HUMAN ACTION RECOGNITION USING ATTENTION BASED SPATIOTEMPORAL GRAPH CONVOLUTIONAL NETWORK

A DISSERTATION

Submitted in partial fulfillment of the

requirementsfor the award of a degree

of

MASTER OF

TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

Submitted by

Anshula Sharma -

2K21/CSE/07

Under the supervision of

Prof. Anil Singh Parihar

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DELHI TECHNOLOGICAL

UNIVERSITY

Bawana Road,

Delhi - 110042

May 2023

DECLARATION

I, Anshula Sharma - 2K21/CSE/07, hereby declare that the work presented in dissertation titled “Human action recognition using attention based spatiotemporal graph convolutional network” which is submitted to Delhi Technological University, Delhi as a part of the requirements for the award of Master of Technology in Computer Science & Engineering degree, is the result of my own independent research and efforts. I affirm that this work has not been submitted in any form or shape, partly or as a whole, with the intention of obtaining a degree, diploma, or certification from any educational institution or organization. The information and findings given in this work are the result of my own study, and any external sources utilized have been properly cited and referenced.

Place: Delhi, India

Date:

Anshula Sharma

2K21/CSE/07

CERTIFICATE

This is to certify that the dissertation titled "Human action recognition using attention based spatiotemporal graph convolutional network" submitted by Anshula Sharma (2K21/CSE/07), Delhi Technological University, Delhi, is a document of the candidate work completed under my supervision. This work is based on the research papers by Yan *et al* [1] and Heidari and Iosifidis [2]. This work is not presented to any other college, foundation, or institution to the best of my knowledge.

Place: Delhi, India
Date:

Prof. Anil Singh Parihar
SUPERVISOR

ABSTRACT

Human skeleton modelling has gained popularity in recent years. As skeleton data successfully handles dynamic settings and complicated backdrops, the human skeleton dynamics include essential information for the identification of human actions. GCNs have shown substantial effectiveness in modelling the non-Euclidean character of human skeleton structures. Human skeleton structures are present in the form of spatiotemporal graphs depicting sequences of body skeletons during an action. We present an attention-based human action recognition model that uses the mechanism of temporal and spatial attention modules to improve identification. The temporal attention module captures the most informative frames from a sequence of skeletons. The spatial attention mechanism then emphasizes the most informative joints from the frames highlighted. Frame selection is then performed to select the skeletons with the highest attention scores. Spatial and temporal modules are incorporated into the graph convolutional network. Both attention modules improve the model's effectiveness and the efficiency of skeleton-based human action identification when used together. The model is evaluated on two benchmarks of the NTURGB+D dataset, i.e., cross-view benchmark and cross-subject benchmark. The top-1 accuracy of both models is compared with existing benchmark techniques. The experimental findings show that our model exceeded the current benchmark methodologies, providing a considerable improvement over the baseline technique.

ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to Prof. Anil Singh Parihar, Department of Computer Science and Engineering, Delhi Technological University, Delhi, for his invaluable guidance and support throughout the course of my study. His experience in the field, in-depth knowledge of the subject matter, and desire to share his knowledge have all played a role in defining the direction and quality of my work. I shall be eternally thankful to him for his unwavering encouragement and support. I am grateful to the panel faculty members for their advice, ongoing monitoring, and encouragement to finish my task. They assisted me throughout by providing fresh ideas, crucial information, and pushing me to finish the assignment.

Anshula Sharma
2K21/CSE/07

CONTENTS

LIST OF FIGURES

Figures **Page No.**

LIST OF TABLES

Table **Page No.**

LIST OF ABBREVIATIONS

1. HAR – Human Action Recognition
2. CNN - Convolutional Neural Network
3. GCN – Graph Convolutional Network
4. LSTM – Long short-term Memory Networks
5. ST-GCN – Spatial Temporal Graph Convolutional Network
6. 2s-AGCN – Two-stream Adaptive Graph Convolutional Network
7. MIB – Multiple Input Branch
8. DGNN – Directed Graph Neural Networks
9. DAG – Directed Acyclic graph
10. TAM – Temporal Attention Module
11. DG-STGCN - Dynamic Group SpatioTemporal GCN
12. STGDN - Spatial Temporal Graph Deconvolutional Network
13. PG-GCN - Pose-guided graph convolutional network
14. GECN - Graph Edge Convolutional Neural Networks
15. RA-GCN - Richly Activated Graph Convolutional Network
16. CAM - Class activation maps

LIST OF SYMBOLS

1. H[l+1] and H[l] – Hidden layers
2. W[l]- Weight matrix
3. A* – Normalized adjacency matrix
4. f_{in} – Input features
5. f_{out} – Output features
6. N_i – Set of neighboring nodes
7. v_i, v_j – i^{th} and j^{th} nodes of a graph
8. X – Input feature vector
9. C_{in} – Input channels
10. T – Count of skeleton frames
11. V – Count of total body joints
12. $\otimes$ - Element-wise multiplication
13. A_t – Temporal attention tensor
14. A_s – Spatial attention tensor
15. D_p – Learnable attention map
16. $\sum$ - Summation
17. $\mathcal{L}$ - Cross Entropy Loss
18. Z_{ik} – Normalization factor
19. $\in$ - element of
20. y_{ij} - ground truth for the i^{th} sample and j^{th} class
21. p_{ij} – forecasted probability of the i^{th} sample belonging to j^{th} class

CHAPTER 1

INTRODUCTION

1.1 Overview

Human action recognition (HAR) [3] is one of the most crucial and active research areas. HAR is concerned with predicting or classifying the action being performed by human beings. In the past few years, several approaches have been explored to efficiently understand the human actions performed. Predicting human actions help in many real-world applications like behavior analysis, gaming, video understanding, video retrieval, and human-robot interaction. There are different data modalities [4] [5] [6] [7] that are explored to improve the prediction for human actions, which include optical flow, RGB images, depth, and body skeletons [8]. Among them, body skeletons are increasingly used due to their compact and action-focused nature. Skeletons are three-dimensional or two-dimensional coordinate representations of human body joints. Skeletons are found in graph formations, where the graph's node reflect the skeleton's joints whereas the edges of the graph indicate the many connections between various body joints. Actions can be identified from the different motion patterns of the joints of the body by leveraging the graphical nature of the skeletons.

Deep Learning (DL) based techniques are widely investigated to effectively predict and recognize human actions. DL approaches include exploring Recurrent Neural Networks (RNN), Graph Convolutional Networks (GCN) [9], long short-term memory (LSTM) networks, Convolutional Neural Networks (CNN), etc. Models using RNN or LSTM [8] [10] [11] architectures suitably model temporal dynamics. Architectures using CNNs [12] [13] [14] [15] reorganize the body joint coordinates into a 2D map. Both types of architectures result in high model complexity and are unable to profit from the skeleton data's non-Euclidean nature. GCNs are formulated from CNNs and work on graphs by inspecting the neighboring graph nodes. The skeletons are present in a spatiotemporal graphical format. The spatial nature is represented by the natural joint relations, while the connection of the same joints over multiple time frames represents the temporal nature of the skeleton. GCNs can model skeleton-based data in the form of graphs and help to recognize human actions.

GCNs [16] [17] [1] have become quite prominent in the field of skeleton-based action recognition. Using deep feed-forward architectures, these approaches use graph convolutional networks to successfully capture the spatiotemporal characteristics inherent in human skeletons. The substantial memory usage and computational overhead required by GCN-based techniques, on the other hand, provide a considerable problem. To address these difficulties, researchers have investigated a variety of strategies targeted at improving GCN memory efficiency. Weight pruning, network compression, quantization, and low-rank approximation are examples of these methodologies. These strategies are designed to decrease the number of parameters and processes, hence increasing memory utilization and computing performance. While all the strategies have shown promise in terms of improving efficiency, action recognition is still computationally expensive when a large number of body skeletons are processed. There is a rising demand for the development of compact and lightweight network architectures especially optimized for action recognition based on skeletons to solve memory and computational efficiency constraints. Such architectures should find a balance between model complexity and performance, allowing for efficient body skeleton processing while retaining high accuracy in action recognition tasks.

