to regulate the flow of information, our model employs stacked convolutional layers to extract spatial and temporal features at different resolutions which is novel.

The contributions of this chapter are summarised as follows: (1) A novel multi-scale spatial decentralised aggregation module for both local and non-local spatial joints modelling (2) An attention-enhanced temporal pyramid aggregation module which can efficiently enlarge the receptive field for short-term and long-term temporal dynamics modelling. (3) The experiments on three public benchmarks show that our proposed method achieves remarkable performances with fewer parameters, validating the effectiveness of our method.

The rest of this chapter is organised as follow. In Section 4.2, we introduce the proposed LSTA-Net. In Section 4.3, we provide the experimental results Finally, we conclude the chapter in Section 4.4.

4.2 Long Short-term Aggregation Network

4.2.1 Multi-scale Decentralised Spatial Aggregation

A graph is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices and $\mathcal{E} \subseteq (\mathcal{V} \times \mathcal{V})$ is the set of edges. The relationship of the graph is represented by the graph adjacency matrix$\mathbf{A} \in R^{N \times N}$ with a 1 or 0 in position (v_i, v_j)according to whether $v_i$and v_j are adjacent or not, where N denotes the number of vertexes. $\mathbf{A}$ is a symmetric

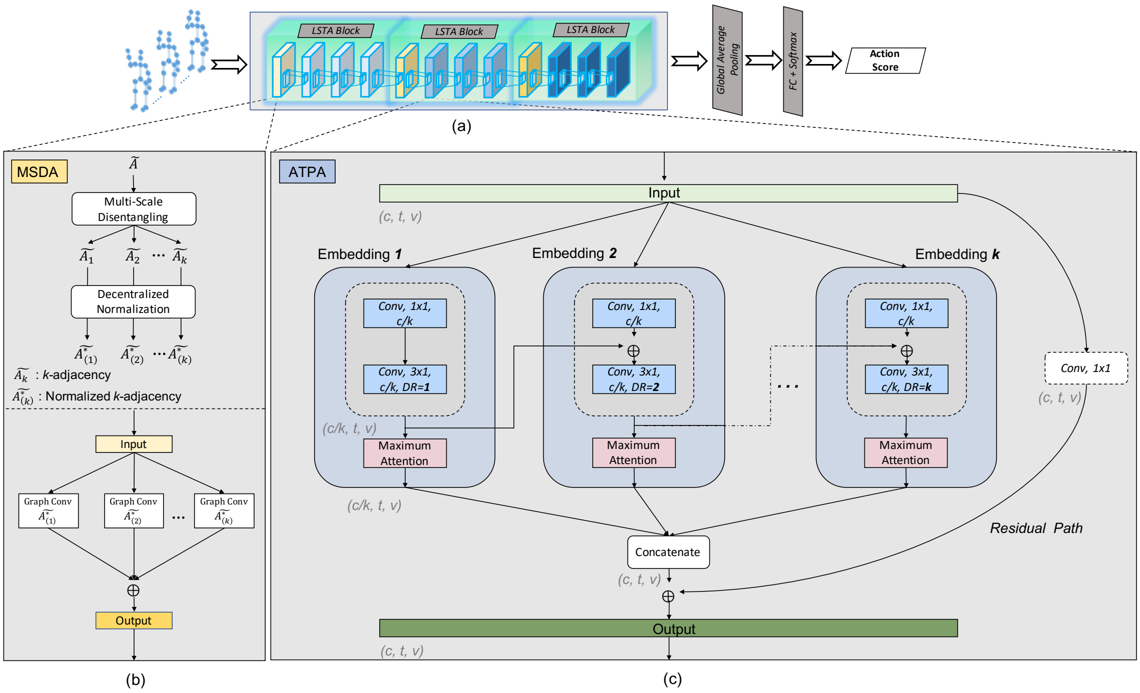

LSTA Block
LSTA Block
LSTA Block
Global Average Pooling
FC + Softmax
Action Score
(a)
MSDA
$\widetilde{A}$
Multi-Scale Disentangling
$\widetilde{A_1}$ $\widetilde{A_2}$ ··· $\widetilde{A_k}$
Decentralized Normalization
$\widetilde{A^*_{(1)}}$ $\widetilde{A^*_{(2)}}$ ··· $\widetilde{A^*_{(k)}}$
$\widetilde{A_k}$: k-adjacency
$\widetilde{A^*_{(k)}}$: Normalized k-adjacency
Input
Graph Conv $\widetilde{A^*_{(1)}}$
Graph Conv $\widetilde{A^*_{(2)}}$
Graph Conv $\widetilde{A^*_{(k)}}$
Output
(b)
ATPA
Input
(c, t, v)
Embedding 1
Embedding 2
Embedding k
Conv, 1x1, c/k
Conv, 3x1, c/k, DR=1
Conv, 3x1, c/k, DR=2
Conv, 3x1, c/k, DR=k
(c/k, t, v)
Maximum Attention
(c/k, t, v)
Conv, 1x1
(c, t, v)
Residual Path
Concatenate
(c, t, v)
Output
(c, t, v)
(c)

Figure 4.1: An illustration of the proposed LSTA-Net, which consists of three LSTA blocks totally. Each LSTA block is comprised of one MSDA module and three A-TPA modules.

matrix while $\mathcal{G}$ is an undirected graph. Human graph sequences contain a set of node features $\mathcal{X} = \{x_t^n | 1 \leq n \leq N, 1 \leq t \leq T; n, t \in Z\}$ represented as a feature tensor $\mathbf{X} \in R^{N \times T \times C}$, where $x_t^n = \mathbf{X}_t^n$ denotes the C dimensional features of the node v_n at time t. Let $\mathbf{W} \in R^{C_{\text{in}} \times C_{\text{out}}}$ be the learnable transformation matrix. Then, the spatial convolution on a given graph can be implemented similar to the convolution on a regular grid graph like an RGB image. Formally, the multi-scale GCN can be expressed as:

$$\mathbf{X}_t^{\text{spat}} = \sigma \left(\sum_{k=0}^{K} \widetilde{\mathbf{A}_k} \mathbf{X}_t \mathbf{W}_k \right), \tag{4.1}$$

where $\mathbf{X}_t \in R^{N \times C_{\text{in}}}$ and $\mathbf{X}_t^{\text{spat}} \in R^{N \times C_{\text{out}}}$ denote the input and output skeleton features for t-th frame, respectively. K indicates the number of scales of the graph to be aggregated. $\sigma(\cdot)$ is the activation function. $\tilde{\mathbf{A}}$ is the normalised adjacency matrix [32, 35] and can be obtained by: $\tilde{\mathbf{A}} = \hat{\mathbf{D}}^{-\frac{1}{2}} \hat{\mathbf{A}} \hat{\mathbf{D}}^{-\frac{1}{2}}$, where $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is the adjacency matrix including the nodes of the self-loop graph and $\hat{\mathbf{D}}$ is the degree matrix of $\mathbf{A}$.

The term of $\widetilde{\mathbf{A}_k}\mathbf{X}_t$ in Eq.(4.1) describes a weighted average feature that is based on the k-order neighbourhood of the selected graph node. However, conventional multi-scale aggregation based on cyclic k-hop neighbours will make low-order neighbours overweight and vice versa. Similar to [42], we propose a novel decentralised multi-scale disentangling scheme to tackle with the bias weighting problem by introducing multiple scales rather than neighbouring scales. Concretely, the elements of k-adjacency matrix $\widetilde{\mathbf{A}_{(k)}}$ are first assigned the value 1 if v_i and v_j are adjacent or their shortest distance is equal to k (i.e., $d(v_i, v_j) = k$). Then, for those elements whose shortest distance lower/shorter than k (i.e., $d(v_i, v_j) < k$), are assigned with a scale-adaptive value $\frac{d(v_i,v_j)}{k}$. Finally, the rest of elements are set to 0. Thus, $\widetilde{\mathbf{A}_{(k)}}$ can be directly obtained by calculating the residuals of the matrix powers of current graph scale and the mean of previous graph scales:

$$\widetilde{\mathbf{A}^*_{(k)}} = \mathbf{I} + \mathbb{1}\left(\widetilde{\mathbf{A}_k} \geq 1\right) - \mathbb{1}\left((\frac{1}{k}\sum_{n=0}^{k-1}\widetilde{\mathbf{A}_n}) \geq 1\right). \tag{4.2}$$

where $\widetilde{\mathbf{A}^*_{(1)}} = \widetilde{\mathbf{A}}$ and $\widetilde{\mathbf{A}^*_{(0)}} = \mathbf{I}$. Then, the spatial decentralized multi-scale aggregation can be rewritten by substituting $\widetilde{\mathbf{A}_k}$ with $\widetilde{\mathbf{A}^*_{(k)}}$ in Eq. (4.1):

$$\mathbf{X}_t^{\text{spat}} = \sigma\left(\sum_{k=0}^{K}\widetilde{\mathbf{A}^*_{(k)}}\mathbf{X}_t\mathbf{W}_{(k)}\right), \tag{4.3}$$

4.2.2 Attention-enhanced Temporal Pyramid Aggregation

Temporal modelling is also essential for skeleton-based action recognition. However, the capture of long-range information is rarely investigated in previous works. Modelling Long-range temporal dependence can alleviate the ambiguous problem between actions by introducing discriminative temporal information and can provide more robust temporal context to help the model learn better representations from spatial-temporal domain. Many existing works [85, 59, 33, 35, 56] adopted the a temporal convolution with a fixed kernel size to process neighbouring frames at a time and then model long-range temporal dependencies by repeated stacking. However, useful features from distant frames have already been weakened after a large number of local convolution operations. [42] expands the temporal receptive field by composing a large number of local operations but fail to well control the balance between performance and parameters. To capture the long-range temporal dependencies in an efficient way and inspired by Res2Net-like architectures [18, 87], we propose the Temporal Pyramid Aggregation module (TPA) which can divide the convolution operation of the input features into a group of subsets. Concatenating all fragments provides additional benefits in terms of multi-level and diverse temporal information. By using all fragments, we preserve not only the broad long-term dependencies but also the fine-grained short-term variations that might be smoothed out if only the last fragment were used. This approach helps the model to maintain rich temporal features at different levels, enhancing the overall capacity to capture subtle motion patterns across time. We transfer such image-based method to skeleton-based action recognition for temporal modelling. Furthermore, to extract the semantic dependencies of skeletal features arbitrarily and efficiently, and inspired by [80], we propose a novel Maximum-response Attention Module that can be readily employed in standard CNN-based architectures.

The architecture of the proposed Attention-enhanced Temporal Pyramid Aggregation module is shown in Figure 4.1(c). Consequently, the model effectively expand the equivalent receptive field of the temporal dimension without introducing additional parameters and time-consuming operations as well as can capture the useful intrinsic correlations of human skeletons.

4.2.2.1 Temporal Pyramid Aggregation

As shown in Figure 4.1(c), given the aggregated spatial graph feature $\mathbf{X}_t^{\text{spat}}$, it generates multiple fragments according to the local temporal convolution. The channel dimension of each fragment thus becomes C/s, where s is set to 6 empirically. These fragments are formulated as a hierarchical residual architecture and thus can be hierarchically processed by temporal convolutions with gradually increasing dilation rates. Formally, this process can be written as:

$$\begin{aligned} &\mathbf{X}_s^{\text{temp}} = conv_{\text{temp}} * \mathbf{X}_s, && s = 1, \\ &\mathbf{X}_s^{\text{temp}} = conv_{\text{temp}} * \left(\mathbf{X}_s + \mathbf{X}_{s-1}^{o}\right), && s = 2, 3, 4, 5, 6, \end{aligned} \tag{4.4}$$

where $\mathbf{X}_s^{\text{temp}}$ is the output of s-th fragment. $conv_{\text{temp}}$ denotes the 3 × 1 temporal sub-convolution. For simplicity, we omit the subscript t of $\mathbf{X}_{ts}$ in Eq.(4.4).

