
Article:
Mohottala, S. orcid.org/0000-0002-6196-2161, Gawesha, A. orcid.org/0000-0001-8946-5629, Kasthurirathna, D. orcid.org/0000-0001-8820-9033 et al. (2 more authors) (2025) Spatio-temporal graph neural network based child action recognition using data-efficient methods: A systematic analysis. Computer Vision and Image Understanding, 259. 104410. ISSN 1077-3142

https://doi.org/10.1016/j.cviu.2025.104410

Computer Vision and Image Understanding

journal homepage: www.elsevier.com

Spatio-Temporal Graph Neural Network based Child Action Recognition using Data-Efficient Methods: A Systematic Analysis

Sanka Mohottala[a,b,**], Asiri Gawesha[a], Dhrashana Karshutrirathna[a], Charith Abhayaratne[d], Pradeepa Samarasinghe[c]

[a]*Department of Software Engineering, Faculty of Computing, Sri Lanka Institute of Information Technology, Sri Lanka*
[b]*Department of Electrical and Electronic Engineering, Faculty of Engineering, Sri Lanka Institute of Information Technology, Sri Lanka*
[c]*Department of Information Technology, Faculty of Computing, Sri Lanka Institute of Information Technology, Sri Lanka*
[d]*School of Electrical and Electronic Engineering, The University of Sheffield, United Kingdom*

ABSTRACT

This paper presents implementations on child activity recognition (CAR) using spatial-temporal graph neural network (ST-GNN)-based deep learning models with the skeleton modality. Prior implementations in this domain have predominantly utilized CNN, LSTM, and other methods, despite the superior performance potential of graph neural networks. To the best of our knowledge, this study is the first to use an ST-GNN model for child activity recognition employing both in-the-lab, in-the-wild, and in-the-deployment skeleton data. To overcome the challenges posed by small publicly available child action datasets, transfer learning methods such as feature extraction and fine-tuning were applied to enhance model performance. As a principal contribution, we developed an ST-GNN-based skeleton modality model that, despite using a relatively small child action dataset, achieved superior performance (94.81%) compared to implementations trained on a significantly larger (x10) adult action dataset (90.6%) for a similar subset of actions. With ST-GCN-based feature extraction and fine–tuning methods, accuracy improved by 10%-40% compared to vanilla implementations, achieving a maximum accuracy of 94.81%. Additionally, implementations with other ST-GNN models demonstrated further accuracy improvements of 15%-45% over the ST-GCN baseline. The results on activity datasets empirically demonstrate that class diversity, dataset size, and careful selection of pre-training datasets significantly enhance accuracy. In-the-wild and in-the-deployment implementations confirm the real-world applicability of above approaches, with the ST-GNN model achieving 11 FPS on streaming data. Finally, preliminary evidence on the impact of graph expressivity and graph rewiring on accuracy of small dataset-based models is provided, outlining potential directions for future research.

1. Introduction

Human Action Recognition (HAR) is a research area focused on detecting and estimating human actions or movements. It plays an important role in many application domains such as surveillance (Vishwakarma and Agrawal, 2013), autonomous driving (Lu et al., 2020), healthcare (Keskes and Noumeir, 2021), entertainment (Shotton et al., 2011), video retrieval (Hu et al., 2007), human-robot interactions (Rodomagoulakis et al., 2016), sports analysis (Martin et al., 2018), Virtual reality (Maqueda et al., 2015).

HAR data can be classified as visual or non-visual modalities based on interpretability (Sun et al., 2022), where visual includes RGB, skeleton, depth, and event streams, while non-visual includes audio, acceleration, radar, and WiFi. While action recognition can be done using those different modali-

**Corresponding author
e-mail: divandyasm@gmail.com (Sanka Mohottala)

ties (Kawashima et al., 2017; George et al., 2020; Wang et al., 2019a), due to practicality and interpretability, our focus in this research is on RGB and skeleton modality based methods.

Child action recognition (CAR) can be considered as a subdomain in HAR and has important applications in video game development (Dong et al., 2020), early detection of autism (Zhang et al., 2021b; Zahan et al., 2023a), safety monitoring (Goto et al., 2013), object-play behavior assessment (Ramesha et al., 2022), video game development (Dong et al., 2020) and many others.

State-of-the-art (SOTA) RGB data based HAR models (Srivastava and Sharma, 2024; Piergiovanni et al., 2023; Wang et al., 2024b; Huang et al., 2023b) as well as skeleton data based HAR models (Liu et al., 2025; Zhou et al., 2024; Xie et al., 2024; Zheng et al., 2024) are developed using adult action centered datasets such as Kinetics-400 (Carreira and Zisserman, 2017), Moments in time (Monfort et al., 2019), UCF-101 (Soomro et al., 2012), Something-something-v1/2 (Goyal et al., 2017), NTU RGB+D 120 (Liu et al., 2020), kinetics-skeleton (Yan et al., 2018). Thus these models are not suitable for child action recognition (CAR) since 1) children are still in their gross and fine motor development stage (Aloba et al., 2019; Jain et al., 2016a) and 2) with skeleton modality, child-adult differences in size and anatomy result in a distribution difference between adult and child action data (Sciortino et al., 2017). These differences can be observed in Figure 1.

Furthermore, this has been verified experimentally as well with ML and DL models (Aloba et al., 2020; Olalere et al., 2021a). To overcome this, CAR models need to be developed using child action based data but lack of child action data due to ethical reasons is a major bottleneck. This research tackles this issue by focusing on data-efficient method in CAR model development.

Skeleton modality data is obtained by using multimodal sensors such as Kinect (Liu et al., 2020) or from pose-estimation methods (Cao et al., 2019; Bazarevsky et al., 2020) on RGB datasets. Compared to RGB modality and other modalities, skeleton modality has several unique advantages. Early research in perception has shown that motion trajectories of the skeleton are sufficient for humans to recognize actions (Johansson, 1973). Spiking neural network inspired by the functionality of the dorsal pathway (Liu et al., 2018a) corroborates this by obtaining on-par accuracy in benchmark datasets indicating the sufficiency of motion information for HAR.