We present an attention-based model for human action recognition in this study that uses both temporal and spatial attention modules to improve identification accuracy. The temporal attention module selects the most informative frames from a sequence of skeletons, capturing the action's critical temporal dynamics. Following that, the spatial attention mechanism highlights the most significant joints within the selected frames, emphasizing their distinguishing characteristics. The computed attention scores are then used to select frames, allowing the identification of the skeletons with the highest attention values. We efficiently utilize both temporal and spatial relationships in the skeletal data by incorporating the attention modules into a graph convolutional network. The temporal and spatial attention mechanisms together improve the efficiency of human action recognition based on skeletons, resulting in further accurate and robust identification results.

1.2 Problem Statement

The major goal is to build and improve a model that can recognize and interpret human actions using skeletal data. Human action recognition is significant in a variety of applications, including sports analysis, human-computer interaction, and video surveillance. Traditional techniques frequently depend on RGB video data, which can be affected by lighting and occlusions. By capturing human actions using the spatial configuration of joints, action recognition based on skeletons provide a more robust and efficient alternative. However, skeleton-based action identification faces a number of obstacles. It is still difficult to extract relevant information from skeletal data and properly capture the spatiotemporal dynamics of human motions. Furthermore, existing approaches usually process the body skeletons in the entire sequence that represents the action performed. This strategy, however, is inefficient in terms of computing time and memory utilization. Our primary focus will be to thoroughly investigate and analyze various Graph Convolutional Network (GCN) approaches in order to provide novel solutions to the problem. We shall aim to bring about novel changes to overcome these challenges by digging into the intricacies of GCN-based techniques.

1.3 General Concepts Involved

In this section, we will discuss the key principles that will serve as the foundation for our analysis: skeleton-based data, graph convolutional networks, and attention processes. Our goal in diving into these concepts is to lay a solid foundation for the subsequent analysis of our proposed model. By understanding the complexity of skeleton-based data representation, the principles of graph convolutional networks, and the importance of attention approaches in improving model performance, we can lay the groundwork for a comprehensive study of our novel approach.

1.3.1 Skeleton-based data

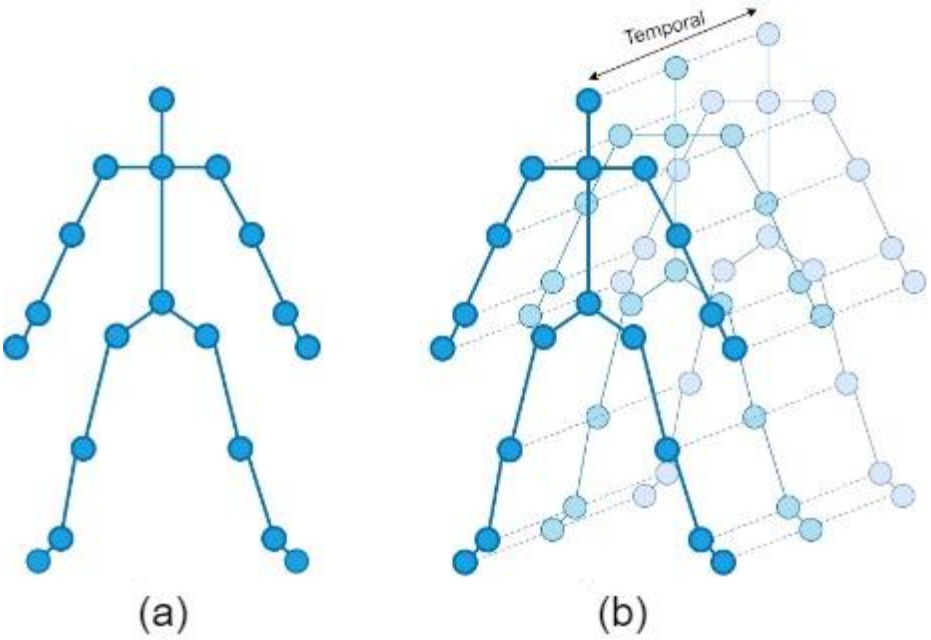

Figure 1: *(a) Spatial and (b) temporal representation of skeleton data* [8].

Action recognition using skeleton-based data [1] has gotten a lot of attention lately. Skeleton-based data depicts the human body as a collection of linked joints that create a skeletal framework. The spatial coordinates of the joints, which can be either 3D locations or 2D projections depending on the sensor used to record the data, are utilized to represent them. Each joint in 3D skeleton data is represented by a three-dimensional coordinate (x, y, z) indicating its position in 3D space. This picture indicates the relative locations and orientations of the bodily components. The joints in 2D skeleton data are represented by their 2D projections on a plane, which are commonly collected through a depth sensor or a camera. Each joint's (x, y) coordinates show its position on the picture plane. Skeleton-based data is usually gathered across time, yielding a temporal series of frames. Each frame depicts the skeletal structure's state at a certain point in time. The frames are taken at regular intervals, resulting in a time series of joint locations. The temporal aspect of skeleton-based data is critical for capturing human body dynamics and motion patterns. Recognition of movements, gestures, and postures is possible by analyzing the time dynamics of joint positions and spatial interactions.

Skeletons are represented as graphs, with joints represented by nodes and edges representing natural connectedness between joints. Every joint in the skeletal sequence is represented by the node set. Every joint represents a different component of the body, such as the head, elbows, wrists, hips, shoulders, knees, and ankles. The edge set is subdivided into two splits. The first split contains edges that connect joints inside the same frame, showing immediate or natural connectedness between joints within a single snapshot of the skeletal data. Edges, for example, connect nearby joints within a single frame, such as the shoulder and elbow or the knee and ankle. The second edge set split captures connections between the same joint in successive frames. These edges illustrate the time-dependent interactions between the joint locations. An edge, for example, may connect the shoulder joint in frame (t) to the shoulder joint in frame (t+1), illustrating the joint's movement. Skeleton-based data is resistant to lighting, motion rates, scene modification, and camera views, and is therefore increasingly utilized. The underlying structure and connectedness of the human body are conveyed by representing skeletal data as a graph. This graph-based model is helpful because it is resistant to changes in illumination, motion rates, scene changes, and camera views. This robustness enables strong analysis and identification of human movements and postures even in diverse and difficult contexts. Skeleton-based data has grown in popularity because of its capability to capture important aspects of human movement while remaining unaffected by external variables.

1.3.2 Graph Convolutional Network

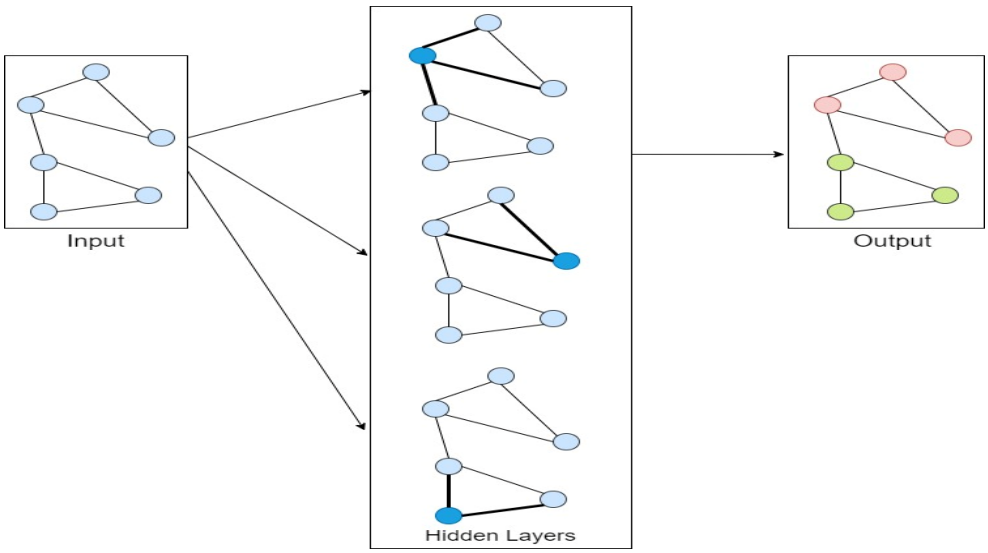

Figure 2: *Graph Convolutional Network* [9].