The above operations endow the different fragments with different receptive fields in temporal dimension.The final output is easily obtained by concatenating outputs of multiple fragments and can be denoted as:

$$\mathbf{X}^{\text{temp}} = [\mathbf{X}_1^{\text{temp}}; \mathbf{X}_2^{\text{temp}}; \mathbf{X}_3^{\text{temp}}; \mathbf{X}_4^{\text{temp}}; \mathbf{X}_5^{\text{temp}}; \mathbf{X}_6^{\text{temp}}]. \tag{4.5}$$

Therefore, the obtained feature $\mathbf{X}^{\text{temp}}$ captures various range of temporal representations while both the short-range and long-range temporal relationships and are well constructed and is theoretically superior to a single local convolution in most existing approaches.

4.2.2.2 Maximum-response Attention Module

As shown in Figure 4.2, given an input feature $\mathbf{X} \in R^{N \times T \times C}$, the channel attention can be computed as:

$$\omega = \tau \left(\mathbf{W} g(\mathbf{X}) \right), \tag{4.6}$$

where $\tau(\cdot)$ is the Sigmoid function and $\mathbf{W}$ is a $C \times C$ transformation matrix. $g(\mathbf{X})$ denotes the function of channel-wise pooling. This can be implemented by the standard $1D$ convolution with kernel size of η for compact representation. Then, we can rewrite Eq.(4.6) in the form of:

$$\omega = \tau \left(conv1D_{\eta} \left(g(\mathbf{X}) \right) \right). \tag{4.7}$$

The above operation only involves the η parameter, which is insignificant compared with the number of parameters of the entire network. However, the $1D$ convolution equipped with a single convolution kernel can only provide a limited receptive field, which leads to insufficient local information interaction. To tackle this problem, we extend the $1D$ convolution to a parallel fashion with different dilation rates. Subsequently, we apply the element-wise maximum operator to obtain the highest response of the input feature to the classifier. We propose to formulate this manipulation in the following manner:

$$\omega_{max} = \max_{\phi \in \Phi} \left(\phi(\omega) \right), \tag{4.8}$$

where $\phi(\omega)$ represents a set of $1D$ convolutions stacking with different dilation rates, and Φ is the total length of attending convolutions. The new features ω_{max} is constructed in such a way not only enhances the interactivity of information, but also makes their output independent of the receptive fields known in advance by the various kernels.

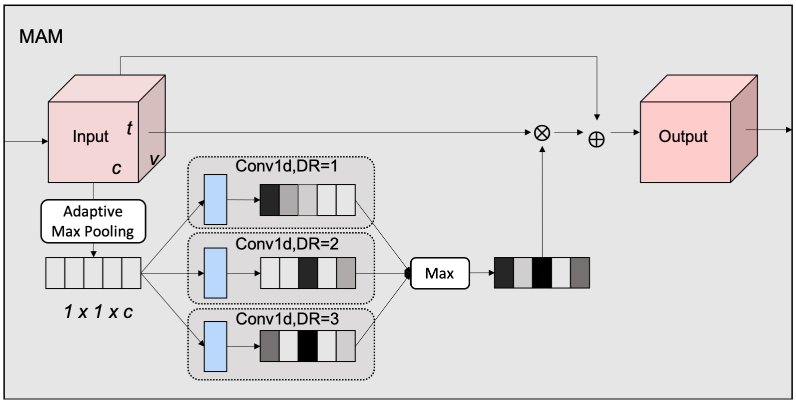


Figure 4.2: An illustration of the proposed MAM Module.

4.2.3 Model Architecture

The overall architecture is shown as Figure 4.1(a), which consist 3 LSTA blocks. Each LSTA block contains a multi-scale spatial decentralised aggregation (MSDA) module and three attention-enhanced temporal pyramid aggregation (ATPA) module to extract spatial and temporal feature alternatively.

The MSDA module is responsible for exploring the semantically related intrinsic connectivity of the disentangled/distant joints in the spatial domain. The ATPA module is capable of modelling the long-term and shot-range temporal relationships over distant frames as well as capturing the cross-channel dependencies. Then, a global average pooling layer and a fully-connected layer together with a softmax layer are adopted to acquire the classification score $\hat{y}$ of the action. Formally, we have:

$$\hat{y} = f_{\mathrm{LSTA}}\left(\widetilde{\mathbf{A}^{*}}; \mathbf{X}^{\mathrm{spat}}_{(1)}, \mathbf{X}^{\mathrm{temp}}_{att(1)}, \ldots, \mathbf{X}^{\mathrm{spat}}_{(L)}, \mathbf{X}^{\mathrm{temp}}_{att(L)}\right), \tag{4.9}$$

where l denotes the LSTA block number, $\mathbf{X}^{\mathrm{spat}}_{(l)}$ and $\mathbf{X}^{\mathrm{temp}}_{att(l)}$ denote the aggregated spatial features and temporal features with the maximum attention modules, respectively. $\widetilde{\mathbf{A}^{*}} = \sum_{k=0}^{K} \widetilde{\mathbf{A}^{*}_{(k)}}$, where $\widetilde{\mathbf{A}^{*}_{(k)}}$ represents the sum of the graph Laplacian matrix $\widehat{\mathbf{A}_{(k)}}$ and its mask $\widehat{\mathbf{A}^{\mathrm{res}}_{(k)}}$. Specifically, $\widehat{\mathbf{A}^{\mathrm{res}}_{(k)}}$ is a learnable, unconstrained graph residual mask that has been employed in [56, 59, 42] to strengthen, weaken, add, or remove edges dynamically.

4.3 Experiments

The proposed method is evaluated on the NTU RGB+D 60 [53], NTU RGB+D 120 [37] and Kinetics-Skeleton [29] datasets. The evaluation protocols are exactly the same as Section 3.3. Below, we first introduce the implementation details of our method. Next, we compare our method with State-of-the-arts methods and report the results respectively. Finally, in the ablation study section, we conduct extensive experiments to verify the effectiveness of the proposed modules. Besides, we also show some visualisation results for analysis.

4.3.1 Implementation Details

The proposed LSTA-Net is implemented by PyTorch toolkit and run on a server with four TitanV GPU. The batch size is set to 64 (16 per worker). The model is trained for 100 epochs with Nesterov momentum (0.9) SGD and the cross-entropy loss. The initial learning rate is set to 0.05 and decayed with a factor of 0.1 at epochs {40,60,80,100}. The weight decay is set to 0.0005 for all experiments. The architecture is the same as the factorised path in [42] but with the different output channels (i.e., 72, 144 and 288 for each LSTA block in sequential). The kernels of adaptive maxpooling layer in MAM module are empirically set to 3 and 5 for A-SDA and A-TPA, respectively. The input skeletal data is padded to T=300 frames by replaying the actions. All sequences are pre-processed with normalisation and translation that employed in [56, 59, 42]. It is worth noting that none of the data augmentation methods is employed in our method.

4.3.2 Comparison with State-of-the-arts

Methods	Param.	NTU RGB+D 60		NTU RGB+D 120	
		X-Sub	X-View	X-Sub	X-Set
PA-LSTM [53]	–	60.7	67.3	25.5	26.3
ST-LSTM [38]	–	69.2	77.7	55.0	57.9
VA-LSTM [88]	–	79.4	87.6	–	–
TCN [31]	–	74.3	83.1	–	–
AGC-LSTM [60]	–	89.2	95.0	–	–
ST-GCN [85]	3.1M	81.5	88.3	70.7	73.2
AS-GCN [35]	-	86.8	94.2	77.9	78.5
AGCN [59]	6.9M	88.5	95.1	82.9	84.9
DGNN [56]	26.2M	89.9	96.1	–	–
NAS-GCN [48]	6.6M	89.4	95.7	–	–
Shift-GCN [12]	2.8M	90.7	96.5	85.9	87.6
MS-G3D [42]	6.4M	91.5	96.2	86.9	88.2
LSTA-Net(ours)	3.1M	91.5	**96.6**	**87.5**	**89.0**

Table 4.1: Comparison of the Top-1 accuracy (%) with the state-of-the-art methods on the NTU RGB+D 60 & 120 datasets. The top part consists of several models without GCN technique, while the middle part contains some GCN-based models. The recent popular GCN-based models are presented with parameter number (million).

Many state-of-the-art methods utilise multi-stream fusion strategies to fuse different modalities data for complementary information combination. To conduct a fair comparison, we adopt the similar multi-stream fusion strategy as [12, 56] , and devise our framework in the three-stream fashion where joint, bone, motion streams are sharing one identical architecture. The initialisation of the ”bone” stream is set to the vector difference of adjacent joints directed away from the center of the human body. Then the ”motion” stream use the temporal difference between adjacent frames of ”joint” or ”bone” as input. Finally, a score-level fusion strategy is applied to obtain the final prediction score.

We compare our full model with other state-of-the-art methods on NTU-RGB+D 60, NTU-RGB+D 120 and Kinetics-Skeleton datasets and the results are shown in Table 4.1 and Table 4.2 under the same setting. On NTU RGB+D 60 dataset, we achieve competitive performance on cross-view and cross-subject benchmarks. For

Methods	Kinetics Skeleton	
	Top-1	Top-5
PA-LSTM [53]	16.4	35.3
TCN [31]	20.3	40.0
ST-GCN [85]	30.7	52.8
AS-GCN [35]	34.8	56.5
AGCN [59]	36.1	58.7
DGNN [56]	36.9	59.6
NAS-GCN [48]	37.1	60.1
MS-G3D [42]	38.0	**60.9**
LSTA-Net (Ours)	**38.1**	60.7

Table 4.2: Comparison of the Top-1 accuracy (%) and Top-5 accuracy (%) with the state-of-the-art methods on the Kinetics Skeleton dataset.

NTU RGB+D 120, our method outperforms other methods on both cross-subject and cross-setup benchmarks. We additionally compare the number of model parameters with several SOTA GCN models in Table 4.1. The result shows that our proposed LSTA-Net is light-weight and keeps the SOTA performance, validating the effectiveness of our proposed factorised long short-term aggregation scheme. For Kinetics-skeleton dataset, our proposed LSTA-Net achieves comparable performance with MS-G3D and outperforms the others. As Kinetics-skeleton dataset has 240,436 training samples, given the fact that our model has only 1/2 parameters as in MS-G3D, the proposed long short-term aggregation strategy is validated to be efficient in such a large dataset.

4.3.3 Ablation Study

In this section, we report the results of our ablation study to validate the effectiveness of our proposed model components or strategies. Unless stated, performance is reported as classification accuracy on the Cross-Subject benchmark of NTU RGB+D 60 dataset using only the joint data.