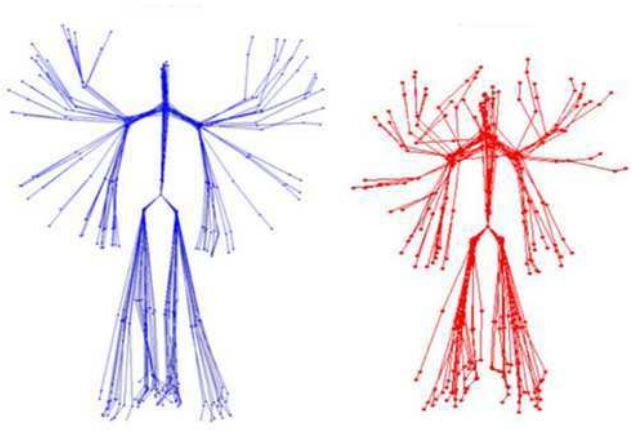


Fig. 1: Adult (blue) vs Child (red) jumping jacks action snapshots taken from a Kinect camera. *Reprinted from (Jain et al., 2016b)*

Skeleton data is not affected by background clutter, clothing, view angle or lighting hence, there is less spurious information (Liu et al., 2018b; Chen et al., 2021). This also results in a cleaner signal with less noise, making them well suited for HAR. With skeleton modality, number of features is also less compared to RGB modality making skeleton modality ideal for data-efficient HAR model development (Fukunaga and Hayes, 1989).

Skeleton data require minimal storage (170x less storage compared to RGB data) and can be processed quickly, so they are well suited for real-time applications and storage-intensive applications (Qin et al., 2022; Lin et al., 2020). Additionally, this preserve the privacy and removes ethnic bias in HAR models (Moon et al., 2023).

Due to the natural graph structure of skeleton modality, graph neural networks (GNNs) is the best suited neural network (NN) architecture for skeleton based HAR and the empirical results show the superiority of GNN methods over other NNs and handcrafted feature-based methods (Yan et al., 2018; Zheng et al., 2024).

Transfer learning (TFL) methods have resulted in remarkable performance with low-data regions across different NN architectures, including GNNs (Yaras et al., 2024; Wu et al., 2021; Kooverjee et al., 2022). TFL has also obtained good performance with HAR models (Dhekane and Ploetz, 2024; Keskes and Noumeir, 2021). Characteristics of source and target datasets such as quality, quantity, diversity and similarity have non-trivial effect on TFL as evident in previous studies (Ehrig et al., 2024; Jain et al., 2022). This motivated us to do a systematic study on the effect of source-target datasets with different TFL methods in CAR.

Architectural features that act as inductive biases result in preferences for some solutions over the others thus resulting in data-efficiency (Romero, 2024). This has been observed across CNN, LSTM, Transformers, GNNs etc (Zhang et al., 2024; Potapczynski et al., 2024; Kayhan and Gemert, 2020; Farina and Slade, 2021). Motivated by this, a detailed study of different spatio-temporal graph neural network (ST-GNN) architectural features was done on CAR.

Due to practical reasons, in HAR and CAR applications, RGB vision systems (i.e., mobile phone, CCTV etc.) are preferred over depth sensors (i.e., Kinect, RealSense). To utilize skeleton modality, pose estimation methods (Cao et al., 2019; Sárándi et al., 2020; Bazarevsky et al., 2020) are used with RGB data. Multi-person contexts, long-shot views, jitter, occlusion and truncation are challenges (Song et al., 2021; Shi et al., 2023) that needs to be addressed with this approach. Motivated by this, we extend the previous experiments to pose estimation based skeleton data as well and conduct experiments to analyse the effect of pose degradation on model performance.

Real-time implementation is vital for HAR in domains like surveillance and autonomous driving (Noor et al., 2024; Deng et al., 2023). Balancing accuracy and latency is also needed. Deployment also demands handling streaming input (Huan et al., 2023) and out-of-distribution data (Roy et al., 2022). To this end, we extend pose-based child action recognition to out-of-distribution streaming scenarios.

Based on the research gaps identified, this work presents the following contributions:

- To the best of our knowledge, this is the first systematic analysis of skeleton modality-based CAR using GNN architectures, covering in-the-lab, in-the-wild and in-the-deployment action recognition.
- We conduct a detailed analysis of transfer learning in skeleton-based HAR with GNNs, examining how (1) source dataset quality, quantity, and diversity, and (2) target dataset class distribution and task similarity impact TFL outcomes.
- We demonstrate that generic HAR models do not translate well to CAR. We analyze how architectural properties, including model complexity, graph expressivity, graph wiring approaches, occlusion robustness, and higher-order information utilization, influence CAR performance.
- We observe that ST-GNN model performance on CAR positively correlates with child age, suggesting that lower CAR performance stems from both dataset limitations and developmental differences in motor skills.
- We show that the limited accuracy of pose-estimation-based models is due to inherent pose-estimation limitations rather than specific action classes.

The codes are made publicly available for reproducing the results and future research.[1]. The rest of the paper is organized as follows. Current state of research areas this study touches is covered in Section 2 . The ST-GNN models, learning methods, action recognition with in-the-lab, in-the-wild, and in-the-deployment datasets are covered in Section 3. Preprocessing stages, common experimental details and employed evaluation methods are discussed in Section 4. Results and interpretations of ST-GNN based in-the-lab, in-the-wild and in-the-deployment protocols are given in the Section 5 while Section 6 concludes the study with some future research directions.

[1] Codes, pre-processed data and implementation results are available from https://github.com/sankamohotttala/ST_GNN_HAR_DEML

2. Related Work

This section surveys prior work in GNNs, HAR, and CAR to provide the necessary context for our methodology and highlight existing research gaps.

2.1. Graph Neural Network

Graph Neural Networks (GNNs) were introduced by the original researchers to work with graph-structured data, iteratively updating node representations by aggregating information from their neighbours (Scarselli et al., 2009; Gori et al., 2005; Gallicchio and Micheli, 2010). Early graph convolution networks (GCN) such as ChebNet and Spectral CNN were based on graph signal processing (Defferrard et al., 2017; Bruna et al., 2014). Improved ChebNet based GCN (Kipf and Welling, 2016) archived SOTA performance on graph benchmark datasets (Wu et al., 2021). Message passing graph neural network (MPGNN) (Gilmer et al., 2017) unified GNN architectures under a principled message passing framework.