Graph convolutional networks (GCNs) [9] are a variant of Convolutional Neural Networks (CNN), which help to generalize graph-structured data. GCNs operate in a manner similar to CNNs by inspecting the neighboring nodes. However, the input is in non-Euclidean structural form data, with each node having varying numbers of connections. Nodes and their connections (edges) with other nodes are represented with the help of an adjacency matrix, which is then introduced to the forward propagation equation. The model learns information about node connection by adding an adjacency matrix to the equation of forward propagation. As shown in fig 2, the different layers of GCN are defined as follows:

1) Input Layer:

   The initial input to the GCN is the graph's node features. These characteristics can include node labels, attributes, or embeddings, which indicate the qualities of every node in the graph. Along with the node features, adjacency matrix is given as the input. The adjacency matrix represents the graph's connectedness by stating which nodes are related to each other.

2) Graph Convolution Layer:

   Neighborhood aggregation is used in each GCN layer to acquire information from a node's adjacent nodes. To aggregate the characteristics of nearby nodes, an adjacency matrix is utilized. A weighted sum or concatenation of the surrounding node characteristics can be used in the aggregation procedure. The same set of weights is shared across all nodes during neighborhood aggregation, allowing parameter efficiency for the model to generalize well to unseen nodes. A learnable weight matrix is then used to convert the aggregated features. The aggregated features are subjected to linear operations in this transformation, allowing the model to learn more expressive representations. To induce non-linearity, a non-linear activation function, like ReLU, is frequently applied after the transformation.

3) Hidden Layers:

   To record more intricate interactions, many GCN layers can be placed on top of each other. Each GCN layer's output is considered as the input to the following layer. The model may capture information from nodes at increasing distances in the network by stacking multiple layers.

4) Output Layer

The GCN creates a representation for the entire graph after the graph convolutional layers, capturing the relationships and characteristics of the nodes in the frame sequence. The layers' output is then sent into a fully connected layer, which is followed by a SoftMax activation function. This enables the model to categorize the action in the video sequence. The number of action classes are represented by the number of units in the fully connected layer.

For propagation of information through a graph with l^{th} hidden layer, denoted by H[l], the equation of forward pass is represented as:

$$H[l+1] = \sigma(W[l]H[l]A') \tag{1.1}$$

In eq 1.1, the next and the current hidden layers are represented by H[l+1] and H[l] respectively, A' represent the normalized adjacency matrix. W[l] depicts the weight matrix. The adjacency matrix is required to be normalized to prevent exploding/ vanishing gradients and numerical instabilities. The skeletal data is presented as graphs, with nodes as joints and the natural connectivity between nodes as edges, which are modeled with the help of GCNs. Neighbor nodes of each joint in the skeleton are divided into three partitions: a) the nodes that are nearer to the center of gravity and associated to the root nodes, b) the actual root node and c) the remaining nodes connected to the root nodes.

In a basic Graph Convolutional Network (GCN) architecture with 2 hidden layers, both hidden layers are placed on top of each other. To enhance the node representations, each hidden layer performs neighborhood aggregation, weight sharing, and feature modification. The initial input to the first hidden layer consists of graph node characteristics and the adjacency matrix. Using the adjacency matrix, the first hidden layer aggregates the characteristics of surrounding nodes. It records local connection patterns as well as information from surrounding nodes. The aggregated node features are applied with the weight matrix. This operation changes the node representations and transforms the aggregated features. It is multiplied by the aggregated features to compute the modified features for each node. This transformation facilitates the model to learn the significance and combination of neighbor information for each node in the graph.

Finally, non-linear activation function is applied to obtain updated node representations. The updated node representations are then served as the second hidden layer input, along with the adjacency matrix. A similar process is used in the second hidden layer as the first hidden layer. The adjacency matrix is used to perform neighborhood aggregation, which takes into account the updated node representations received from the previous layer. The neighborhood features are then combined and modified using a learnable weight matrix and non-linear activation function. This transformation phase aids in the capturing of higher-order dependencies and more complicated node interactions.

1.3.3 Attention

Attention mechanisms are effective deep learning approaches for increasing focus on select components or areas of input data and capturing their relevance. These mechanisms seek to simplify complicated tasks by selectively attending to relevant sections of input while discarding less significant information. Attention may be seen as a method of allocating resources or cognitive emphasis to select sections of the information. Instead of equally processing all input, attention mechanism facilitates the model to focus its computing resources on the most relevant or salient components. The basic principle of attention is to give weights or priority scores to various portions of the information depending on their relevance to the task in hand. These weights represent the value or significance of each input component. Attention processes prioritize relevant information by assigning greater weights to essential components and lower weights to less important ones. Attention modules were initially utilized in encoder-decoder designs, such as machine translation jobs. When creating the target sequence, the attention mechanism enables the decoder to selectively focus on distinct regions of the source sequence. This increases the model's capacity to capture long-term relationships and successfully align input-output combinations. However, attention mechanisms have gained popularity and are now used in a variety of fields, including human action recognition. In the context of action recognition, attention mechanisms can be used to emphasize key spatial or temporal portions of the input video frames or skeleton-based data that are most significant for detecting distinct actions. Attention models can successfully capture essential features or motion patterns critical for discriminating between distinct actions by focusing on distinguished body parts or temporal segments.

CHAPTER 2

RELATED WORK

2.1 Related Work

Recently, there have been many advancements made for the task of human action recognition based on skeletal data [1] [18] [19]. Graph convolutional networks (GCN) have contributed enormously to generalizing graph-based skeleton data. Convolution operations cannot exploit the non-Euclidean nature of the skeleton graphs and therefore, are not that well-defined. However, GCNs utilize the graphical nature and model dynamic graphs for large-scale human skeleton data. Initially, Spatial-Temporal Graph Convolutional Networks (ST-GCN) is introduced by Yan *et al.* [1] to model the dynamics of data based on skeletons. ST-GCN was developed through a series of skeletal graphs, in which the nodes depict the skeleton's joints. Temporal and spatial edges are introduced, where the inherent connectivity of the joints is highlighted by spatial edges and the connection of the same joints throughout subsequent time steps is denoted by temporal edges. For implementing graph convolutions, the weights exchanged are defined using an adjacency matrix by the different nodes of the graph. ST-GCN contains 9 layers of spatial-temporal graph convolutions and takes the skeleton graphs as input. ST-GCN is the initial model to model skeleton-based data using graph convolutions.

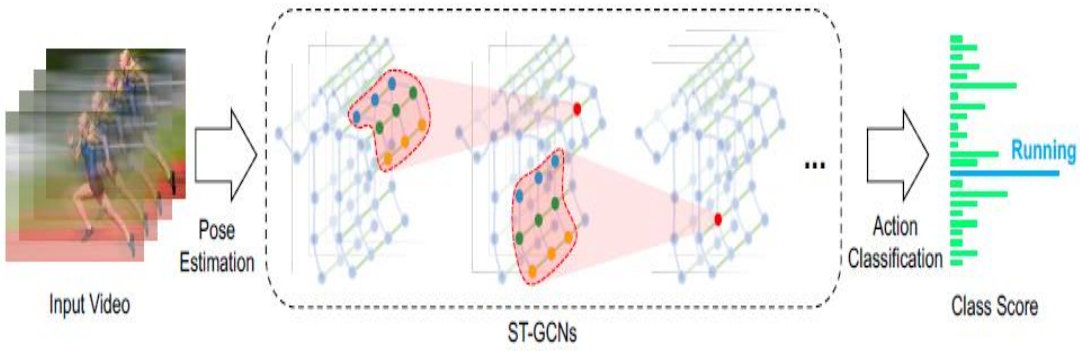

Figure 3: *ST-GCN applied to analyze spatiotemporal skeleton sequences of videos* [1].