MSDA module In Table 4.3, we compare the proposed MSDA with the basic adjacency powering method and disentangling [42] method in terms of the number

Spatial Aggregation	Number of Scales			
	K=1	K=4	K=8	K=12
GCN	87.1	88.2	88.6	88.1
MS-GCN[42]	87.1	88.2	88.9	88.2
SDA (Ours)	87.1	88.3	**89.1**	88.3

Table 4.3: Ablation study on spatial aggregation scheme. Numbers are Top-1 accuracy (%).

of scales. We replace the spatial aggregation strategy of the LSTA blocks, referred to as 'GCN', 'MS-GCN' and 'MSDA', respectively. We observe that our decentralised aggregation strategy can outperform basic adjacency powering method on different scales.

ATPA module To validate the effectiveness of our attention-enhanced temporal aggregation method, we conduct ablation experiments on different temporal aggregation schemes, and the results are shown in Table 4.4. From the table we can see that our proposed temporal pyramid aggregation (w/o attention) scheme outperforms the direct aggregation as in MS-TCN[42]. The final result can be further improved when combining with maximum response attention (w/ attention). We also conduct extensive experiments to explore the parameter selection in TPA and MAM module, the results are shown in Table 4.5 and Table 4.6

The motion stream indeed captures temporal information, focusing on the overall movement dynamics over time. However, this ATPA module adds value by further enhancing the temporal modeling at multiple scales. While the motion stream provides a broad and continuous representation of the temporal dynamics, the ATPA module allows the model to focus on both short-term and long-term dependencies more effectively by applying attention mechanisms and pyramid aggregation.

This combination helps the model not only to capture the general flow of motion but also to preserve and highlight critical temporal patterns that may be missed or underemphasised by the motion stream alone. Thus, the ATPA module comple-

Temporal Aggregation	Params	Attention	Acc
MS-TCN[42]	1.2M	-	88.2
TPA(ours)	1.0M	-	88.5
ATPA(ours)	1.0M	✓	**89.1**

Table 4.4: Comparisons between regular MS-TCN and our TPA module with or without Temporal Maximum Attention. Numbers are Top-1 accuracy (%).

Method	Number of Subsets	Acc
ATPA	$s = 4$	89.0
	$s = 6$	**89.1**
	$s = 8$	88.9

Table 4.5: Comparisons between TPA modules with different s. s is the number of subsets for sub-convolution operations. Numbers are Top-1 accuracy (%).

MAM	Number of Dilation Rates						
	K=3	K=5				K=7	K=9
	1∼3	1	1∼2	1∼3	1∼4	1∼3	1∼3
Acc	88.6	88.3	88.6	**89.1**	88.4	88.4	88.1

Table 4.6: Parameter selection of MAM. K denotes the kernel size of $1D$ convolution, and 1∼3 indicates the dilation rates of 1, 2 and 3.

ments the motion stream by providing finer temporal details and ensuring that both global and local temporal structures are adequately modelled, which ultimately improves the overall performance.

Visualisation We visualise the output feature maps of the last LSTA-block in Figure 4.3. For the spatial modelling (TOP), the size of green circle around each joint indicates its importance. We can see our model can focus on the parts that are most relevant to the action. Specifically, both hands are well focused for actions ”type on a keyboard”, ”put on a hat”; the model focuses more on arm parts for action ”hand waving”; for ”walking” action, the model focuses on the lower body, especially the feet and knees.

For the temporal modelling, we show an example of the learned feature responses for several frames and the corresponding skeleton sketches are in Figure 4.3 (BOTTOM). For action ”hand waving”, we can see the model focuses more on

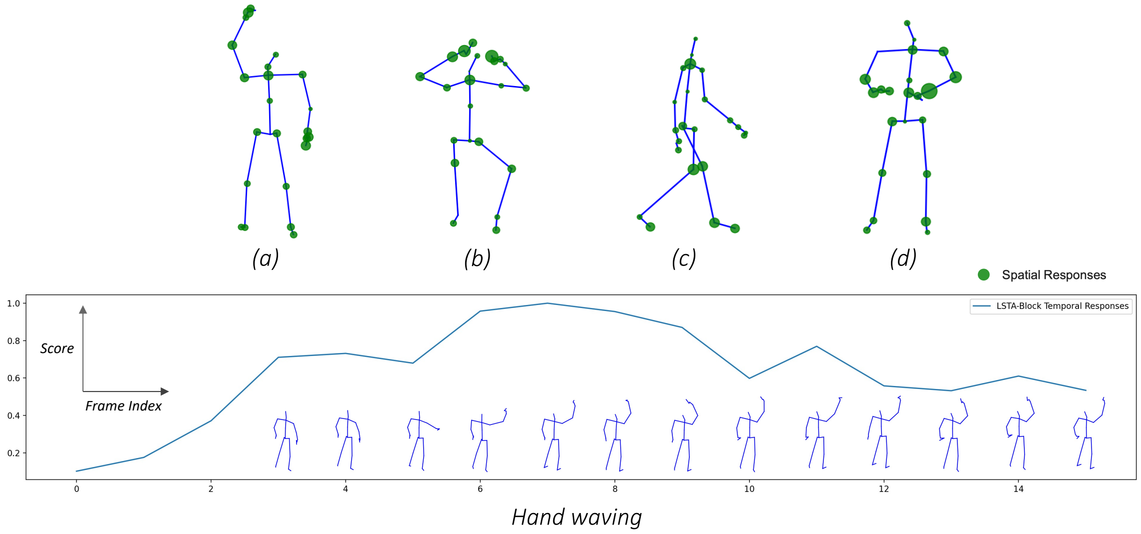


Figure 4.3: TOP: Examples of the joint feature responses for four actions (a) "walking" (b) "put on a hat" (c)"hand waving" (d) "type on a keyboard". The size of green circles indicates the importance of the joint. BOTTOM: Visualisation of the temporal feature responses for each of the frame for action "hand waving". X-axis denotes the input skeleton frame index, and Y-axis indicates the importance of each frame (scaled to range $[0, 1]$).

the process of raising hand in temporal domain and also pays attention on localised motion patterns in the spatial domain (i.e., "the raised hand"), suggesting the capability of our model in capturing the spatio-temporal dependencies in skeleton-based action recognition.

4.4 Chapter Summary

We propose a factorised architecture which alternately performs spatial feature aggregation and temporal feature aggregation for efficient long short-term dependency modelling to skeleton-based action recognition. We develop three modules in our architecture: a multi-scale spatial aggregation (MSA), temporal pyramid aggregation (TPA) and maximum-response attention module (MAM). MAM is an almost parameter-free local attention module, which can be easily integrated into both MSA and TPA. Finally, we achieve comparable performances on three public benchmarks while keeping lowest model complexity.

Chapter 5

Part-aware Prototypical Graph Network for One-shot Skeleton-based Action Recognition

The DualHead-Net introduced in Chapter 3 and the LSTA-Net introduced in Chapter 4 both focus on fully-supervised action recognition, where large quantities of well-labelled data is required for training. Moreover, the test set comprises action samples that must belong to the same categories as the training set. In this chapter,to alleviate the aforementioned limitations, we introduce a novel part-aware prototypical graph network that addresses significant challenges in one-shot skeleton-based action recognition, such as inadequate part-level generalisation capabilities for new actions. Unlike existing approaches that mainly focus on body-level action modelling, o ur method utilises unique part-aware prototypes to learn transferable knowledge. This contribution not only enhances the accuracy of action recognition in one-shot setting but also improves the efficiency of learning from limited examples. Our approach is novel in its integration of part-aware attention fusion mechanism, which has been demonstrated to significantly outperform existing models in tests conducted across multiple datasets,These advancements provide substantial benefits for applications where quick and accurate recognition from minimal data is crucial.

5.1 Introduction

Skeleton-based action recognition, due to its advantage of preserving subject privacy and robustness, has attracted increasing attention during the past few years [2, 7, 8, 84, 58, 54, 41]. Existing skeleton-based action recognition methods typically focus on the problem of many-shot classification [8, 57, 41], where each class has substantial amount of samples during training. Nevertheless, the acquisition of well-annotated skeletal sequences is labour-intensive and time-consuming. As such, in the low-data regime, few-shot learning approaches [73, 63] provide a promising strategy and yet they are rarely investigated in the skeleton-based action recognition.

In this chapter, we study the problem of one-shot skeleton-based action recognition, which poses unique challenges in learning novel action classes given knowledge from known base classes*. Several recent attempts [43, 78] on the one-shot skeleton-based action recognition utilise metric learning [47] or meta-learning framework [63] and mainly focus on learning a holistic representation of actions based on graph neural networks, or a spatio-temporal representation for temporal alignment [78]. However, it is particularly challenging to capture the fine-grained action space commonly-seen in practice with such global representations. For instance, in order to distinguish the *'staple book'* from *'open a box'* in the NTU RGB+D 120 [37] dataset, it is crucial to model the local region around hands due to the subtle differences between those two classes. The holistic representations, unfortunately, are often unable to focus on such local spatial features given the small support set in

* We follow the commonly-adopted one-shot learning setting here. The action categories are divided into two sets: base classes and novel classes. We assume that the base classes have sufficient examples per class for training while we only have one support sample per novel class. Given a query sample from the novel classes, we aim to find which class it belongs to.

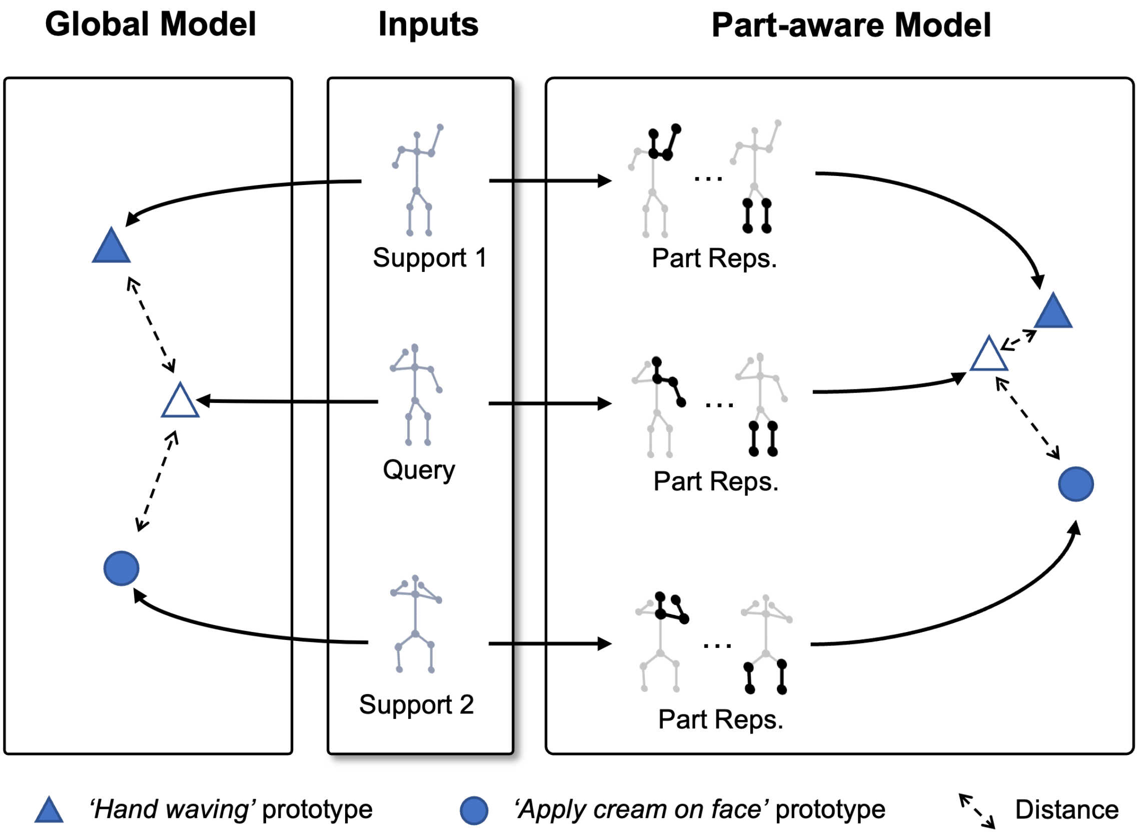


Figure 5.1: Existing methods(left) typically rely on holistic representation of actions. In contrast, ours(right) adopts a part-aware model to learn from multiple part graphs for one-shot skeleton-based action recognition.

the few-shot setting, resulting in poor generalisation for fine-grained action classification. Recently, many fully-supervised skeleton-based works [79, 65] utilise the partial body analysis for local region representation modelling. However, these works mainly focus on learning a representation from body-level and part-level simultaneously where complex part-and-body-level graph architecture and a large amount of data are required during training stage. In this one-shot recognition task, directly adopting such learning strategy cannot perform well in learning discriminative features for novel actions.