GraphSAGE (Hamilton et al., 2018) achieved SOTA on inductive tasks, GAT (Veličković et al., 2017) improved expressivity via attention-based neighbor weighting, and GIN (Xu et al., 2019) further improved expressiveness to the level of the Weisfeiler-Lehman (1-WL) test. These GNN architectures have been applied to tasks such as few-shot image classification, semantic segmentation, visual reasoning, recommendation systems and predict molecular properties (Wu and Xin, 2025; Aflalo et al., 2023; Wang et al., 2023, 2024a; Batatia et al., 2025). Quantization and pruning have also improved the practicality of GNNs for real-world, resource-limited applications (Chen et al., 2023).

Graph rewiring mitigates oversmoothing and oversquashing by adjusting the graph structure, either before training (Attali et al., 2024) or during it (Zhu et al., 2022). Although effective across various tasks, the impact of these GNN features on HAR is still unclear (Feng and Meunier, 2022). We address this by experimentally evaluating which architectural features best support HAR performance.

2.2. Human Action Recognition

Initial skeleton based HAR models were based on handcrafted features such as covariance matrices of joint trajectories, lie group represented skeleton and hidden markov model (Xu et al., 2016; Hussein et al., 2013; Zhou et al., 2009). With RGB modality, handcraft methods based on spatio-temporal volume, spatio-temporal interest point (STIP) and trajectory were used (Wang et al., 2011; Bobick and Davis, 2001; Laptev, 2005). Deep learning methods leveraging RGB data and optical flow have employed (2+1)D and 3D CNN architectures for action recognition (Hara et al., 2018; Feichtenhofer, 2020; Wang et al., 2015; Simonyan and Zisserman, 2014). Similarly, these architectures have also been applied to skeleton sequence data (Li et al., 2018; Huynh-The et al., 2020; Liu et al., 2017b). With skeleton modality, HAR has obtained SOTA using GNN methods on benchmark datasets (Yan et al., 2018; Hu et al., 2022; Shi et al., 2019; Xie et al., 2024; Zhou et al., 2024). GNN-based HAR models have also been developed for real-time implementation (Noor et al., 2024; Dong et al., 2020; Chi et al., 2025).

Current SOTA HAR models are mainly evaluated on large-scale (e.g., Kinetics-400, NTU RGB+D) or medium-scale (e.g., UCF-101, HMDB) datasets (Carreira and Zisserman, 2017; Liu et al., 2019, 2017a; Soomro et al., 2012; Kuehne et al., 2011). However, detailed experiments on how performance varies with dataset size are lacking, limiting insights into model data efficiency. To address this, we experiment with different GNN architectures on CAR datasets.

2.3. Child Action Recognition

Research show that child and adult actions differ in variation, stability, and intensity, and these differences are perceptible enough for observers to categorize individuals as children or adults based on skeletal motion (Aloba, 2019; Aloba and Anthony, 2021; Jain et al., 2016b).

Early work in CAR used ML methods (Tsiami et al., 2018; Rehg et al., 2013). Neural network based methods such CNN, LSTM have been used for CAR with children aged 4-5 on private RGB modality based datasets (Zhang et al., 2021b; Suzuki

et al., 2019; Olalere et al., 2021b; Amemiya et al., 2020; Huang et al., 2023a). Recent work has used GNN and transformer based approaches on publicly available skeleton modality based datasets as well (Mohottala et al., 2022b; Kim et al., 2023; Zahan et al., 2023b).

While there are claims about publicly available CAR datasets (Lemaignan et al., 2018; Rajagopalan et al., 2013; Kim et al., 2023; Olalere et al., 2021b), some of them were not available and others were not complete. Thus, in this work, fully available CAR datasets were considered and the experiments were conducted on them (Vatavu, 2019; Mohottala et al., 2022a).

3. Methodology

The ST-GNN-based Child Action Recognition (CAR) implementations address four key aspects: architecture, dataset selection, learning methods, and hyper-parameter tuning. Experiments were systematically designed across these dimensions to achieve state-of-the-art performance, as outlined in the following subsections. Figure 2 provides an overview of the implementations. Graph Neural Networks (GNNs) are introduced in Section 3.1 as a foundation for the ST-GNN architectures (Section 3.2), starting with ST-GCN. Transfer learning methods are described in Section 3.3, and heuristic hyper-parameter tuning is detailed in Section 4.2. Detailed descriptions of in-the-lab (Section 3.4), in-the-wild (Section 3.5), and in-the-deployment (Section 3.6) implementations are provided next.

3.1. Graph Neural Network (GNN)

GNNs are used with graph structured data to learn a feature embedding that can be used in downstream tasks such as node classification, link prediction, graph classification etc.

Many GNN architectures can be understood within the message passing graph neural network (MPGNN) framework. As shown in Figure 3, the process can be explained using a target node u (e.g., node A in Figure 3), following three main steps.

- Message ($\mathcal{F}_m$): This transforms the input embeddings h_v^k of all neighbourhood nodes $\mathcal{N}(u)$ of target node u where k is the layer. This is generally a dense layer or a multi-layer perceptron (MLP).
- Aggregation ($\mathcal{F}_A$) : This is a permutation-invariant function that aggregates the messages from previous step. In many of GCN ,GAT and other architectures, this is either summation or average function.
- Update ($\mathcal{F}_U$): This function takes the output from the previous step as well as a transformed target node's feature vector $\mathcal{F}_t(u)$ as inputs. Generally, $\mathcal{F}_u$ is an non-linear function resulting in updated representation of node u, denoted by h_u^{k+1}. $\mathcal{F}_t$ could also be the zeros function, the identity function, a dense layer or an MLP.