Shi *et al.* [18] proposed a technique called two-stream adaptive graph convolutional network (2s-AGCN) to improve the performance of spatiotemporal graph convolutional networks (ST-GCNs) for skeleton-based action recognition. The goal of 2s-AGCN was to enable flexible learning of the graph structure for different GCN layers, enhancing accuracy in detecting human actions. 2s-AGCN uses information from the bones that connect two joints in addition to the joint information provided by the skeleton joints. Joint information, which represents the locations and orientations of particular joints, is called first-order information. Bone-related information, on the other hand, is called second-order information since it captures the lengths and orientations of the bones that connect the joints. Since the model incorporates both first and second-order information, it is able to efficiently represent the skeleton-based data, capturing both the local interactions between joints and the global structural properties of the human body. 2s-AGCN improves the model's flexibility and efficiency by taking into account both first and second-order information. It allows the model to incorporate the discriminative characteristics and temporal dynamics that are critical for action recognition. By allowing the network to adapt to the hierarchical relationships of different GCN layers, the model's performance is increased even more.

Shi *et al.* [20] proposed directed graph neural networks (DGNN) in w hich a directed acyclic graph (DAG) is used to represent the skeleton data based on bones and joints. Information related to joints, bones, and their relationships are modeled using DGNN. DGNN has numerous layers that are supplied vertices and edges' characteristics. The properties of vertices and edges are modified in each layer based on their nearby vertices and edges. The lowest layers retrieve local vertex and edge information, whereas the higher layers retrieve more global information. Further, the input graphs to the directed graph block are adaptively learned for flexible graph construction and enhancing action recognition. Finally, motion information is extracted from consecutive frames and a two-stream framework is formed which accesses the information related to bone and joint, which is then fused together to give the output.

Cheng *et al.* [21] proposed Shift-GCN which explored the shift operations on the graph convolutions. Flexible receptive fields are provided with Shift-GCN for both temporal and spatial graphs. The information on surrounding nodes is shifted to the

present node using shift graph operation which is non-local, in which a node's field of reception includes the whole skeleton graph. For the temporal graph, the adaptive shift operation is used to adaptively adjust the receptive field. STGCN is used as the backbone network for Shift-GCN in which the regular temporal graph convolution is replaced by the temporal shift graph and the conventional spatial graph convolution is replaced by spatial shift graph.

To further boost the efficiency, ShiftGCN++ [22] is proposed, which is a lightweight action recognition model. ShiftGCN++ incorporates 4 techniques namely lightweight architecture search, dynamic shift graph convolution, explicit spatial position encoding, and margin ReLU distillation. Lightweight architecture search aids in the development of lightweight architecture. Spatial modeling is enhanced using spatial location encoding explicitly. Dynamic shift graph helps to reduce the computational cost while margin ReLU distillation helps to increase the model performance by knowledge transfer from a big instructor network to a small student network, where ShiftGCN is considered as the instructor network.

Chen *et al.* [23] proposed a technique in which posture information is integrated with graph convolutional networks. The authors address the problem of successfully exploiting the intrinsic posture information in skeleton sequences to enhance discriminative ability of GCN-based models. The suggested technique includes the usage of a pose-guided graph convolutional network (PG-GCN) to direct the graph creation process. In particular, the posture information is utilized to determine the graph's edge connections, reflecting the spatial interactions between skeletal joints. To capture local and global relationships through time, the PG-GCN design combines both spatial and temporal GCN layers. Furthermore, during the feature aggregation phase, a pose-guided attention method is used to attend to important joints while inhibiting irrelevant ones. The authors demonstrate the efficiency of combining posture information and applying GCNs in capturing temporal and spatial correlations, resulting in enhanced recognition accuracy. The technique provided by the authors open up new possibilities for human action recognition tasks using pose-guided graph convolutional networks.

Miao *et al.* [24] proposed a unique graph convolutional operator optimized for skeleton-based action recognition challenges. By including the central difference operation into the graph convolution operation, the authors solve the difficulty of properly capturing the dynamic information found in skeletal sequences. This operator combines the difference between neighboring frames into the graph convolution process, allowing the model to capture motion-related details. To manage the variable number of nodes (joints) in different actions, the suggested technique additionally makes use of graph pooling and graph un-pooling processes. The authors emphasize the necessity of taking both temporal and spatial information into account in skeleton-based action identification, and gives insights into the efficiency of the central difference operator in capturing motion dynamics from series of skeletons.

Shi *et al.* [25] proposed a skeleton-based action recognition model using a multi-stream adaptive graph convolutional network (AGCN). An adaptive graph convolutional operation-based approach is proposed by the authors to handle the difficulty of properly capturing both spatial and temporal connections in skeletal sequences. The proposed AGCN architecture is made up of numerous streams, each of which focuses on a different component of the data. By leveraging various graph architectures and graph convolutional layers, these streams capture various degrees of spatial and temporal information. The adaptive graph convolutional operation alters the graph connections dynamically based on the input data, letting the model to understand the relevance of distinct joints and their interactions in an adaptable manner. Furthermore, the authors discuss a self-attention mechanism that improves the discriminative strength of learnt characteristics by highlighting significant joints and suppressing irrelevant ones. The model highlights the usefulness of adaptive graph convolutions and multi-stream architectures in capturing spatial and temporal connections, as well as the importance of self-attention mechanisms in boosting recognition accuracy.

Zhang *et al.* [19] proposes Graph Edge Convolutional Neural Networks (GECNs) as a unique solution for skeleton-based action recognition. The authors address the problem of describing the interactions between edges in skeletal data in order to reflect spatial interdependence. The GECN design uses graph convolutional layers to gather information from surrounding edges and spread it throughout the graph.

The model can collect fine-grained spatial information and develop discriminative features for action detection by taking edge connections into account. The authors also present an attention mechanism that weights the value of individual edges according to their relevance to the action being done, allowing the network to focus on discriminative inputs. The authors emphasize the importance of modelling edge interactions in skeleton data, as well as the efficiency of graph convolutional neural networks and attention mechanisms for capturing spatial dependencies and boosting action detection accuracy.

Song *et al.* [26] presented a novel technique termed richly activated Graph Convolutional Network (RA-GCN) to improve the resilience of action recognition models. The technique employs a multi-stream architecture to investigate discriminative characteristics across all skeleton joints, hence decreasing susceptibility to non-standard skeletons. When opposed to previous streams, each stream in the GCN is in charge of learning characteristics from less active joints. Class activation maps (CAM) are utilized to measure the activation degrees of skeletal joints, and data from inhibited joints is only passed to the next stream. This method yields extensive characteristics that cover all active joints. The presented technique considerably reduces performance deterioration due by synthetic occlusion and jittering, managing occluded and disturbed joints well.

Peng *et al.* [27] uses a Spatial Temporal Graph Deconvolutional Network (STGDN) to provide a unique technique for skeleton-based human action recognition. The authors use a graph deconvolutional technique to solve the difficulty of capturing both temporal and spatial connections in skeleton sequences. The STGDN design encodes spatial information with graph convolutional layers and captures temporal dynamics with deconvolutional layers. By propagating information backward in time, the deconvolutional layers are responsible for recreating the temporal evolution of the skeletal data. The model can successfully capture fine-grained motion patterns and discriminative characteristics for action detection by taking both temporal and spatial elements into account. Furthermore, the authors suggest a spatial temporal graph pooling method that minimizes dimensionality while keeping critical information. The authors underline the necessity of adding graph deconvolutional layers for capturing temporal dynamics in skeletal sequences and underscores the efficiency of the suggested spatial temporal graph pooling procedure.

Lee *et al.* [28] presented an architecture named as Hierarchically Decomposed Graph Convolutional Network (HD-GCN) as well as a unique Hierarchically Decomposed Graph (HD-Graph). Each joint node in the HD-GCN architecture is efficiently decomposed into numerous sets, allowing the extraction of important architecturally nearby and distant edges. These edges are then used to build an HD-Graph, which includes them in the same semantic areas as the human skeleton. The authors develop an attention-guided hierarchy aggregation (A-HA) module to emphasize the HD-prominent Graph's hierarchical edge sets. Furthermore, they use a six-way ensemble technique that only uses the joint and bone streams, ignoring any motion stream.