To address the aforementioned limitations, we propose a novel part-aware prototypical representation learning framework for the one-shot skeleton-based action recognition, as illustrated in Figure 5.1. Specifically, the part-aware prototypes are learned via the meta-learning framework [63] and our method first captures the skeleton motion patterns at *body level* and then attends to *part level* for part-aware prototypes learning. Different from previous methods where both body-level and part-level representations are learned simultaneously for recognition, we devise a novel class-agnostic attention fusion mechanism which selects part-level representations to generate a part-aware prototype for each class. Our attention mechanism can highlight the importance of parts for each action class based on an contrastive learning manner.

Our part-aware prototypical graph network consisting of three modules: a cascaded embedding module for computing part-based representations, an attention-based part fusion module for generating the part-aware prototypical representation, and a matching module to produce the final classification. For the embedding module, we adopt a modified GCN design [84, 57] with two stages. The first stage takes the input skeleton sequence and use a spatio-temporal graph network to compute initial context-aware features for all the joints. The second stage is composed of multiple part-level graph networks, each of which is defined on a local part region generated according to a set of rules such as semantic partitions. The initial joint features are then fed into those part-level graph networks to produce part-aware representations. Subsequently, our part fusion module combines all the part-aware features weighted by a part-level attention and generate a part-aware prototype for the input skeleton sequence. Finally, the matching module outputs the class label of the query based on the cosine distance between the part-aware prototype of the query and support examples.

The main contributions of this chapter are summarised as follows: (1) We propose a novel one-shot learning strategy for skeleton-based action recognition based on part-aware prototypical representation learning. (2) We develop a part-aware prototypical graph network to capture the skeleton motion patterns at two distinctive spatial levels and a class-agnostic attention mechanism to highlight the important parts for each action class. (3) We achieve the new state-of-the-art on two public datasets under one-shot learning setting.

The remainder of this chapter is organised as follows. In Section 5.2, we introduce our proposed part-aware prototypical graph network for one-shot skeleton-based action prediction in detail. We present the experimental results and comparisons in Section 5.3. Finally, we conclude the chapter in Section 5.4.

5.2 Part-aware Prototypical Graph Network

5.2.1 Problem Definition

We consider the problem of one-shot skeleton-based action recognition, which aims to classify skeleton sequences from only one labelled sample per class. To this end, we adopt a meta-learning strategy [63] that builds a meta learner $\mathcal{M}$ to resolve a family of classification tasks(also called episodes) $\mathcal{T} = \{T\}$ sampled from an underlying task distribution $P_{\mathcal{T}}$.

Formally, each meta classification task T, consists of a support set S with labeled skeleton samples and a set of query skeleton sequences Q. In the C-way one-shot setup, the support set $S = \{(\mathbf{x}_c^s, c) | c \in \mathcal{C}_T\}$, where $\mathbf{x}_c^s$ indicates the skeleton sequence, c indicates the action label, C_T is the subset of class sets for the task T and $|C_T| = C$. The query set $Q = \{(\mathbf{x}_j^q, c_j^q)\}$, where $\mathbf{x}_j^q$ is the query skeleton sequence, and $c_j^q \in \mathcal{C}_T$ is the corresponding label which is known during training but unknown during testing.

We introduce a meta training set $\mathcal{T}_{train} = \{(\mathcal{S}_n, \mathcal{Q}_n)\}_{n=1}^{|\mathcal{T}_{train}|}$ over the training class(also called base class) set C^{train}. The meta learner $\mathcal{M}$ is therefore trained episodically on the tasks $\mathcal{T}_{train}$ and is able to encode the knowledge on how to perform action recognition on different action categories across tasks. Finally, to evaluate our meta learner, we construct a test set of tasks $\mathcal{T}_{test} = \{(\mathcal{S}_m, \mathcal{Q}_m)\}_{m=1}^{|\mathcal{T}_{test}|}$, where the test class(also called novel class) set $\mathcal{C}^{test}$ is non-overlapped with $\mathcal{C}^{train}$.

5.2.2 Metric Learning for Skeleton Data

The meta classification task T is tackled by learning the distance between two skeleton sequences. Concretely, given a query sample $\mathbf{x}^q$ and a support sample $\mathbf{x}^s$, the goal is to learn a model D that can measure the distance between $\mathbf{x}^q$ and $\mathbf{x}^s$. Formally,

$$Distance = D(\mathbf{x}^q, \mathbf{x}^s). \tag{5.1}$$

To achieve this, we decompose the goal into three steps. Firstly, a cascaded graph embedding network $\mathcal{F}_{embed}$ is employed to transform the raw inputs into multiple part-based representations.

$$\{\Gamma_1, \Gamma_2, ..., \Gamma_K\} = \mathcal{F}_{embed}(\mathbf{x}), \tag{5.2}$$

where $\Gamma_k \in R^d$, d is the dimension of the feature. Then we adopt a part fusion module $\mathcal{F}_{fuse}$ to fuse the part-based representations and generate part-aware prototypical embeddings:

$$\epsilon = \mathcal{F}_{fuse}(\Gamma_1, \Gamma_2, ..., \Gamma_K). \tag{5.3}$$

We separately generate embeddings for support sample and query sample. Finally, we exploit a distance function $d(.,.)$ on the query embedding and support embedding to calculate their distances. Below we will introduce our model architecture design for the embedding and fusion module.

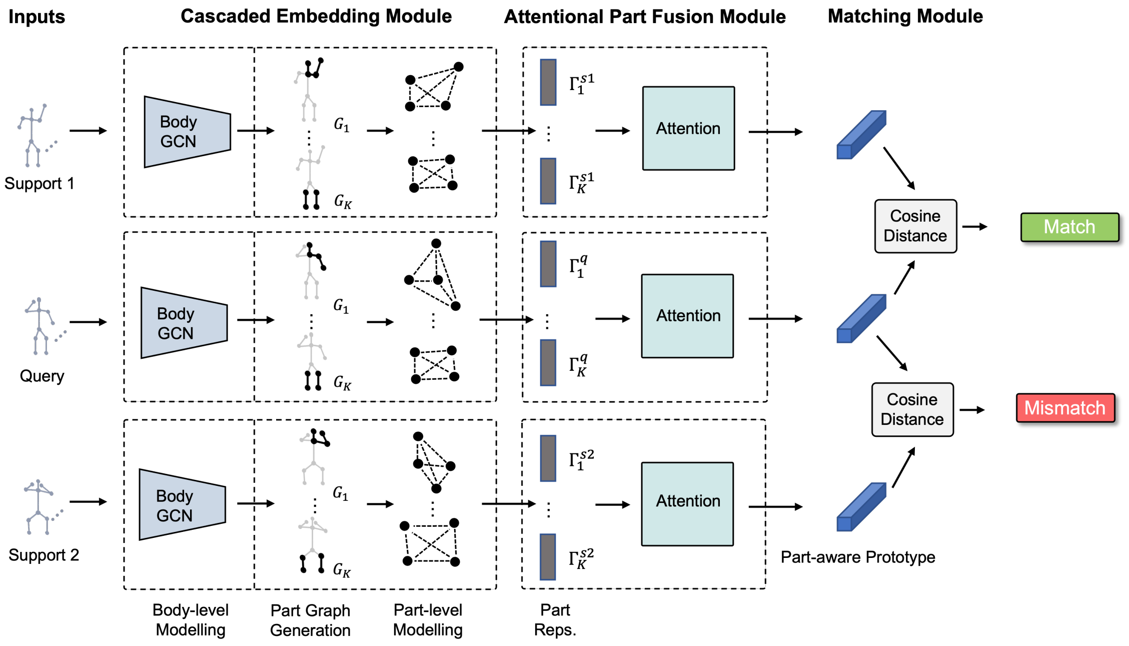


Figure 5.2: Overview of our framework. Cascaded embedding module extracts part-based representations with a two-stage graph network. In the first stage, a body GCN computes an initial context-aware features for all joints. The second stage is about part-level modelling, where we first generate multiple part graphs according to a set of rules, and then feed the representations sampled by the part graphs into a series of part GNNs to compute part representations. The attentional part fusion module highlights important parts based on a class-agnostic attention mechanism, and generates part-aware prototypes. The matching module outputs the class label of the query based on the cosine distance between the part-aware prototype of the query and support examples.

5.2.3 Network Architecture

An overview of our framework is shown in Figure 5.2. We implement $\mathcal{F}_{embed}$, $\mathcal{F}_{fuse}$ and distance calculation process with three modules: cascaded embedding module, attentional part fusion module and matching module. Concretely, given a skeleton sequence $\mathbf{x} \in R^{D\times T\times V}$, where $D \in \{2, 3\}$ denotes the 2D or 3D coordinates of joints, T indicates the sequence length, and V represents the number of semantic joints. The goal of the cascaded embedding module $\mathcal{F}_{embed}$ is to transform a raw sequence $\mathbf{x}$ into multiple part-based representations. The attentional part fusion module highlights important parts and generate part-aware prototypical representations. And the matching module aims to exploit the prototypical representations to perform classification. Below, we introduce the details of our network design.

5.2.3.1 Cascaded embedding Module

In the meta-learning framework, the graph network is employed to enable the meta knowledge learned from base actions is transferable to novel actions. Prior many-shot graph networks [8, 41, 84] rely heavily on holistic body-level representations. However, these holistic representation-based approaches are unreliable to be generated to novel classes because they cannot effectively capture subtle discrepancies of different classes which is the key to distinguishing the fine-grained action sequences. To this end, the proposed model is designed based on novel prototypes to enhance part-level patterns by modelling graphs in multiple spatial regions.

The proposed network attempts to extract useful patterns in a two-stage manner. In the first stage, we employ a basic graph embedding network to generate a context-aware feature of V joints and is referred as body-level modelling. In the second stage, we construct different part-level graphs based on the output of first stage, and exploit different local graph networks for part-level modelling. Then a cascaded two-stage graph network is formed by successively extracting both body-level and part-level features. Details are described below.