These steps are given in equation 1 where node u is the target node. A specific implementation (Hamilton, 2020) of MPGNN is given in equation 2 where dense layers are used for $\mathcal{F}_t$ and $\mathcal{F}_m$ and summation function is used for $\mathcal{F}_A$.

$$h_u^{(k+1)} = \mathcal{F}_U\left(\mathcal{F}_t\left(h_u^{(k)}\right), \mathcal{F}_A\left(\left\{\mathcal{F}_m\left(h_v^{(k)}\right) \,\middle|\, v \in \mathcal{N}(u)\right\}\right)\right), \tag{1}$$

$$\mathrm{h}_u^{(k+1)} = \sigma\left(\mathrm{W}_{self}^{(k)}\mathrm{h}_u^{(k)} + \sum_{v\in\mathcal{N}(u)} \mathrm{W}_{neigh}^{(k)}\mathrm{h}_v^{(k)} + \mathrm{b}^{(k)}\right), \tag{2}$$

where

$$\mathrm{W}_{self}^{(k)},\ \mathrm{W}_{neigh}^{(k)} \in \mathbb{R}^{\mathrm{d}^{(k)}\times\mathrm{d}^{(k-1)}},$$
$$\mathrm{b}^{(k)} = \mathrm{b}_{self}^{(k)} + \sum_{v\in\mathcal{N}(u)} \mathrm{b}_{neigh}^{(k)}.$$

3.1.1. Graph Convolutional Network (GCN)

GCN (Kipf and Welling, 2016) uses $\mathcal{F}_m$ for $\mathcal{F}_t$ as well thus the resultant weight sharing makes it less expressive than MPGNN (equation 2). Transformed node embeddings ($\mathcal{F}_m$) are multiplied by a local structure specific weight resulting in equation 3. These inductive biases make GCN pay attention to the local graph structure and the simple design makes it a computationally efficient architecture:

$$\mathrm{h}_u^{(k+1)} = \sigma\left(\sum_{v\in\mathcal{N}(u)\cup\{u\}} \frac{\mathrm{W}^{(k)}\mathrm{h}_v^{(k)}}{\sqrt{|\mathcal{N}(u)|\,|\mathcal{N}(v)|}}\right). \tag{3}$$

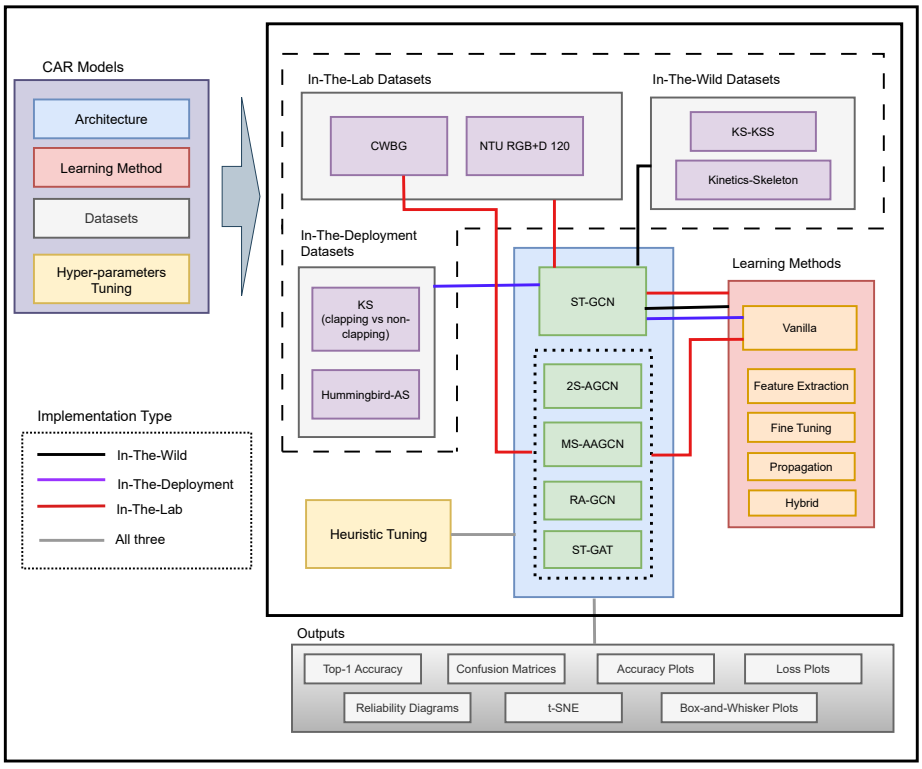


Fig. 2: Overview Diagram of Methodology

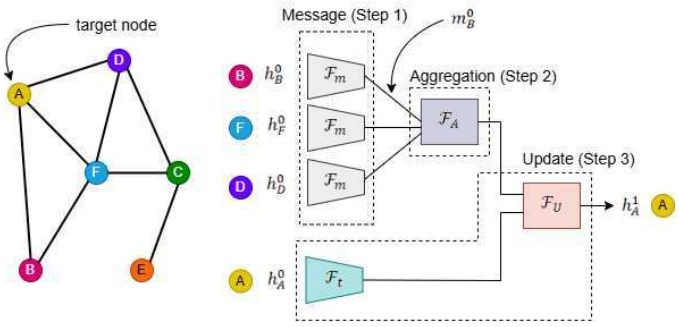


Fig. 3: MPGNN Framework

3.1.2. Graph Attention Networks (GAT)

GAT (Veličković et al., 2017) extend GCN by replacing the fixed, structure-based aggregation weights with learnable, attention-based weights, allowing the model to adaptively focus on the most relevant neighbours during message passing (equation 5). By relaxing the fixed weights, GAT becomes more expressive and offers improved structural awareness.

However, due to its weak inductive bias, GAT may underperform in low-data regimes if GCN's bias benefits representation learning in HAR tasks. Conversely, if GCN's bias hinders learning, GAT may outperform due to its higher expressivity, even with limited data.