Su *et al.* [29] presented demonstrates an unsupervised method to skeleton-based action recognition. The authors suggest a two-step approach that includes the PREDICT and CLUSTER phases. They develop a unique unsupervised prediction method in the PREDICT phase that learns motion patterns from unlabeled skeletal sequences. This technique uses a temporal encoder-decoder network to estimate future frames based on previous frames, making it easier to acquire discriminative motion representations. They use clustering techniques in the CLUSTER phase to organise the expected motion patterns into action clusters. Following that, the clusters are labelled depending on the majority voting of the associated acts. By utilizing temporal prediction and clustering techniques, the proposed framework presents a viable strategy for unsupervised action recognition, bypassing the requirement for labelled training data.

Zhang *et al.* [30] incorporates context-aware graph convolutions for skeleton-based action recognition. The authors aim to improve identification accuracy by leveraging contextual information surrounding joints in skeletal data. They presented a context-aware graph convolutional network (CAGCN) architecture that takes into consideration both local and global contexts in the skeleton's graph representation. By utilizing a dynamic context-aware adjacency matrix, the CAGCN efficiently captures the spatial interdependence between joints. This matrix is dynamically updated to weight joint connections based on their contextual importance. In addition, the authors present a context-aware aggregation module that aggregates the properties of surrounding joints while taking their context into account. This allows the model to focus on important data while suppressing noise or unnecessary data. The CAGCN

model improves recognition accuracy by effectively capturing and exploiting contextual information in the skeletal data by introducing context-awareness into graph convolutions.

Zhang *et al.* [31] The authors propose a Semantics-Guided Neural Network (SGNN) architecture that uses semantic information in skeletal sequences to increase recognition performance while decreasing computational complexity. The Spatial Temporal Graph Convolutional Network (ST-GCN) and the Semantic-Guided Module are the two main components of the SGNN (SGM). The ST-GCN detects spatial and temporal correlations in skeletal data, allowing for successful feature extraction. The SGM makes use of semantic information by assigning a semantic label to each joint, offering direction to the network during training. This semantic guidance directs the network's attention to discriminative characteristics while reducing feature extraction redundancy. In addition, the authors present an efficient inference technique that uses semantic guidance to accomplish action recognition with low computing complexity. The SGNN is an effective solution for efficient and accurate skeleton-based human action recognition applications due to the merging of semantics-guided modules with an efficient inference method.

Heidari and Iosifidis [2] proposed a temporal attention block (TAM), which extracts the skeletons that are the most informative. The architecture was based on ST-GCN, with spatial convolutions in the initial layers and spatiotemporal convolutions in the later part. In TAM, attention maps are produced which extract each skeleton's average feature value in the series. The attention maps are then sorted in a decreasing order to determine the most informative skeletons, which are then introduced in the further layers. Extracting the most informative skeletons helps to improve computational efficiency. Duan *et al.* [32] proposed a Dynamic Group SpatioTemporal GCN (DG-STGCN) consisting of spatial and temporal modules, DG-GCN and DG-TCN. Skeleton data is dynamically modeled spatially and temporally in a group-wise manner. For DG-GCN, learnable coefficient matrices are enabled for inter-joint spatial modeling while for DG-TCN, different receptive fields are adopted which further fuses joint-skeleton motion patterns to model dynamic temporal information. Additionally, uniform sampling is explored for temporal data augmentation which substantially improved the performance of the model and avoided the possibility of overfitting.

CHAPTER 3

PROPOSED MODEL

3.1 About the Model

In the below section, the proposed human action recognition model which is based on skeletal data is discussed. The model leverages a graph-based representation of the skeleton data and graph convolutional layers to capture temporal and spatial connections between joints. To improve its ability to focus on salient and relevant aspects in the skeletal data, the model utilizes an attention module. The model uses the mechanism of temporal and spatial attention modules to improve recognition. The temporal attention module captures the most informative frames from a sequence of skeletons. The spatial attention mechanism emphasizes the most informative joints from the frames highlighted. Frame selection is then performed to select the skeletons with the highest attention scores. By including the attention mechanism, the model is able to gather and leverage the most relevant information, boosting its overall performance in skeleton-based action recognition. In the following subsections, the network's components are discussed along with the network architecture and loss function.

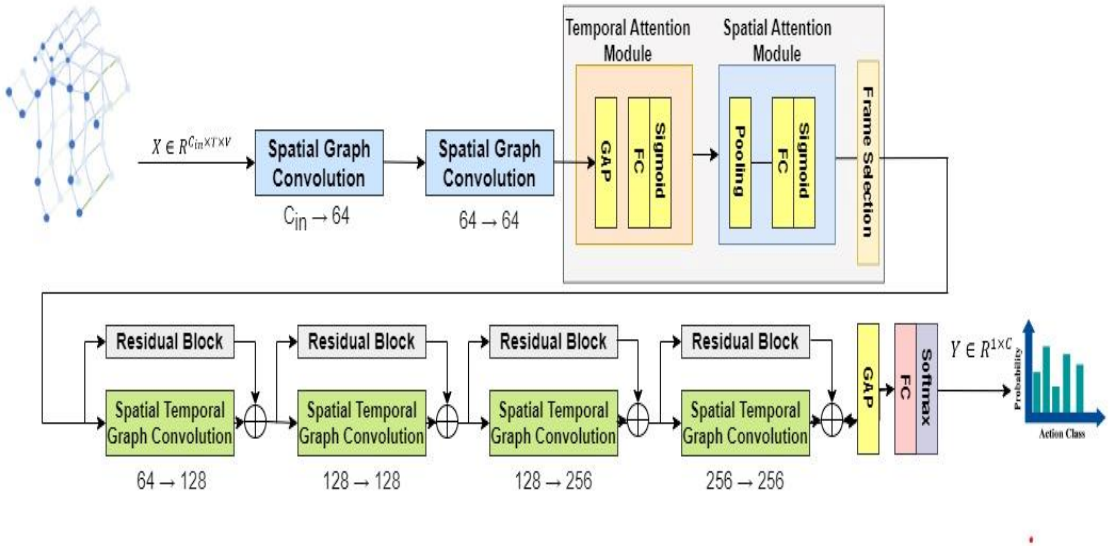

Figure 4: *Network architecture representing the different blocks in the model proposed.*

3.1.1 Spatial Graph Convolution Block

Spatial Graph Convolution block focuses on capturing spatial relationships. It operates by considering the nodes in the network as skeletal joints and the edges as connections that describe spatial interactions between these joints. The block aggregates information from nearby nodes in order to capture local interactions and dependencies. Input to the spatial graph convolution block is a set of joint characteristics. Spatial convolution for input joint i, having input feature f_{in}, is defined as:

$$f_{out}(v_i) = \sum_{v_k \in N_i} \left(\frac{1}{Z_{ik}}\right) f_{in}(v_k) w(l_i(v_k)) \quad (3.1)$$

In eq 3.1, N_i represents the set of neighboring vertices that are connected to vertices across different frames. The output features are represented by f_{out} for the node v_i. $l_i(v_k)$ maps the neighborhood nodes of vertex v_i to one of the three partitions. Z_{ik} is the normalization factor, used to normalize the contribution of all the neighborhood sets. The weight matrix w is a unique matrix associated with all the neighborhood sets. The block accepts as input the joint characteristics, which are commonly depicted as a tensor of shape (N, C_{in}, T, V), where N denotes the batch size, T represents the sequence length, C_{in} represents the number of input channels, and V represents the number of joints. The first step is to define the joint's spatial connections. Typically, an adjacency matrix is used to depict pairwise interactions between the joints. The adjacency matrix specifies which joints are linked and how strong those links are. The block then runs a graph convolution operation, which updates the characteristics of each joint by aggregating information from its nearby joints. Non-linear transformations are used after the graph convolution function to incorporate non-linearity and improve discriminative capability of the features. The feature vector, $X \in R^{N \times C_{in} \times T \times V}$, is given as the input, where N represents the batch size, C_{in} represents the channels used as input, V denotes the count of body joints, and T denotes the count of skeleton frames. Therefore, spatial convolution is defined as:

$$X' = \mathrm{ReLU}\left(\sum_p (A'_p \otimes D_p) X W_p\right) \quad (3.2)$$

In eq 3.2, A'_p is the adjacency matrix that is normalized and is established by combining the 3 neighboring subsets. The learnable attention map, D_p, emphasizes the adjacency

matrix components, the weight matrix W_p is responsible for transforming the node features in each partition. Finally, X' is the output obtained after the spatial convolution of the input feature, X.