5.2.3.1.1 Body-level modelling The body-level modelling takes the initial sequences $\mathbf{x} \in R^{D \times T \times V}$ as inputs and generates the body representations $\Gamma \in R^{d_0 \times T_0 \times V}$, where d_0 is the dimension of the features and T_0 is the number of temporal frames. Note that T_0 is not equal to T because the down sampling operation is adopted in the temporal dimension for the body-level module. Specifically, the ST-GCN [84] is adopted for the backbone of the body-level module in order to extract discriminative joints features. Since each layer of ST-GCN consistently takes the

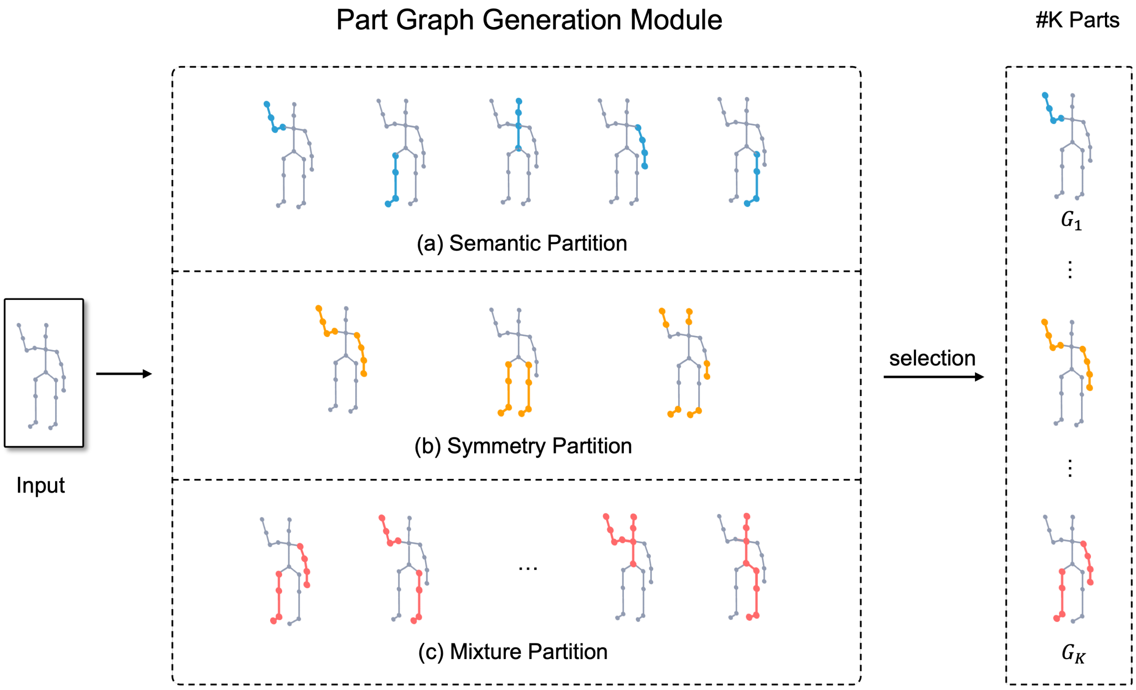


Figure 5.3: We generate part graphs based on a set of rules: (a) semantic partition; (b) symmetry partition and (c) mixture of semantic and symmetry partition.

human body structure as the default connection of the graph, the message passing through different joints is not easily over-smoothed compared to the global self-attention based structures [58, 8]. We adopt a shallower ST-GCN as the original version which contains L_{body} layers.

5.2.3.1.2 Part-level modelling Unlike body-level modelling, our part-level module focuses on enhancing regional patterns of the body-level representations. Defining part graphs based on motion along the time domain offers significant advantages for enhancing action recognition, particularly within the realm of skeleton-based action recognition. This method enhances the precision of detecting and differentiating actions by focusing on temporal variations in movement. It improves the model's ability to maintain temporal coherence across sequences and increases sensitivity to subtle motion movements. Moreover, it offers robustness against environmental changes, making it suitable for complex settings. Overall, this approach enriches action recognition models, supporting more advanced tasks like multi-task and transfer learning, such as one-shot setting.

Concretely, we construct K part graphs $\{G_i\}$ using several heuristic rules derived from human body structural characteristics, and then we sample the joint representation of each part graph G_i from the body representation Γ. The representation of each part is then fed into a part graph network to enhance the local correlation between the part joints. The global pooling is performed on the output of each part graph network to produce part representations $\{\gamma_i \in R^d\}$. The details are illustrated below.

5.2.3.1.3 Part Graph Generation We generate part graphs based on the natural characteristics of human skeleton structure which is illustrated in Figure 5.3. The rules defined for the part generation is described as follows:

- *Semantic partition.* Since the skeleton joints are semantically aligned over different samples, an intuitive idea is to divide the body joints into groups according to their semantic meanings. A semantic partition strategy is proposed to produce the sub-graphs and is shown in Figure 5.3(a).
- *Symmetry partition.* Since the actions are usually correlated with specific joints such as wrists and knees, the generation of such sub-graphs is defined based on the symmetry as shown in Figure 5.3(b).
- *Mixture partition.* Since some actions are performed by a combination of semantic parts, such mixture partition enhance the flexibility and is shown in Figure 5.3(c).

For each part graph G_i, we sample the corresponding joint representations from the output of body-level module $\Gamma \in R^{d_0 \times T_0 \times V}$ to obtain the part representations

$\Gamma_i^0 \in R^{d_0 \times T_0 \times V_i}$, where V_i is the joint number of G_i. Sampling operations are performed only in the spatial dimension to ensure that all temporal information can be preserved for the following part modelling.

5.2.3.1.4 Part Graph Neural Network Since the design of part generation does not have a constraint that the joints must be connected to each other, the GCN [84] style architecture may not be the optimal choice for being the network structure in this situation. To cope with this problem, a densely connected graph network is exploited to completely establish correlations. Specifically, the part graph network is similar to AGCN [58] but but only preserves the non-local style messages passed between joints in the spatial dimension. In this way, the temporal convolutions can be alternately operated after the spatial message passing to capture extensive motion patterns. Additionally, partial graph networks are allowed to share their weights to minimise parameter burden.

5.2.3.2 Attentional Part Fusion Module

The structure of our attentional part fusion module $\mathcal{F}_{fuse}$ is shown in Figure 5.4. It aims to highlight important parts and generate the part-aware prototypical representation by parts fusion. Specifically, we denote the output of the part graph network as $\Gamma_i \in R^{d_1 \times T_1 \times V_i}$. We then perform an average pooling over the spatial and temporal dimensions to receive the part representations: $\gamma_i = \mathrm{AvgPool(\Gamma_i) \in R^{d_1}}$. The part representations $\{\gamma_i | i = 1, 2, ..., K\}$ are concatenated to form a unified representation, followed by a semantic attention module $\mathcal{A} \in R^K$, where each element in $\mathcal{A}$ indicates the importance of an individual part. The attention score $\mathcal{A}$ is computed using a simple multi-layer perception(MLP) and a *sigmoid*$(\cdot)$ function to convert the resulting values to $[0, 1]$:

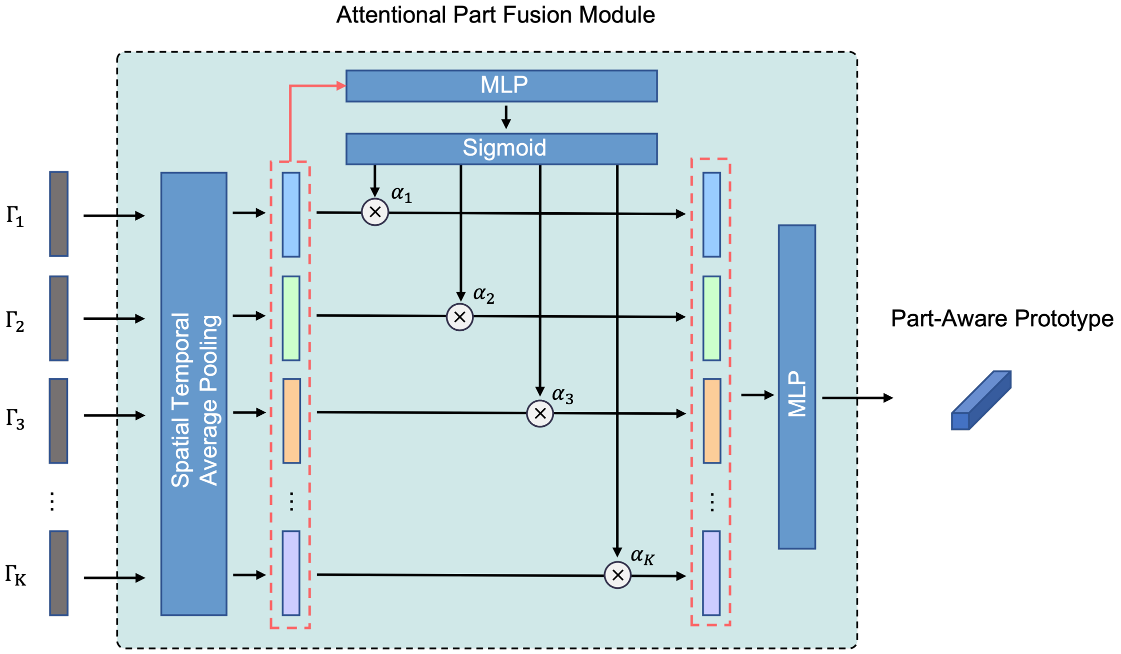


Figure 5.4: This whole diagram describes the calculation process of our attention part fusion module. It can highlight important parts and then generate **Part-aware Prototype**. Details can be found below.

$$\mathcal{A} = \mathrm{MLP}(\gamma_1 \oplus \gamma_2 \oplus ... \oplus \gamma_{\mathrm{K}}), \tag{5.4}$$

where $\oplus$ denotes the concatenate operation. The semantic attention is used to weight the part representations as follows:

$$\gamma_i' = \alpha_i \odot \gamma_i, \tag{5.5}$$

where $\alpha_i \in [0, 1]$ is the i-th value of $\mathcal{A}$, $\odot$ indicates the element-wise multiplication.

Finally, we fuse the part representations to generate the graph embedding $\epsilon \in R^d$ by the MLP layer on the concatenated features:

$$\epsilon = \mathrm{MLP}(\gamma_1' \oplus \gamma_2' \oplus ... \oplus \gamma_{\mathrm{K}}'). \tag{5.6}$$

5.2.3.3 Matching module

Our matching module aims to perform classification by utilising the generated part-aware prototypes and graph embeddings. A query sample is classified by assigning the class of the nearest support sample using a distance function $d(.,.)$. In this work, we focus on mining the part-level patterns in the skeleton data, which is encoded in the cascaded embedding module $\mathcal{F}_{embed}$. Hence we simply adopt the dot product to the normalised graph to compute their cosine similarities:

$$d(\epsilon^q, \epsilon^s) = -(\frac{\epsilon^q}{||\epsilon^q||})^T \cdot \frac{\epsilon^s}{||\epsilon^s||}. \tag{5.7}$$

During training, the query skeleton is classified through the $softmax(\cdot)$ of the distances to the support skeletons.

5.3 Experiments

We evaluate our proposed one-shot learning method on NTU-RGB+D 120 [37] and NW-UCLA [77]. Below, we firstly depict information about the implementation details. Secondly, we introduce the evaluation protocols in details and compare the proposed model with the state-of-the-art methods based on the same evaluation protocols. Thirdly, we comprehensively evaluate the effectiveness of each component of the proposed model by showing the results of ablation studies. Finally, we report some visualisation results for analysis.