The attention calculation equation is given in equation 4 where both W and a are learnable parameters. Use of multi-headed attention improves the expressivity even more by allowing GAT to capture diverse patterns:

$$\alpha_{u,v} = \frac{\exp\left(\mathrm{a}^{\mathrm{T}}\left[\mathrm{W}\mathrm{h}_u \parallel \mathrm{W}\mathrm{h}_v\right]\right)}{\sum_{w \in \mathcal{N}(u)\cup\{u\}} \exp\left(\mathrm{a}^{\mathrm{T}}\left[\mathrm{W}\mathrm{h}_u \parallel \mathrm{W}\mathrm{h}_w\right]\right)}, \tag{4}$$

$$\mathrm{h}_u^{(k+1)} = \sigma\left(\sum_{v \in \mathcal{N}(u)\cup\{u\}} \alpha_{u,v} \mathrm{W}^{(k)} \mathrm{h}_v^{(k)}\right). \tag{5}$$

3.2. Spatial-Temporal Graph Neural Network (ST-GNN)

We hypothesize that architectural improvements can lead to better performance, even with small datasets. Based on this assumption, we evaluated the performance of several state-of-the-art human action recognition models on child action recognition. Our implementations started with our TensorFlow-based ST-GCN model, which we used in our previous work (Mohottala et al., 2022b,a) as well. We then gradually introduced

four state-of-the-art architectures which were selected based on novel architectural components and functional codebases. Initial implementations were done to reproduce the SOTA results on benchmark HAR datasets given in their corresponding papers.

3.2.1. Spatial-Temporal Graph Convolutional Network

ST-GCN (Yan et al., 2018) architecture utilized the graph structure of human skeleton of Kinect camera output and the pose estimation output of RGB videos with the graph convolution operation. In this architecture, only the first order information of data were used such that the input contained either the real-world coordinates of kinect camera or the RGB camera pixel coordinates.

3.2.2. Other ST-GNN Implementations

With 2s-AGCN model architecture (Shi et al., 2019), rather than restricting the GNN adjacency matrix representation to the static skeleton structure, it is formulated as a combination of static skeleton structure, an adaptive graph and a data-dependant graph. Furthermore, second order information such as bone vectors at each time step are included in input, essentially giving the model a strong inductive prior.

Improving on the adaptive graph aspect of 2s-AGCN model, MS-AAGCN (Shi et al., 2020) model initializes the adaptive graph with static skeleton structure and combines it with the data-dependant graph. Furthermore, a gating mechanism is included between these two graphs to stabilize the model. Attention is also included on spatial, temporal and channel directions to improve the performance. Finally, improving on 2-stream aspect, more refined second order information is included with skeleton motion resulting in 4-streams.

RA-GCN (Song et al., 2021) introduces a novel pre-processing stage where first order, second order and second order motion information are combined as a new feature vector. With this approach previous multi-stream implementations can be combined to a single implementation. Furthermore, to make the architecture robust to occlusion, truncation and jitter, trainable mask approach is introduced with multiple streams where each stream is conditioned to give attention to a more discriminative set of joints. Finally, a new loss function is introduced to train the model without need of ensemble model as in 2s-AGCN and MS-AAGCN.

In previous methods, GCN based spatial feature extraction and 1-D convolution based temporal feature extraction were independent of each other. However in ST-GAT (Hu et al., 2022), GAT is used where receptive field is not limited to a single frame hence, short-term motion feature can also be extracted. Similar to MS-AAGCN, a 4-stream approach is used which achieves current SOTA results.

3.3. Learning Methods

The limited size of available child activity datasets can lead to poor performance in deep learning models. Improved learning methods is a promising approach to overcome this challenge. After the vanilla implementations, techniques such as fine-tuning (Tan et al., 2018) and feature extraction (Zhang et al., 2019) were used to enhance the performance of the model.

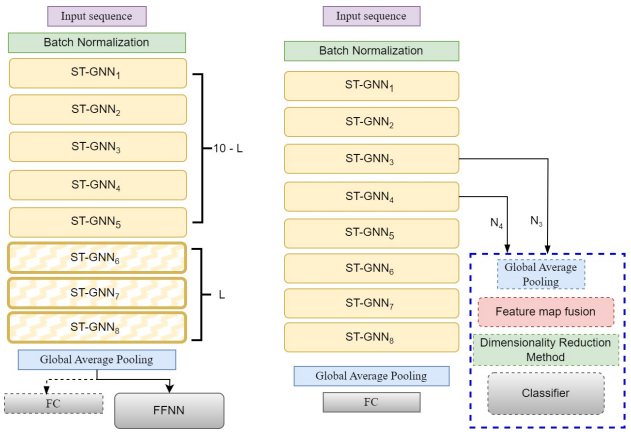
3.3.1. Vanilla Method

Initial implementations were done by training the model from scratch without any modifications. Optimizer hyper-parameter tuning was done across all protocols to achieve the best performance while architectural hyper-parameter tuning was done to observe the effect of model capacity. Experiments on the batch normalization were also done to select the best normalization method.

3.3.2. Fine Tuning Method

Fine tuning of the pre-trained ST-GCN model was done using several methods as detailed in transfer learning literature (Zhang et al., 2019). Fine-tuning model pipeline is shown in Figure 4.

- Frozen layer approach - Fine tuning L top ST-GCN layers, where $1 \leq L < 10$
- Propagation approach - Fine tuning all ST-GCN layers ($L = 10$).



(a) Fine-tuning (b) Feature extraction

Fig. 4: ST-GCN architecture and transfer learning pipeline

- Hybrid approach - In the original ST-GCN model, L top ST-GCN layers were randomly initialized where $1 \leq L \leq 10$.

    1. Hybrid-Frozen : Combines feature extraction and standard deep learning together. Bottom $10 - L$ ST-GCN layers were kept frozen.

    2. Hybrid-FineTuned : Combines fine tuning and standard deep learning together. Bottom $10 - L$ ST-GCN layers were fine tuned.

Fully connected (FC) layer was removed after the initial source dataset based model training and a randomly initialized feed forward neural network (FFNN) was connected to the global average pooling layer output to function as the classifier. FFNN architecture was selected as part of the architecture hyper-parameter tuning.

3.3.3. *Feature Extraction Method*

Inspired by (Awais et al., 2020; Yu et al., 2017), the feature representations were extracted from feature maps of the ST-GCN model as shown in Figure 4. The original ST-GCN model was then supplemented with either a flattening layer or a global average pooling (GAP) layer as the intermediate layer between the ST-GCN model and the classifier. Fusion of feature maps was done with consecutive maps from two or three layers of ST-GCN as shown by N_4 and N_3 in Figure 4. Enhancing this approach further, dimensionality reduction techniques such as principle component analysis (PCA) and truncated SVD were employed. Experiments were conducted using support vector machine (SVM), logistic regression as well as feed forward neural network (FFNN) as the classifier.