3.1.2 Temporal Convolution Block

To capture the dynamics of temporal information in the skeletal data, the temporal convolution block is introduced in the model. The block takes as input the output of the spatial graph convolution block. To include each node's temporal neighbors, the neighborhood of each vertex is extended to include the temporal dimension. In particular, each node in the preceding and following skeleton frames is linked to the same node, resulting in a temporal neighborhood size of 2 for each node. To process this expanded neighborhood, a 2D convolution is performed to output of the spatial convolution block, denoted as X'. The inserted kernel size, represented as Kt, determines the convolution operation's temporal receptive field. The temporal convolution block aggregates the features of each individual body joint at various time steps by using a fixed kernel size, allowing the model to capture the dynamics of temporal dimension and patterns contained in the skeletal data. The temporal convolution block is essential for simulating the dynamics of temporal information within skeletal data. The block improves the model's capacity to capture and evaluate the temporal dependencies and the changes of the skeletal sequences by adding the temporal neighbors and using the 2D convolution operation with a predefined kernel size.

3.1.3 Temporal Attention Module

Temporal attention is used to detect relevant frames within a sequence of frames. The focus is on recognizing critical frames within a sequence of frames. Its goal is to emphasize and recognize the frames that make a significant contribution to the overall interpretation of an action. To accomplish this, the temporal attention module computes the average activation over all joints and channels for each frame. This stage aggregates the collective information inside each frame, indicating its overall significance. The model learns the weights associated with each frame by passing the aggregated frame-level activations through a linear layer. A sigmoid activation function is then applied to these weights which produces attention weights for each frame. These attention weights

serve as a mask, selectively amplifying or suppressing specific frame activations. The model can successfully focus on the frames that are deemed significant or informative for the given action by applying attention weights to the input tensor.

$$A_t = \text{Sigmoid}(W_t H(l)) \tag{3.3}$$

where W_t is the learnable transformation matrix for temporal attention, the l^{th} hidden layer output is denoted as H(l) and the temporal attention tensor is A_t. The input tensor is given the temporal attention tensor to modulate the activations of frames according to their attention weights.

3.1.4 Spatial Attention Module

Spatial attention is concerned with identifying important joints within the attended frames received through the temporal attention mechanism. The objective is to identify the joints in the temporal attention module-highlighted frames that contain the most discriminative information for identifying distinct actions. The spatial attention module accomplishes this by computing the average activation for each joint over all frames and channels. The module captures the joint's total contribution by aggregating information across frames. The joint-level activations are then transferred to a linear layer, which allows the model to learn the weights associated with each joint. These weights are then fed into a sigmoid activation function, which produces attention weights that indicate the relative significance of each joint. The generated attention weights operate as a mask, selectively adjusting the activations of the attended frames. The model can dynamically enhance or lessen the activations of individual joints based on their relevance by adding attention weights to the attended frames.

$$A_s = \text{Sigmoid}(W_s M(l)) \tag{3.4}$$

In eq 3.4, M(l) is the temporal attention output, W_s is learnable transformation matrix for spatial attention, and A_s is spatial attention tensor. To find a subset of most informative skeletons from a given sequence of skeletons, the skeletons are sorted in a decreasing order according to their attention weights. These selected skeletons are then incorporated into the network's subsequent layers for additional processing and analysis.

The model prioritizes and focuses on the most relevant and informative skeletons within the sequence by identifying the skeletons with the highest attention values. This enables the network to focus on the skeletons that contribute the most to the understanding and identification of the action being conducted.

3.1.5 Network Architecture

Skeleton data is fed as the input to the model, which comprises the coordinate locations of all the joints of the skeleton. The model proposed is made up of six graph convolutional layers and one attention module. Initially, the 2 spatial graph convolutional layers perform spatial convolutions on the data, converting it to a 64-dimensional feature space. The attention module consists of temporal and spatial attention mechanism that selects discriminative features from a subset of skeletons based on the 2nd graph convolutional layer's output. The feature dimension goes from 64 to 128 and finally to 256 in the final two levels of the model, increasing the representation capability. The last four graph convolutional layers combine spatial and time convolutions, following that, batch normalization and ReLU activation is performed. The model incorporates the ResNet module into the last four graph convolutional layers to effectively utilize the input skeleton data. With a stride of 2, the third and fifth graph convolutional layers, which incorporate temporal convolution, serve as pooling layers. Each skeleton sequence's enhanced spatiotemporal characteristics are routed via a pooling layer with a global average, yielding a 256-dimensional output feature vector. Finally, human actions are classified using a fully linked layer that includes a SoftMax classifier. To reduce classification error, the model trains end-to-end via backpropagation.

3.2 Loss Function

Cross entropy loss is also known as log loss, that is commonly employed in deep learning and machine learning, particularly for classification problems requiring many mutually exclusive classes. It calculates the difference between the predicted probability distribution and the actual probability distribution. In binary classification scenarios, cross entropy loss computes the negative logarithm of the anticipated probability for the true class. It extends to average the loss over all classes in multi-class classification by comparing the projected probability distribution with the one-hot encoded actual label distribution. This loss function aggressively penalizes confident inaccurate predictions, pushing models to learn precise and well-calibrated probability distributions across classes. Models try to improve classification performance by decreasing cross entropy loss during training. It is represented as:

$$\mathcal{L} = -\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij} \log\left(p_{ij}\right) \tag{3.5}$$

In eq 3.5, N is the count of samples in the batch, C is the count of classes, y_{ij} is the ground truth for the i^{th} sample and j^{th} class, and p_{ij} is the forecasted probability of the i^{th} sample belonging to the j^{th} class.

CHAPTER 4

METHODOLOGY

4.1 Dataset

We experiment with the model on a large-scale indoor action recognition dataset, NTU-RGB+D [32]. It is the most comprehensive datasets for annotations in 3D for human action recognition tasks. The RGB and depth video clips acquired by Microsoft Kinect v2 sensors are included in the dataset. It covers 60 action classes covering over 56,880 action samples, with the actions carried out by 40 distinct test participants. The action classes cover a wide variety of daily actions, including walking, waving, clapping, sitting down, getting up, and playing musical instruments. It consists of both individual actions and interactions between two subjects. The annotations that are obtained by the Kinect depth sensors offer 3D joint positions (X, Y, Z). A total of 25 joints is present in each subject in the skeleton series.

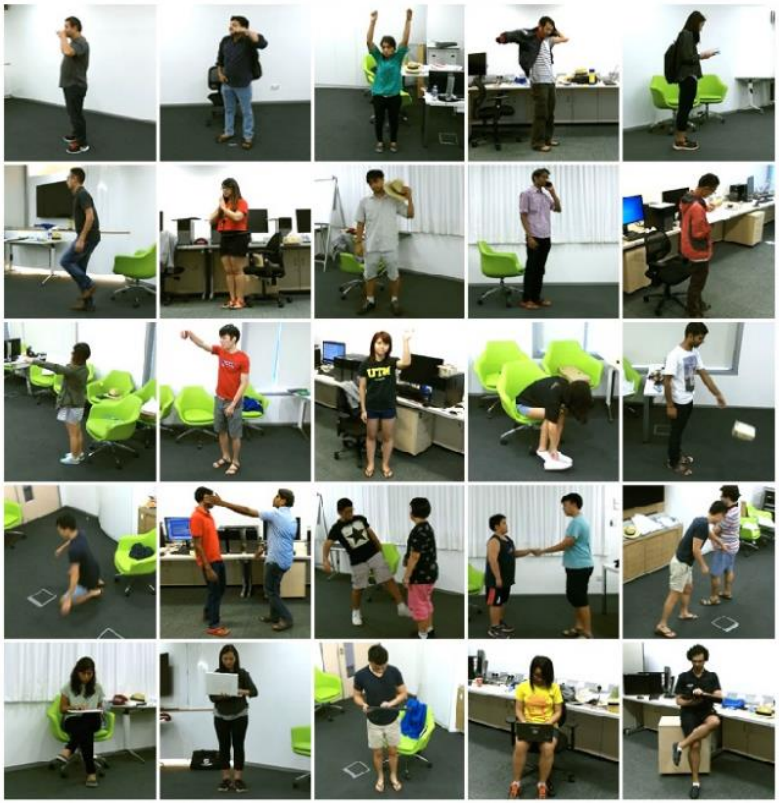

Figure 5: *NTU-RGB+D dataset samples* [32].