5.3.1 Implementation Details

For a c-way, 1-shot setting, we randomly sample c classes with each class containing only 1 example as the support set. We construct the query set to have c examples, where each unlabelled sample in the query set belongs to one of the c classes. Thus each task during meta training stage has a total of 2c examples. We report the accuracy by adopting the evaluation protocols described above in the following experiments. For NTU-RGB+D 120, we adopt the data-preprocessing procedure as introduced in [90]. Specifically, if one frame contains two persons, the frame is split into two frame by making each frame contain one human skeleton. The entire skeleton sequence is segmented into 64 clips equally, and one frame from each clip is randomly selected to have a new sequence of 64 frames. During training, data argumentation is applied by randomly rotating the 3D skeletons to some degrees at sequence level to be robust to the view variation. Three degrees (around X, Y , Z axes, respectively) between $[-17, 17]$ are randomly select for one sequence. For NW-UCLA, we use the same data-preprocessing in [11], where total 64 frames are sampled from original data. During training and testing, the maximum frame number is set to $T = 64$.

In our experiments, we adopt $L_{body} = 5$ layers of ST-GCN[84] as the backbone network for body-level representation learning. Then, we generate K part graphs, each with V' node representation and feed into $L_{part} = 5$ layers of non-local blocks for part-level representation modelling. Finally, an attention-based mechanism is adopted to fuse part representations.

We optimised our model with Stochastic gradient descent(SGD), with a starting learning rate of 0.1 and decaying at 100 and 200 epochs by 0.1. We report the performance at 300 epochs. All experiments are conducted using PyTorch deep learning framework with 4 Tesla V100 GPUs.

5.3.2 Evaluation Protocols

NTU RGB+D 120. The protocol adopted for evaluating the effectiveness of the proposed model follows the standard one-shot protocol introduced in [37]. Concretely, the entire dataset is divided into two folders, including the auxiliary set *a.k.a.* training set and the one-shot evaluation set. The **Auxiliary Set** contains all samples from 100 classes which are used for both training and validation. The **One-shot Evaluation Set** contains 20 novel classes, namely, *A1, A7, A13, A19, A25, A31, A37, A43, A49, A55, A61, A67, A73, A79, A85, A91, A97, A103, A109, A115*, please refer to Section 2.1 for more details. For each novel class, only one sample is selected as the exemplar, and [37] is referred for more details [†]. All remaining samples contained in the novel classes can be used for testing the model performance.

NW-UCLA. The protocol for NW-UCLA [77] is ingeniously designed in this paper since no previous one-shot protocols are available. Specifically, the dataset is partitioned into the **Auxiliary Set** including *A1, A3, A5, A7, A9*, and the **Evaluation Set** containing *A2, A4, A6, A8, A10*, please refer to Section 2.1 for more details. The test phase is analogous to the descriptions given for the NTU RGB+D 120 dataset.

5.3.3 Quantitative Results

We evaluate our proposed method on two public benchmarks under one-shot setting.

On **NTU RGB+D 120** dataset, we perform experiments on five different experimental class reduction ratios, from 20 to 100. For a fair comparison, an identical evaluation protocol is adopted for all listed methods. The results are shown in Table 5.1. Specifically, the proposed model obtains an accuracy of 65.6% with 100

[†] https://github.com/shahroudy/NTURGB-D

training classes, significantly outperforming state-of-the-art methods [37, 44, 43, 78] based on the one-shot learning by 8.6% and exceeding the APSR method [37] by a large margin of 10.3%. As the number of training classes decreases, the gaps in the accuracy of the listed methods compared to our method gradually narrow but there is still a gap of at least about 5% (*e.g.,* the JEANIE [78] method achieves 38.5% accuracy using 20 training classes, while our method reaches 43.0% accuracy). Furthermore, it presents four variants based on the ProtoNet [63] as the baseline methods, including ST-GCN [84], MS-G3D [41] and CTR-GCN [8]. As shown in Table 5.1, the variant of ProtoNet [63]+ST-GCN [84] gives better results than the other three variants in most cases. It suggests that directly adopting advanced graph networks in many-shot setting can not bring better generalisation in one-shot setting, where the key is to learn novel action classes given knowledge from known base classes. In contrast, our part-aware prototype modelling exploits part representations for metric learning and is more effective.

We also evaluate the proposed part-aware architecture on NW-UCLA [77] to verify its effectiveness and generalisation for one-shot skeleton-based action recognition. Since there is no previous work on one-shot learning based on this dataset, we thus re-implement the open-sourced models, SL-DML [44] and Skeleton-DML [43] and the comparisons are mainly made between the proposed model and four baseline models. Table 5.2 shows the results on NW-UCLA dataset. Our method performs the best among all listed methods, followed by the ProtoNet [63]+MS-G3D [41] with 2.1% lower accuracy.

5.3.4 Ablation Studies

In this subsection, we perform ablation study to evaluate the effectiveness of our proposed modules and attentional fusion strategy. Except for the experiments in **Incremental ablation study**, all the following experiments are performed by mod-

# Training Classes	20	40	60	80	100
APSR [37]	29.1	34.8	39.2	42.8	45.3
SL-DML [44]	36.7	42.4	49.0	46.4	50.9
Skeleton-DML [43]	28.6	37.5	48.6	48.0	54.2
JEANIE [78]	38.5	44.1	50.3	51.2	57.0
ProtoNet [63]+ST-GCN [84]	41.5	49.6	54.2	55.2	61.1
ProtoNet [63]+MV-IGNet [79]	41.6	49.2	53.1	54.5	60.1
ProtoNet [63]+MS-G3D [41]	41.1	48.7	54.4	52.7	59.5
ProtoNet [63]+CTR-GCN [8]	39.9	49.1	53.6	54.2	58.8
Ours	**43.0**	**50.3**	**55.7**	**56.5**	**65.6**

Table 5.1: Experimental results on NTU RGB+D 120 under different number of training classes. Numbers are the Accuracies(%). 'ProtoNet + *' indicates the ProtoNet is implemented with * as the embedding network.

Method	Accuracy(%)
SL-DML [44] †	65.6
Skeleton-DML [43] †	72.8
ProtoNet [63]+ST-GCN [84]	79.8
ProtoNet [63]+MV-IGNet [79]	80.9
ProtoNet [63]+MS-G3D [41]	81.2
ProtoNet [63]+CTR-GCN [8]	80.7
Ours	**83.3**

Table 5.2: Experiments on NW-UCLA dataset.'†' indicates the results are implemented by ourselves, based on their released codes under the same evaluation protocol.

ifying the target component based on the full model. All the experiments are conducted on the NTU-RGB+D 120 dataset, with 100 classes for training, 20 classes for testing.

5.3.4.1 Incremental ablation study

In Table 5.3, we evaluate the effectiveness of our model components in an incremental manner. We start from the baseline network, ST-GCN, which only takes the human body structure as the default connection of the graph for body-level representation learning. We then add our proposed components one-by-one. Specifically, we replace the last 5 layers of baseline network with our part-level embedding network, the Table 5.3 shows our **part-level modeling** improves the performance from

Method	Params	Body-Level	Part-Level	Attention	Accuracy(%)
Baseline	1.5M	✓	-	-	61.1
Ours	1.8M	✓	✓	-	**62.9**
	1.9M	✓	✓	✓	**65.6**

Table 5.3: Ablation study of different modules on NTU RGB+D 120. 'Body-level' indicates the original body-level representation modelling, 'Part-level' indicates the proposed part-level modelling. 'Attention' indicates the attentional part fusion.

# Parts	Params	Accuracy(%)
K = 0	1.5M	61.1
K = 5	1.6M	62.2
K = 10	1.9M	**65.6**
K = 20	2.6M	63.0
K = 30	3.3M	58.5

Table 5.4: Ablation study on the number of generated part graphs.

61.1% to 62.9%, demonstrating the effectiveness of our proposed part-level modelling structure. Note that our **part GNNs** are shared across all constructed sub-graphs. As a result, the part-level modelling does not introduce much parameters compared with baseline network. We can also observe that, with the help of **attentional part fusion** strategy, the performance can be further improved by 2.7%.

5.3.4.2 Number of part graphs

We generate multiple part graphs based on the natural characteristics of human skeleton structure according to the rules. We hence perform an ablation study of the number of part graphs used for part modelling, shown in Table 5.4. We can observe that, as the graph number K grows, the accuracy gradually improves. When $K = 10$, it reaches the best performance. When the graph number is very large ($K = 30$), the performance drops to 58.5%. This is because that too many part graphs contain redundancy and our attentional fusion module can not generate informative prototypes from the noisy part representations. Since our graphs are generated according to a series of rules, the size of our graphs is fixed. At this stage, redundancy does indeed occur as the number of graphs increases. However, our

# Heads	Params	Accuracy(%)
1	1.9M	**65.6**
2	2.2M	61.3
4	2.8M	60.6
8	3.9M	59.6

Table 5.5: Different number of self-attention heads in our adopted part GNNs.

Method	Accuracy(%)
w/o attention	62.9
Self-attention	61.2
MLP-attention (Ours)	**65.6**

Table 5.6: Comparison of different fusion strategies.

research has already verified the effectiveness of current part graphs; in the future, we can explore novel flexible part graph generation rules to generate smaller size parts as the number of parts increases to validate the redundancy issue.

5.3.4.3 Number of self-attention heads in part GNNs

In our part-level embedding network, we adopt a non-local style message passing regime to capture spatial joints correlations. In Table 5.5, we exploit the effect of different self-attention heads and report the performance. We found that 1-head structure achieves the best performance, which is different from state-of-the-art methods in many-shot setting, where they typically adopt multi-head attention. Our part GNNs model spatial patterns at a small region, which allows us to decompose the global representation into multiple local descriptors, hence each part GNNs can adopt a simpler structure.

5.3.4.4 Different fusion strategies

In our method, we adopt a simple MLP-based attentional fusion strategy to fuse different parts and generate part-aware prototypes. We replace our MLP-attention with a single self-attention layer on the part embeddings, and directly pool the

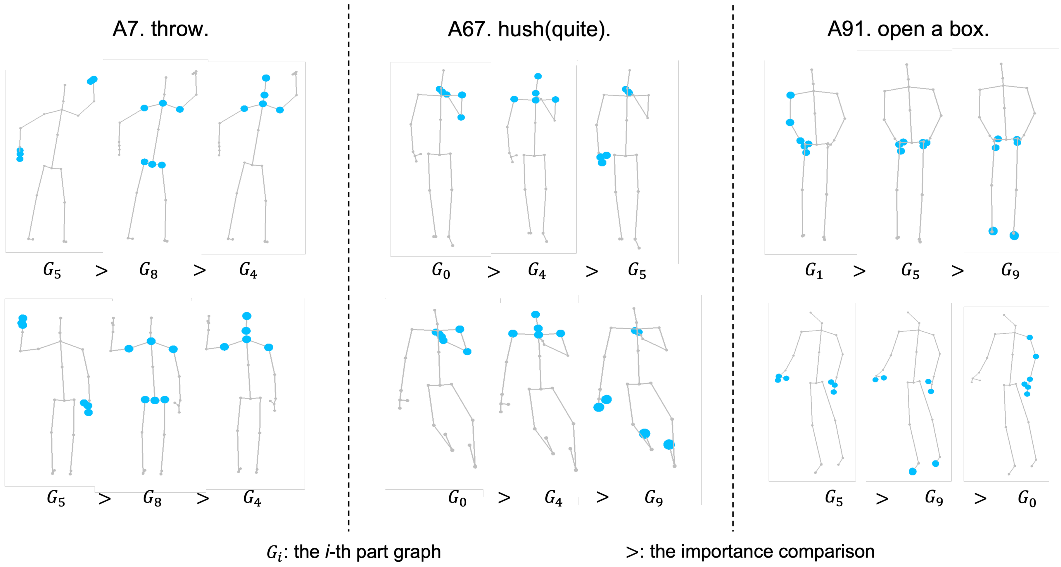


Figure 5.5: Visualisation of the attention prediction in our attentional part fusion module on novel actions, where total $K = 10$ part graphs are generated. For each skeleton sample in NTU RGB+D 120, the top-3 important part graph partitions are visualised. In each column, two samples from the same action class are visualised. We can observe that, for different actions, our attention block can select different information parts, while for the same action, attention is similar.

resulting parts as new prototypes, the result significantly drops from 65.6% to 61.2%, as shown in Table 5.6. We guess that the self-attention mechanism will make the parts over-smooth and the fused prototypes are not as discriminative as before.