Based on the results detailed in (Mohottala et al., 2022b), FFNN was used as the classifier in the final feature extraction with the output from final ST-GCN layer. FFNN contains 3 layers, first 2 with 196 and 128 units and the final layer with units equal to the number of target dataset classes.

3.4. *In-the-Lab Action Recognition*

A literature review was carried out on the child action recognition (CAR) related datasets and the benchmark human action recognition (HAR) datasets. While the majority of the datasets used in CAR research comes from private datasets (Zhang et al., 2021b; Suzuki et al., 2019; Amemiya et al., 2020), few of those datasets were publicly available. Details of these datasets are given in Table 8 in the supplementary document. As the Child datasets, CWBG (Vatavu, 2019) and the KS-KSS (Mohottala et al., 2022a) datasets were used in this study while for benchmark datasets, which are predominantly adult data, NTU RGB+D (Shahroudy et al., 2016), NTU 120 (Liu et al., 2019) and kinetics-skeleton (Yan et al., 2018) datasets were used.

Initial experiments were conducted using CWBG and NTU datasets which were collected in carefully designed lab environments, thus we call these in-the-lab datasets. Kinect cameras were used to capture the data while actions were performed based on a prompted command rather than a natural scenario. Due to the data collection environment, there is less pose degradation causes such as occlusion and truncation. Due to the depth images taken by Kinetic cameras, resultant skeleton poses contain the real-world 3D positions as well, resulting in highly accurate pose estimations. Due to these reasons, in-the-lab data were used across all implementation at the first stage.

As the second stage, we extended several experimentations to in-the-wild data which were collected in natural settings using standard RGB cameras. As the child dataset, we used the KS-KSS (Mohottala et al., 2022a) dataset which we created and

Table 1: Action Class Coverage Across CWBG and KS-KSS Dataset Protocols

Index	Action Class	CWBG				KS-KSS			
		Full	Similar	Dissimilar	Shared	Full	Large	Balanced	Small
0	Hopscotch					✓	✓	✓	
1	Clapping					✓	✓	✓	
2	Bouncing on trampoline					✓	✓	✓	
3	Baseball throw					✓	✓	✓	
4	Climbing tree					✓	✓	✓	
5	Cutting watermelon					✓			✓
6	Squat					✓			✓
7	Pull ups					✓			✓
8	angry like a bear	✓	✓						
9	applaud	✓	✓	✓	✓				
10	climb ladder	✓	✓	✓					
11	crouch	✓	✓	✓	✓				
12	draw circle	✓	✓						
13	draw flower	✓	✓	✓					
14	draw square	✓	✓						
15	fly like a bird	✓	✓	✓					
16	hands up	✓	✓	✓	✓				
17	jump	✓	✓	✓	✓				
18	scratch like a cat	✓							
19	slice carrots	✓							
20	stand on one leg	✓		✓					
21	throw ball	✓		✓	✓				
22	turn around	✓		✓					

released publicly. As for the adult dataset, we used the kinetics-skeleton dataset. We call these datasets in-the-wild data due to the natural setup and as a result of that, considerable occlusions and truncation can be observed which became a part of our study as well. To extract skeletons, OpenPose (Cao et al., 2019), a 2D pose estimation model was used.

3.4.1. CWBG: In-The-Lab Child Dataset

CWBG (Vatavu, 2019) is a dataset recorded in a controlled lab environment using a Kinect v1 camera with approximately 10FPS frame rate. This is a balanced dataset with 15 action classes resulting in 1312 skeleton sequences which amount to 90 skeleton sequences per class and the children were asked to perform those action through verbal commands. It contains 30 children who are between the ages of 3 - 6 with a uniform gender and age distribution as shown in Table 5 where the CWBG (Vatavu, 2019) child ID is given. Group 3 contains the children between age 3-4, group 4 contains the children who are between 4-5 and final group 5 contains children between age 5-6.

3.4.2. CWBG Dataset Protocols

Following the cross subject evaluation method introduced in NTU-60 (Shahroudy et al., 2016) dataset, we also use a similar x-sub evaluation method with our protocols. We select 21 children (70%), with a uniform age and gender distribution, for the training phase and rest of the 9 children (30%) were used for testing phase. With cross subject evaluation, personal information leakage between train and test is removed and the resultant model performance can be generalized to outside personals as well. To analyse the effect of information leakage, we also devised a random split based evaluation method with similar train test distribution.

Since all these 15 classes aren't present in any large-scale dataset, comparison of child-data based model and adult-data based model with CAR is not possible. But there are five similar classes between CWBG and NTU datasets as given in the Table 2 and Table 1. We introduce CWBG-Shared protocol to compare the performance of those implementations. Following the leave-one-out cross validation (LOOCV) method, we evaluate the model performance on each child and give a robust evaluation of the model with another protocol as well.

- CWBG-Cross-Subject: The dataset is split based on subject IDs to ensure no child appears in both training and test sets, effectively preventing subject-specific information leakage.
- CWBG-Random: Randomly split the train and test sets disregarding the child ID. Stratified splitting was used to get balanced subsets with similar train-test distribution to other CWBG protocols.
- CWBG-LOOCV: Similar to LOOCV evaluation, training was done with 29 children data and tested with the remaining child. Final model evaluation was done averaging the results.

Low inter-class variations and high intra-class variations can adversely affect classification tasks. Since the children in CWBG are between 3-6, there could be differences in understanding the actions asked to perform as well as the way they perform. Due to complex nature of human actions, higher intra-class variations can always be present in a HAR dataset. Given that there are similar classes in this dataset, resulting low inter-class variations could also affect the model performance. Thus going beyond the full CWBG implementation (CWBG-Full protocol in Table 1), we develop several CWBG dataset based protocols to analyse these effects.