The authors recommend for 2 evaluation benchmarks.:

1. **Cross-view (X-view):** In the X-view benchmark, three different cameras are used to capture the videos. The training set comprises of clips captured from camera 3 and camera 2 and consist of 37, 820 video clips. The test set involves videos captured from camera 1 and consists of 18, 960 video clips.
2. **Cross-subject (X-sub):** X-sub benchmark focuses on cross-subject action recognition. The dataset is split into two parts: training and testing, with every part including distinct participants. The test set contains 16560 videos while the training set has 40320 videos.

For each sample in the dataset, there are 300 frames. If there are less than 300 frames in the sample, the sequence of frame is replicated until the sample has 300 frames. A tensor with the size of (3x300x25) is taken as the input information where 3 is the number of channels of RGB videos, 25 is the count of skeletal joints and 300 is the frame count.

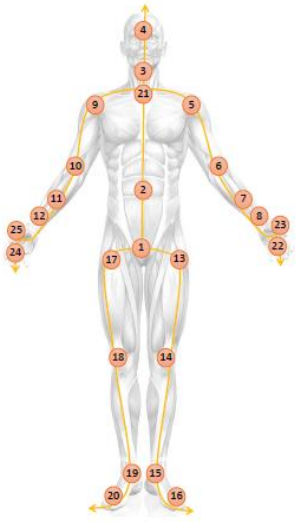

Figure 6: *NTU-RGB+D skeleton data with 25 joint coordinates* [32].

4.2 Training Metrics

Our model is trained on an Intel Xeon CPU @2.30GHz with NVIDIA TESLA P100 GPU. Pytorch deep learning frameworks are used to carry out the experiments. Initially, a learning rate of 0.1 is set. The learning rate is then gradually lowered at epochs 30 and 40 by 10, with the total count of epochs set as 50. Through

backpropagation, the SGD optimizer is utilized to improve the model's parameters at the time of training with a cross-entropy loss function. A weight decay of 0.0001 is applied. The NTU-RGB+D dataset uses a batch size of 32.

4.3 Performance Metrics

We assess action recognition performance using the NTU-RGB+D dataset's Top-1 and Top-5 accuracies for cross-view and cross-subject benchmarks both. The training sets of both benchmarks are used to train the model and the accuracies are reported on the validation sets. The Top-1 and Top-5 accuracies and weights in each epoch for both benchmarks are continuously stored, using which model checkpoints are created. On the cross-subject benchmark, our method produced Top-1 and Top-5 accuracy of 83.58 percent and 97.07 percent, respectively, and Top-1 and Top-5 accuracy of 91.22 percent and 98.85 percent on the cross-view benchmark. The change in accuracy is very obvious at epoch 30, and after epoch 40, the accuracy gradually stabilizes.

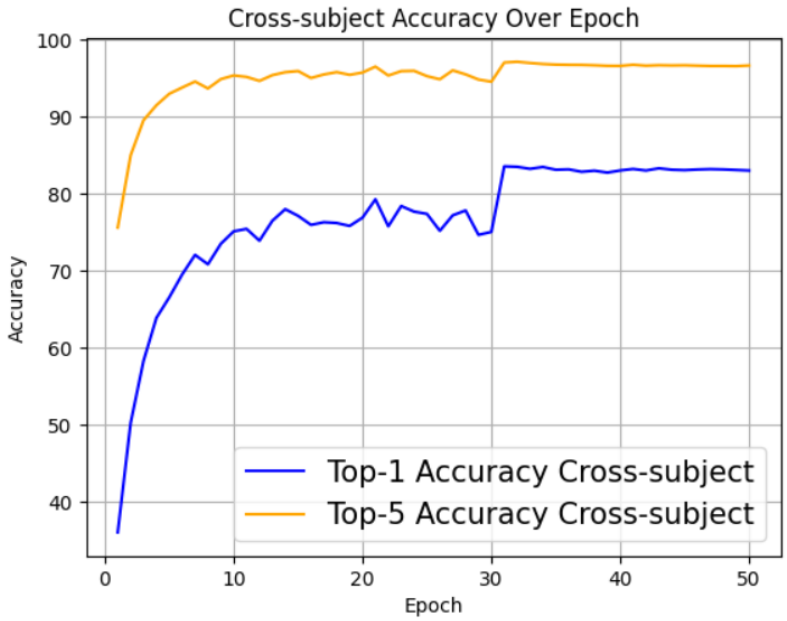

Figure 7: *Top-1 and Top-5 accuracy of NTU-RGB+D's cross-subject benchmark.*

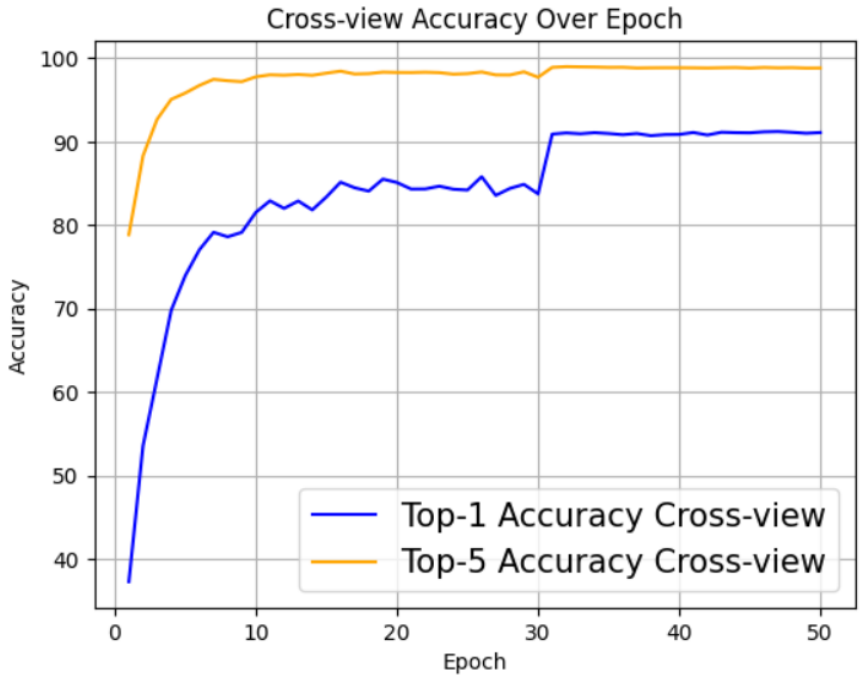

Figure 8: *Top-1 and Top-5 accuracy for NTU-RGB+D's a cross-view benchmark.*

4.4 Qualitative and Quantitative Evaluation

On the NTU-RGB+D dataset, the performance of our proposed method is compared with other skeleton-based techniques. The performance is estimated using the dataset's cross-view and cross-subject benchmarks. The top-1 accuracy of both benchmarks is compared with other methods. The proposed model's accuracy is compared with DeepLSTM [32], TCN [13], C-CNN+MTLN [33], STA-LSTM [34], ST-GCN [1] and DPRL+GCNN [35]. Table 1 shows the outcomes of the suggested approach on the NTU-RGB+D dataset, which shows that our method surpassed all the other methods by a considerable proportion. Our technique greatly surpasses the other methods. RNN-based and CNN-based methods are highly complex and are unable to model the non-Euclidean nature of skeletal data adequately. As a result, RNN-based DeepLSTM and STA-LSTM utilize LSTMs to model temporal information. However, the methods overlook the rich spatial information that is critical for human action recognition. Conversely, CNN-based TCN and C-CNN+MTLN methods fail to capture fine-grained spatial information, and hence limit the accuracy of the models.