5.3.5 Visualisation

In Figure 5.5, we qualitatively visualise the attention prediction of attentional part fusion module. For each skeleton, we show the top-3 important part graphs. We can observe that, for different actions, our attention block can select different information parts, while for the same action, attention is similar. This demonstrates that our attention mechanism is class-agnostic.

In Figure 5.6, we quantitatively compare our model with baseline global modelling method on 8 novel fine-grained actions which highly rely on hands. We can observe that our method outperforms the baseline ST-GCN on each class, demon-

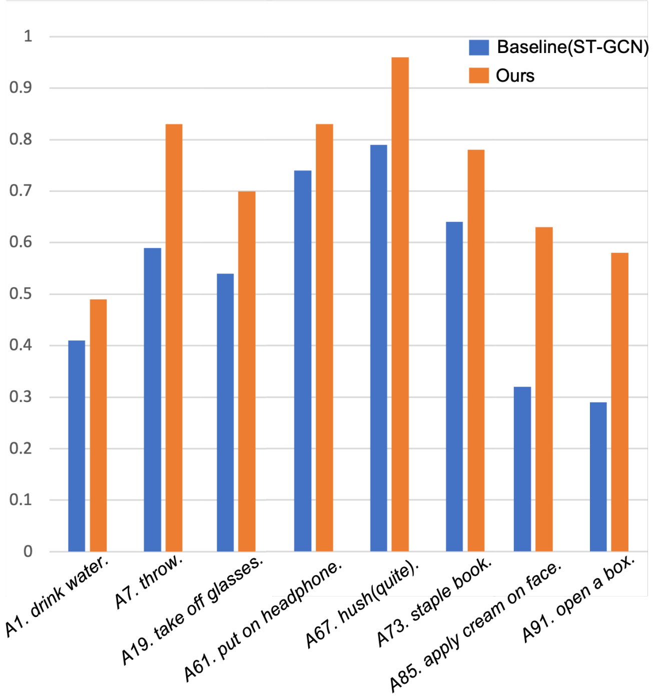


Figure 5.6: Performance comparison on 8 novel fine-grained action classes on NTU RGB+D 120.

strating the effectiveness of our local part-level modelling.

5.4 Chapter Summary

In this paper, we propose a novel part-aware prototypical graph network for one-shot skeleton-based action recognition, aiming to learn a rich fine-grained representation for action concepts via meta-learning framework. Our network consists of three main modules: a cascaded embedding module to extract part embeddings, where both body-level and part-level modelling are cascaded performed to capture skeleton motion patterns, an attentional part fusion module to generate part-aware prototypical representation, and a matching module to produce final classification. We evaluate our method on two public benchmarks, namely NTU RGB+D 120 and NW-UCLA dataset under one-shot setting. The results show that our method is able to achieve state-of-the-art under all setups, demonstrating the effectiveness of our proposed part-aware one-shot learning strategy.

Chapter 6

Conclusion

6.1 Conclusion

In this thesis, we have proposed three novel network models for human action recognition from 3D skeleton sequences.

First, we introduce a DualHead-Net model in order to capture both coarse-grained and fine-grained motion patterns from dynamic skeleton joints. Our key idea is to utilise two branches of interleaved graph networks to extract features at two different temporal resolutions. The branch with lower temporal resolution captures motion patterns at a coarse level, while the branch with higher temporal resolution is able to encode more subtle temporal movements. Such coarse-level and fine-level feature extraction are processed in parallel and finally the outputs of both branches are fused to perform dual-granularity action classification. We further propose a cross head communication strategy which can mutually enhance the features of both heads. We conduct extensive experiments on three large-scale datasets and our model achieves State-of-the-Art performance on all of these benchmarks.

Second, we introduce a LSTA-Net model to capture various spatio-temporal dependencies from body joints. Specifically, we propose a multi-scale decentralised aggregation (MSDA) module which is capable of extracting multi-scale spatial features. For long-term temporal modelling, we propose an attention-enhanced temporal pyramid aggregation (ATPA) module to model long-range temporal dynamics by deforming the ordinary convolution layer into a set of sub-convolutions and formulating them with a pyramid-like/hierarchical structure allowing residual-like connections between adjacent subsets. Besides, our attention module, namely maximum-response attention (MA) module is proposed for improving the local cross-channel interaction via multiple high efficient 1-D convolutions. In our framework architecture, these modules are complementary and stacked together which endows the network with both longshort spatial and temporal modelling abilities. The experimental results show that our proposed LSTA-Net achieves competitive or the-state-of-the-art performances with less than half of the parameters of MS-G3D.

Finally, as the acquisition of well-annotated skeletal sequences is labour-intensive and time-consuming, we study the problem of one-shot skeleton-based action recognition and introduce a part-aware prototypical graph network to learn transferable representation from known base action classes to novel action classes. Our model captures skeleton motion patterns at two distinctive spatial levels, one for global contexts among all body joints, referred to as body level, and the other attends to local spatial regions of body parts, referred to as the part level. We also devise a class-agnostic attention mechanism to highlight important parts for each action class. Specifically, we develop a part-aware prototypical graph network consisting of three modules: a cascaded embedding module for our dual-level modelling, an attention-based part fusion module to fuse parts and generate part-aware prototypes, and a matching module to perform classification with the part-aware representations. The extensive experiments demonstrate the effectiveness of our proposed one-shot action recognition framework.

6.2 Future Work

6.2.1 Generalised Few-shot Learning for Skeleton-based Action recognition

Existing works, such as those highlighted by NTU-120 [37] and Skeleton-DML [43] demonstrate that models trained on a set of action categories can abstract this knowledge to efficiently learn novel classes with minimal data, sometimes as little as one example per class. This capability of transferring learned behaviours to recognise new actions with very few examples is crucial for adapting to dynamic environments where new actions are frequently introduced.

However, these models usually have been evaluated only on their ability to classify novel classes, without considering their performance on previously learned or base classes. This evaluation protocol does not reflect the complexities of real-world systems, where the ability to discriminate between well-learned base classes and less-representative novel classes is crucial. Real-world applications, especially in dynamic environments like healthcare, sports, and interactive gaming, require a system that maintains high accuracy across both old and new actions without suffering from catastrophic forgetting.

To address these challenges, Generalised Few-Shot Learning (GFSL) is proposed, which not only allows the learning system to quickly adapt to new tasks with minimal data but also ensures that the knowledge of base classes is not overshadowed. This dual focus is essential for deploying practical skeleton-based action recognition systems that are robust and scalable.

Extending to Low-Shot Settings: Building on the one-shot learning scenario, we can naturally extend our approach to a low-shot setting, where the model is trained on a base dataset and then tested on a novel dataset without class overlap.

This extension requires sophisticated handling of the trade-offs between maintaining performance on base classes and adapting to novel classes. A new joint evaluation protocol, as suggested by [24, 81], should be adopted. This protocol facilitates fair comparisons by integrating performance measures across both novel and base classes in a unified label space, thus providing a more holistic measure of model effectiveness.

Here are some future directions:

Integration of Continual Learning: To better manage the trade-offs between learning new actions and retaining old ones, integrating continual learning strategies could prevent the forgetting of previously learned actions while incorporating new ones. Techniques like elastic weight consolidation or replay memory could be instrumental.

Cross-Domain Adaptability: Exploring GFSL models' adaptability across different domains, such as transitioning learned behaviours from sports analytics to healthcare monitoring, could significantly broaden the applicability of action recognition systems.

Enhanced Data Augmentation: Developing advanced data augmentation techniques that simulate a wider variety of human actions from limited examples could improve the robustness and generalisability of GFSL models.

Explainability and Trust: As GFSL systems are deployed in more critical applications, improving their explainability and building trust among users will become increasingly important. Research into interpretable machine learning models that provide insights into decision-making processes could address these concerns.

By addressing these future directions, research in GFSL for skeleton-based action recognition can significantly advance, leading to more versatile, efficient, and trustworthy systems that are capable of meeting the diverse and evolving demands of real-world applications.

6.2.2 Real-World Multi-Modal Action Recognition

Here are some motivations should be considered in real-world action recognition,

Integration of Multi-Modal Data for Comprehensive Analysis: The unique challenges presented by real-world settings require robust solutions that leverage the complementary strengths of various data modalities. While 3D skeleton sequences excel in providing structured information about human posture and motion, they often lack contextual details that are critical for recognising complex interactions, particularly those involving objects.

Enhancing Recognition through Data Fusion: For certain actions, especially where human-object interactions are involved, appearance information from RGB videos, contextual depth data, and the structured dynamics from skeleton data can collectively enhance recognition accuracy. Multi-modal data fusion leverages these diverse data streams, providing a richer and more discriminative feature set for action recognition. By integrating these modalities, systems can achieve a more nuanced understanding of the scene, leading to better performance in complex scenarios.

Datasets and Implementation Challenges: Several existing datasets for skeleton-based action recognition, such as NTU RGB+D 120 [37] and Toyota Smarthome [15], provide not only the skeleton sequences but also corresponding RGB videos and depth maps. This availability facilitates the exploration of fusion approaches to aggregate features extracted from these varied sources.

Specifically, unlike the NTU RGB+D 120 dataset, which is collected in a controlled environment, the Toyota Smarthome Untrimmed (TSU) dataset features a wide variety of activities performed in spontaneous manners. This dataset is particularly challenging as it targets the activity detection task in long, untrimmed videos. Such datasets are crucial for developing and testing models designed to operate in less constrained, real-world environments.

In the future, some directions can be explored,

Fusion Techniques: Investigate advanced fusion techniques that can effectively integrate the strengths of RGB data, depth information, and skeletal data. This includes exploring architectures like late fusion, where features from each modality are combined at a decision level, or early fusion, which integrates data at the input stage to capture inter-modal dynamics more effectively.

Temporal and Spatial Alignment: Address the challenges of aligning temporal and spatial data across modalities, which is critical for synchronous processing and accurate interpretation of concurrent data streams.

Real-World Application Scenarios: Focus on the adaptation of these multi-modal systems to specific real-world scenarios such as surveillance, healthcare monitoring, and interactive systems, where the variability and unpredictability of human actions present significant recognition challenges.

The fusion of RGB, depth, and skeleton data holds significant promise for advancing the state of action recognition technologies. By continuously refining the approaches to effectively merge these modalities, future systems can better address the nuanced demands of real-world applications, thereby enhancing both their accuracy and applicability.