To analyse the effect of inter-class variations, we devise two protocols as two balanced subsets of this dataset, one with

Table 2: Similar classes in CWBG and NTU Datasets

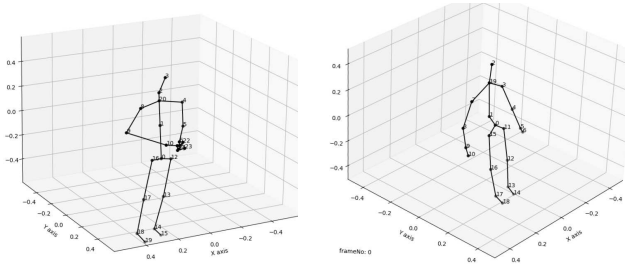
CWBG dataset	NTU dataset
Hands up	Capitulate
Jump	Jump up
Crouch	Fall down
Throw ball	Throw
Applaud	Clapping

higher inter-class variations (CWBG-Dissimilar) and the other with lower inter-class variations (CWBG-similar). Exact composition of these protocols are given in the Table 1. ST-GCN based and transfer learning based implementations were then extended to these two protocols as well. Furthermore, to analyse the effect of age and gender of children on model performance, more protocols were introduced and the details of those protocols and the subsequent experimental results are given in the supplementary document.

3.4.3. Transfer Learning: NTU as Source Dataset

NTU-120 (Liu et al., 2019) dataset was used as the source dataset with transfer learning methods in in-the-lab setting. NTU-120 dataset contains 3D skeleton sequences belonging to 94 action classes and 26 interaction classes. With NTU-120 dataset, we introduced several protocols, with varying amounts of source dataset diversity and size, to achieve improved performance with transfer learning. Furthermore, with each NTU-{44,60,120} protocol, a NTU-{44,60,120}-FRA protocol was introduced by down-sampling each skeleton sequence by selecting each of the 3rd frame.

- NTU-120: The full NTU RGB+D-120 dataset was used with a different dataset splitting method as detailed in Section 4.1 than the one proposed in (Liu et al., 2019). With this approach, we were able to remove any potential bias resulting from an unbalanced data distribution.
- NTU-60: We used the full NTU RGB+D dataset along with 11 interaction classes.
- NTU-5: This contains the 5 NTU classes that are similar to 5 CWBG classes as in Table 2.
- NTU-44: A qualitative curriculum learning inspired approach was used by analysing the confusion matrices of NTU-60/120 to select best performing classes. To reduce ambiguities introduced from spatial and temporal symmetrical classes, we kept only one such class in this subset.
- NTU-44-Best: A quantitative curriculum learning inspired approach was used by selecting the best performing 44 classes from the NTU-120 implementation.
- NTU-44-Worst: A quantitative curriculum learning inspired approach was used by selecting the worst performing 44 classes from the NTU-120 implementation.



(a) NTU skeleton (b) CWBG skeleton

Fig. 5: Skeleton structures

Since CWBG dataset is recorded with a Kinect version 1 camera, skeleton structure contains only 20 joints and there also exists structural differences between Kinect version 2 camera based NTU dataset where 25 joints are present. While GCN and GAT architectures are robust to topological variations and utilizes the feature information rather than structural information, structural improvements on NTU dataset resulted in improved performance with preliminary experiments. Based on these results, rewired 20-joint skeleton sequence version of NTU was used across all above NTU protocols in transfer learning experiments. Results are given in the supplementary document and the original structure and the modified structure are shown in the Fig 5.

3.5. In-The-Wild Action Recognition

Due to the limited availability of publicly accessible RGB-modality child action datasets, we developed the KS-KSS dataset (Mohottala et al., 2022a) for Child Action Recognition (CAR) by annotating a subset of the Kinetics-600 dataset (Carreira et al., 2018) and have made it publicly available. We primarily included motion-oriented classes in the KS-KSS dataset. To increase its complexity, we added challenging classes such as climbing tree and cutting watermelon, which involve truncation and occlusion. Classes were selected to ensure a substantial

number of child instances, making KS-KSS suitable for in-the-wild CAR scenarios.

OpenPose was used for pose estimation with KS-KSS video data since (Sciortino et al., 2017) shows that in child pose estimation, it performs better than other methods even when truncations and occlusions are present. OpenPose provides two pose estimation models called BODY_25 (i.e., OpenPose 2019 version) and COCO (i.e., OpenPose 2016 version). The BODY_25 is faster than COCO in extraction process and, its accuracy is also improved by 7% as detailed in (Cao et al., 2017).

COCO-skeleton (G_{coco}) contains 18 vertices (V_1) and BODY_25-skeleton (G_{body}) contains 25 vertices (V_2) and since $V_1 \subset V_2$ in $G_{coco} = (V_1, E_1)$ and $G_{body} = (V_2, E_2)$, G_{coco} skeleton structure could be used with the BODY_25 extracted data. Thus, skeleton extraction was done for the full kinetics600 subset using BODY_25 model.

Since kinetics-skeleton is created with COCO model and graph structure has to be same in all data, we used G_{coco} for KSS pre-trained model based implementations. In our implementations, we used kinetics-skeleton as the source dataset in transfer learning approaches. In those experiments, only 392 classes were used (kinetics-skeleton-392) since some of the data from the other 8 (Table 1) classes are used in the KS-KSS dataset.

To conduct experiments with the in-the-wild data, four protocols based on KS-KSS dataset were introduced with a 3:1 train, test split. Implementations were separately done with KS and KSS datasets, thus X in each protocol name indicating both KS and KSS. X-Full contain all 8 classes and X-large contain the five largest classes by sample size. By selecting 250 videos (X = KS) or 110 videos (X=KSS), we introduced a balanced version of this protocol. X-Small protocol contains smallest 3 classes of KS-KSS dataset. Protocol details are given in the Table 1.

3.5.1. In-The-Wild Data: Accuracy vs Confidence Evaluation

Skeleton based model implementation can be adversely affected by pose estimation process when RGB video based datasets are used. In general, pose estimation can have errors due to occlusion, truncation, and jitter. Here, we study the effect of occlusion and truncation on action recognition task using confidence value, which acts as a heuristic measure.

For each joint v, OpenPose outputs a 3-element vector containing $[x_v, y_v, c_v]$ where c_v denotes the confidence value and is within the range of [0, 1]. This confidence value resulted from the non-maximum suppression functionality in key point extraction, which can be interpreted as a probability value used as the heuristic metric to measure the effect of truncation and occlusion.