Table 1: *Quantitative comparison of existing notable methods with the model.*

Model	Cross-subject (%)	Cross-view (%)
DeepLSTM [32].	59.8	66.8
TCN [13].	74.3	83.1
STA-LSTM [34].	73.4	81.2
ST-GCN [1].	81.5	88.3
DPRL+GCNN [35].	83.5	89.8
C-CNN+MTLN [33].	79.6	84.8
Ours	**83.6**	**91.2**

Furthermore, GCN-based methods outperform RNN and CNN-based methods. ST-GCN is recognized as the benchmark for GCN-based methods and exhibits significant advancements compared to CNN and RNN-based approaches. It serves as a foundational reference point, showcasing substantial improvements in performance within the research field. However, ST-GCN is outperformed by our method over a large margin in cross-view and cross-subject benchmarks both. While our method is comparable to the DPRL+GCNN method on the cross-subject benchmark, it outperforms the NTU-RGB+D dataset's cross-view benchmark. In comparison with other methods, our method yields a good result which showcases that the usage of attention modules significantly enhances the performance of the human action recognition model based on skeletons. Our model successfully acquires dynamic features and enhances the accuracy of human action recognition.

CHAPTER 5

CONCLUSION

We developed an attention-based graph convolutional network for human action recognition. Utilization of temporal and spatial attention mechanisms helped in enhancing the model's performance. The temporal module captures the important frames within a sequence of skeletal frames. The spatial module focuses on identifying important joints from the attended frames obtained from the temporal attention module. Graph convolutional network (GCNs) is applied to capture the spatiotemporal dynamics of data based on skeletons for human action recognition. We evaluated the model on a widely used NTU-RGB+D benchmark, assessing its top-1 and top-5 accuracies on the dataset's cross-view and cross-subject benchmarks. In comparison to other skeleton-based models, our model performed better. We observed a significant difference in performance between RNN and CNN-based methods and our method. Furthermore, our method outperformed other GCN-based models, demonstrating the benefit of incorporating spatial and temporal attention mechanisms to graph convolutional network which enhanced the model's accuracy and efficiency.

References

[1] S. Yan, Y. Xiong and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the AAAI conference on artificial intelligence*, 2018.

[2] N. Heidari and A. Iosifidis, "Temporal attention-augmented graph convolutional network for efficient skeleton-based human action recognition," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021.

[3] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," 2014.

[4] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *Advances in neural information processing systems*, 2014.

[5] L. Wang, Y. Qiao and X. Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015.

[6] D. Tran, L. Bourdev, R. Fergus, L. Torresani and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015.

[7] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang and D. Lin, "Temporal action detection with structured segment networks," in *Proceedings of the IEEE international conference on computer vision*, 2017.

[8] Y. Du, W. Wang and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015.

[9] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[10] J. Liu, A. Shahroudy, D. Xu and G. Wang, "Spatio-temporal lstm with trust gates for 3d human action recognition," in *Computer Vision--ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, 2016.

[11] S. Song, C. Lan, J. Xing, W. Zeng and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *Proceedings*

of the AAAI conference on artificial intelligence, 2017.

[12] H. Liu, J. Tu and M. Liu, "Two-stream 3d convolutional neural network for skeleton-based action recognition," *arXiv preprint arXiv:1705.08106*, 2017.

[13] T. S. Kim and A. Reiter, "Interpretable 3d human action analysis with temporal convolutional networks," in *2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW)*, 2017.

[14] Q. Ke, M. Bennamoun, S. An, F. Sohel and F. Boussaid, "A new representation of skeleton sequences for 3d action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[15] B. Li, Y. Dai, X. Cheng, H. Chen, Y. Lin and M. He, "Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep CNN," in *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 2017.

[16] M. Niepert, M. Ahmed and K. Kutzkov, "Learning convolutional neural networks for graphs," in *International conference on machine learning*, 2016.

[17] W. Hamilton, Z. Ying and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, 2017.

[18] L. Shi, Y. Zhang, J. Cheng and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.

[19] X. Zhang, C. Xu, X. Tian and D. Tao, "Graph Edge Convolutional Neural Networks for Skeleton-Based Action Recognition," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[20] L. Shi, Y. Zhang, J. Cheng and H. Lu, "Skeleton-based action recognition with directed graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.

[21] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng and H. Lu, "Skeleton-based action recognition with shift graph convolutional network," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[22] K. Cheng, Y. Zhang, X. He, J. Cheng and H. Lu, "Extremely Lightweight Skeleton-Based Action Recognition With ShiftGCN++," *IEEE Transactions on Image Processing*, 2021.

[23] H. Chen, Y. Jiang and H. Ko, "Pose-Guided Graph Convolutional Networks for Skeleton-Based Action Recognition," *IEEE Access*, 2022.

[24] S. Miao, Y. Hou, Z. Gao, M. Xu and W. Li, "A Central Difference Graph Convolutional Operator for Skeleton-Based Action Recognition," *IEEE Transactions on Circuits and Systems for Video Technology,* 2022.

[25] L. Shi, Y. Zhang, J. Cheng and H. Lu, "Skeleton-Based Action Recognition With Multi-Stream Adaptive Graph Convolutional Networks," *IEEE Transactions on Image Processing*, 2020.

[26] Y.-F. Song, Z. Zhang, C. Shan and L. Wang, "Richly Activated Graph Convolutional Network for Robust Skeleton-Based Action Recognition," *IEEE Transactions on Circuits and Systems for Video Technology,* 2021.

[27] W. Peng, J. Shi and G. Zhao, "Spatial Temporal Graph Deconvolutional Network for Skeleton-Based Human Action Recognition," *IEEE Signal Processing Letters*, 2021.

[28] J. Lee, M. Lee, D. Lee and S. Lee, "Hierarchically Decomposed Graph Convolutional Networks for Skeleton-Based Action Recognition," *arXiv preprint arXiv:2208.10741,* 2022.

[29] K. Su, X. Liu and E. Shlizerman, "PREDICT & CLUSTER: Unsupervised Skeleton Based Action Recognition," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[30] X. Zhang, C. Xu and D. Tao, "Context Aware Graph Convolution for Skeleton-Based Action Recognition," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[31] P. Zhang, C. Lan, W. Zeng, J. Xing, J. Xue and N. Zheng, "Semantics-Guided Neural Networks for Efficient Skeleton-Based Human Action Recognition," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[32] A. Shahroudy, J. Liu, T.-T. Ng and G. Wang, "Ntu rgb+ d: A large scale dataset for 3d human activity analysis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[33] Q. Ke, M. Bennamoun, S. An, F. Sohel and F. Boussaid, "A New Representation of Skeleton Sequences for 3D Action Recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[34] S. Song, C. Lan, J. Xing, W. Zeng and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *Proceedings of the AAAI conference on artificial intelligence*, 2017.

[35] Y. Tang, Y. Tian, J. Lu, P. Li and J. Zhou, "Deep progressive reinforcement learning for skeleton-based action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[36] H. Duan, J. Wang, K. Chen and D. Lin, "DG-STGCN: Dynamic Spatial-Temporal Modeling for Skeleton-based Action Recognition," *arXiv preprint arXiv:2210.05895*, 2022.

PAPER NAME

Thesis_Anshula_Sharma_2k21_cse_07 (3).pdf

WORD COUNT

9389 Words

CHARACTER COUNT

54312 Characters

PAGE COUNT

41 Pages

FILE SIZE

805.9KB

SUBMISSION DATE

May 30, 2023 4:45 PM GMT+5:30

REPORT DATE

May 30, 2023 4:46 PM GMT+5:30

12% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 7% Internet database
- 7% Publications database
- Crossref database
- Crossref Posted Content database
- 9% Submitted Works database

Excluded from Similarity Report

- Bibliographic material
- Cited material