Furthermore, while skeleton data provides numerous benefits, integrating RGB data can enhance system capabilities in ways that skeleton data alone cannot achieve: (1) Contextual and Environmental Cues: RGB data can provide context that is lost when only using skeleton data. For example, the interaction with objects that are not part of the human skeleton but are crucial for understanding the action being performed. (2) Complementary Information: RGB videos can offer additional details such as object textures, colours, and fine-grained motions that are not captured by skeleton data but may be crucial for accurately recognising complex activities.

The integration of RGB data into skeleton-based action recognition systems does not necessarily negate the advantages of using skeleton data but rather enhances the system's overall robustness and accuracy. By carefully balancing when and how RGB data is used, it is possible to design a system that leverages the strengths of both modalities without significant compromises.

Bibliography

[1] Jake K. Aggarwal and Lu Xia. Human activity recognition from 3d data: A review. *Pattern Recognition Letters*, 2014.

[2] Chaitanya Bandi and Ulrike Thomas. Skeleton-based action recognition for human-robot interaction using self-attention mechanism. In *FG*, 2021.

[3] Carlos Caetano, François Brémond, and William Robson Schwartz. Skeleton image representation for 3d action recognition based on tree structure and reference joints. In *SIBGRAPI*, 2019.

[4] Carlos Caetano, Jessica Sena, François Brémond, Jefersson A Dos Santos, and William Robson Schwartz. Skelemotion: A new representation of skeleton joint sequences based on motion information for 3d action recognition. In *AVSS*, 2019.

[5] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *TPAMI*, 2018.

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[7] Tailin Chen, Desen Zhou, Jian Wang, Shidong Wang, Yu Guan, Xuming He, and Errui Ding. Learning multi-granular spatio-temporal graph network for skeleton-based action recognition. In *ACM MM*, 2021.

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

[9] Zhan Chen, Sicheng Li, Bing Yang, Qinghan Li, and Hong Liu. Multi-scale spatial temporal graph convolutional network for skeleton-based action recognition. In *AAAI*, 2021.

[10] Ke Cheng, Yifan Zhang, Congqi Cao, Lei Shi, Jian Cheng, and Hanqing Lu. Decoupling gcn with dropgraph module for skeleton-based action recognition. In *ECCV*, 2020.

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

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

[13] Sangwoo Cho, Muhammad Maqbool, Fei Liu, and Hassan Foroosh. Self-attention network for skeleton-based human action recognition. In *WCACV*, 2020.

[14] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L. Yuille, and Xiaogang Wang. Multi-context attention for human pose estimation. *CVPR*, 2017.

[15] Rui Dai, Srijan Das, Saurav Sharma, Luca Minciullo, Lorenzo Garattoni, Francois Bremond, and Gianpiero Francesca. Toyota smarthome untrimmed: Real-world untrimmed videos for activity detection. *TPAMI*, pages 1–1, 2022.

[16] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *CVPR*, 2015.

[17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *ICML*, 2017.

[18] Shanghua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip HS Torr. Res2net: A new multi-scale backbone architecture. *TPAMI*, 2019.

[19] Xiang Gao, Wei Hu, Jiaxiang Tang, Jiaying Liu, and Zongming Guo. Optimized skeleton-based action recognition via sparsified graph regression. In *ACM MM*, 2019.

[20] Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *ICLR*, 2018.

[21] Jia Gong, Zhipeng Fan, Qiuhong Ke, Hossein Rahmani, and Jun Liu. Meta agent teaming active learning for pose estimation. In *CVPR*, 2022.

[22] Fei Han, Brian Reily, William Hoff, and Hao Zhang. Space-time representation of people based on 3d skeletal data: A review. *CVIU*, 2017.

[23] Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton. Enhanced computer vision with microsoft kinect sensor: A review. *IEEE transactions on cybernetics*, 2013.

[24] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *CVPR*, 2017.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, 2015.

[26] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018.

[27] Zhen Huang, Xu Shen, Xinmei Tian, Houqiang Li, Jianqiang Huang, and Xian-Sheng Hua. Spatio-temporal inception graph convolutional networks for skeleton-based action recognition. In *ACM MM*, 2020.

[28] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[29] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. 2017.

[30] Qiuhong Ke, Mohammed Bennamoun, Senjian An, Ferdous Sohel, and Farid Boussaid. A new representation of skeleton sequences for 3d action recognition. In *CVPR*, 2017.

[31] Tae Soo Kim and Austin Reiter. Interpretable 3d human action analysis with temporal convolutional networks. In *CVPRW*, 2017.

[32] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *ICLR*, 2016.

[33] Bin Li, Xi Li, Zhongfei Zhang, and Fei Wu. Spatio-temporal graph routing for skeleton-based action recognition. In *AAAI*, 2019.

[34] Chao Li, Qiaoyong Zhong, Di Xie, and Shiliang Pu. Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. *IJCAI*, 2018.

[35] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019.

[36] Wenbo Li, Longyin Wen, Ming-Ching Chang, Ser Nam Lim, and Siwei Lyu. Adaptive rnn tree for large-scale human action recognition. In *ICCV*, 2017.

[37] Jun Liu, Amir Shahroudy, Mauricio Lisboa Perez, Gang Wang, Ling-Yu Duan, and Alex Kot Chichung. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *TPAMI*, 2019.

[38] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In *ECCV*, 2016.

[39] Jun Liu, Gang Wang, Ping Hu, Ling-Yu Duan, and Alex C Kot. Global context-aware attention lstm networks for 3d action recognition. In *CVPR*, 2017.

[40] Mengyuan Liu, Hong Liu, and Chen Chen. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognition*, 2017.

[41] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *CVPR*, 2020.

[42] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *CVPR*, 2020.

[43] Raphael Memmesheimer, Simon Häring, Nick Theisen, and Dietrich Paulus. Skeleton-dml: Deep metric learning for skeleton-based one-shot action recognition. In *WACV*, 2022.

[44] Raphael Memmesheimer, Nick Theisen, and Dietrich Paulus. Signal level deep metric learning for multimodal one-shot action recognition. *ICPR*, 2020.

[45] Richard Moore, Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark J. Finocchio, Alex Aben-Athar Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. *CVPR*, 2011.

[46] Tsendsuren Munkhdalai and Hong Yu. Meta networks. *ICML*, 2017.

[47] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *ECCV*, 2020.

[48] Wei Peng, Xiaopeng Hong, Haoyu Chen, and Guoying Zhao. Learning graph convolutional network for skeleton-based human action recognition by neural searching. In *in AAAI*, 2020.

[49] Liliana Lo Presti and Marco La Cascia. 3d skeleton-based human action classification: A survey. *Pattern Recognition*, 2016.

[50] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. *ICLR*, 2017.

[51] Bin Ren, Mengyuan Liu, Runwei Ding, and Hong Liu. A survey on 3d skeleton-based action recognition using learning method. *Cyborg and Bionic Systems*, 2020.

[52] Andrei Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *ICLR*, 2019.

[53] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *CVPR*, 2016.

[54] Junxiao Shen, John Dudley, and Per Ola Kristensson. The imaginative generative adversarial network: Automatic data augmentation for dynamic skeleton-based hand gesture and human action recognition. In *FG*, 2021.

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

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

[57] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-Based Action Recognition with Multi-Stream Adaptive Graph Convolutional Networks. *TIP*, 2019.

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

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

[60] Chenyang Si, Wentao Chen, Wei Wang, Liang Wang, and Tieniu Tan. An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In *CVPR*, 2019.

[61] Chenyang Si, Ya Jing, Wei Wang, Liang Wang, and Tieniu Tan. Skeleton-based action recognition with spatial reasoning and temporal stack learning. In *ECCV*, 2018.

[62] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 2014.

[63] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *NIPS*, 2017.

[64] Sijie Song, Cuiling Lan, Junliang Xing, Wenjun Zeng, and Jiaying Liu. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In *AAAI*, 2017.

[65] Yi-Fan Song, Zhang Zhang, Caifeng Shan, and Liang Wang. Stronger, Faster and More Explainable: A Graph Convolutional Baseline for Skeleton-based Action Recognition. *ACM MM*, 2020.

[66] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. *CVPR*, 2019.

[67] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019.

[68] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, 2014.

[69] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

[70] Gül Varol, Ivan Laptev, and Cordelia Schmid. Long-term temporal convolutions for action recognition. *TPAMI*, 2017.

[71] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 2017.

[72] Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *CVPR*, 2014.

[73] Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *NIPS*, 2016.

[74] Hongsong Wang and Liang Wang. Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks. *CVPR*, 2017.

[75] Jiang Wang, Zicheng Liu, Ying Wu, and Junsong Yuan. Mining actionlet ensemble for action recognition with depth cameras. *CVPR*, 2012.

[76] Jiang Wang, Zicheng Liu, Ying Wu, and Junsong Yuan. Learning actionlet ensemble for 3d human action recognition. *TPAMI*, 2014.

[77] Jiang Wang, Xiaohan Nie, Yin Xia, Ying Wu, and Song-Chun Zhu. Cross-view action modeling, learning and recognition. In *CVPR*, 2014.

[78] Lei Wang, Jun Liu, and Piotr Koniusz. 3d skeleton-based few-shot action recognition with jeanie is not so naïve. 2021.

[79] Minsi Wang, Bingbing Ni, and Xiaokang Yang. Learning multi-view interactional skeleton graph for action recognition. *TPAMI*, 2020.

[80] Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. Eca-net: Efficient channel attention for deep convolutional neural networks. In *CVPR*, 2020.

[81] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *CVPR*, 2018.

[82] Lu Xia, Chia-Chih Chen, and Jake K. Aggarwal. View invariant human action recognition using histograms of 3d joints. *CVPR*, 2012.

[83] Chunyu Xie, Ce Li, Baochang Zhang, Chen Chen, Jungong Han, Changqing Zou, and Jianzhuang Liu. Memory attention networks for skeleton-based action recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

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

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

[86] Fanfan Ye, Shiliang Pu, Qiaoyong Zhong, Chao Li, Di Xie, and Huiming Tang. Dynamic gcn: Context-enriched topology learning for skeleton-based action recognition. In *MM*, 2020.

[87] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R. Manmatha, Mu Li, and Alexander Smola. Resnest: Split-attention networks. In *CVPR Workshops*, 2022.

[88] Pengfei Zhang, Cuiling Lan, Junliang Xing, Wenjun Zeng, Jianru Xue, and Nanning Zheng. View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In *ICCV*, 2017.

[89] Pengfei Zhang, Cuiling Lan, Wenjun Zeng, Junliang Xing, Jianru Xue, and Nanning Zheng. Semantics-guided neural networks for efficient skeleton-based human action recognition. In *CVPR*, 2020.

[90] Pengfei Zhang, Cuiling Lan, Wenjun Zeng, Junliang Xing, Jianru Xue, and Nanning Zheng. Semantics-guided neural networks for efficient skeleton-based human action recognition. In *CVPR*, 2020.

[91] Xikun Zhang, Chang Xu, and Dacheng Tao. Context aware graph convolution for skeleton-based action recognition. In *CVPR*, 2020.

[92] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE MultiMedia*, 2012.

[93] Wentao Zhu, Cuiling Lan, Junliang Xing, Wenjun Zeng, Yanghao Li, Li Shen, and Xiaohui Xie. Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. In *AAAI*, 2016.