For the purpose of this study, kinetics-skeleton test set was used which is a balanced set containing approximately 50 videos per each class resulting in 19796 videos. OpenPose extracted kinetic-skeleton test set was used to obtain the confidence values. ST-GCN model trained with kinetics-skeleton train set was used to obtain the predicted class values on the kinetics-skeleton test set.

To perform this analysis, we first define the set S^y for each class y (equation 6), where S^y comprises all skeleton sequence information belonging to class y including their corresponding confidence and accuracy values.

Initial analysis was done by separating each skeleton sequence to two sets for two most active persons, where most active person is denoted by $p = 0$ and the second most active person is denoted by $p = 1$ (equation 7). Set of confidence values of each node in each of these sets is given by equation 8. In the same equation, $\mathcal{T}$ denotes the set of frames where each frame is denoted by t. Set of nodes is denoted by $\mathcal{V}$ while each node is defined by v.

$$\mathcal{S}^y = \left\{ s^{k_y} \mid k_y \in \mathcal{N}_y \right\}, \tag{6}$$

where

$$s^{k_y} = \left\{ \hat{s}_0^{k_y}, \hat{s}_1^{k_y}, \hat{y}^{k_y}, y \right\}, \tag{7}$$

$$\hat{s}_p^{k_y} = \left\{ c_v^{t,k_y,p} \mid \forall t \in \mathcal{T},\ \forall v \in \mathcal{V} \right\}, \tag{8}$$

$$\bar{k}_y = \left| \mathcal{N}_y \right|.$$

In above equations,

$\hat{y}^{k_y}$: predicted class from ST-GCN model for the skeleton sequence k_y,

$\mathcal{N}_y$: set of all skeleton sequences from class y,

k_y: index of each skeleton sequence in class y,

s^{k_y}: confidence and accuracy values for skeleton sequence k_y,

$\mathcal{S}^y$: confidence and accuracy values for all skeleton sequences in class y,

$c_v^{t,k_y,p}$: confidence value of each v node at frame t of the person p in skeleton sequence k_y,

$\hat{s}_p^{k_y}$: set of confidence values of all nodes of the person p in skeleton sequence k_y,

As the second step, analysis was done considering average classwise confidence ($\bar{c}_p^y$) as the independent variable (equation 9) and the classwise accuracy ($\bar{a}^y$) as the dependant variable (equation 10).

$$\bar{c}_p^y = \frac{\sum_{k_y \in \mathcal{N}_y} \sum_{t \in \mathcal{T}} \sum_{v \in \mathcal{V}} c_v^{t,k_y,p}}{|\mathcal{T}|\,|\mathcal{V}|\,\left|\mathcal{N}_y\right|}, \tag{9}$$

$$\bar{a}^y = \frac{\sum_{i \in \mathcal{N}_y} 1\,(y_i = \hat{y}_i)}{\left|\mathcal{N}_y\right|}. \tag{10}$$

Due to only considering the average class-wise confidence, information of confidence variance within the class is lost. To mitigate this issue an approach analogous to reliability diagram was also used (equations 11, 12).

$$\hat{s}_p^k = \left\{ c_v^{t,k,p} \mid \forall t \in \mathcal{T},\ \forall v \in \mathcal{V} \right\}, \tag{11}$$

where,

$\mathcal{K}$: set of all skeleton sequences in kinetics-skeleton test set where $|\mathcal{K}| = 19796$,

k: index of each skeleton sequence.

$c_v^{t,k,p}$: confidence value of each node v at frame t of the person p in skeleton sequence k,

$\hat{s}_p^k$: set of confidence values of all nodes of person p in skeleton sequence k.

$$\mathcal{S} = \left\{ s^k \mid k \in \mathcal{K} \right\}, \tag{12}$$

where,

$$s^k = \left\{ \hat{s}_0^k, \hat{s}_1^k, y, \hat{y}, p^k \right\}$$

$$p^k \in [0, 1]$$

In equation 12,

s^k: Confidence values $\hat{s}_0^k$, predicted class y ,actual class $\hat{y}$ and the probability values p^k for skeleton sequences k,

$\mathcal{S}$: Set of s^k for all samples in the k set,

In this approach, for each video in the test dataset ($\mathcal{S}$), average confidence value ($\bar{c}_{avg}^k$) as well as average confidence value per person($\bar{c}_p^k$) was calculated as in equation 14 and equation 13.

$$\bar{c}_p^k = \frac{\sum_{t \in \mathcal{T}} \sum_{v \in \mathcal{V}} c_v^{t,k,p}}{|\mathcal{T}|\,|\mathcal{V}|} \tag{13}$$

$$\bar{c}_{avg}^k = \frac{\sum_{p=0}^{1} \bar{c}_p^k}{2} \tag{14}$$

Based on those values, samples were assigned a bin (b) in a manner similar to histograms and the final normalized accuracy per bin (h_{acc}^b) was calculated as in equation 15. Further extending this approach, instead of accuracy, each sample's resultant probability from softmax layer (p^k) was used to calculate the final normalized probability per bin as in equation 16.

$$h_{acc}^b = \frac{\sum 1\,(y^m = \hat{y}^m)}{\left|\bar{c}_p^m\right|} \tag{15}$$

$$h_{prob}^b = \frac{\sum p^m}{\left|\bar{c}_p^m\right|} \tag{16}$$

where,

$$\forall \bar{c}_p^m \in \left[\frac{1}{20} b\,,\ \frac{1}{20}(b+1) \right)$$

$$b = \{0, 1, .., 19\}$$

3.5.2. In-The-Wild Data: RGB Modality Implementations

In the Long-term Recurrent Convolutional Network (LRCN) architecture (Donahue et al., 2017), visual features are extracted through CNNs and the temporal information from those extracted features are learnt through LSTMs. Furthermore, ImageNet 1.2M dataset (Deng et al., 2009) was used to pre-train the CNN feature extractors of the LRCN model as done in similar other HAR architectures (Wang et al., 2019b), (Ng et al., 2018). While the original model uses both RGB and Optical Flow modalities, we have restricted it to RGB modality. The CNN base of the implemented LRCN model is based on Resnet-152 architecture instead of AlexNet. This reimplementation of LRCN was used with KS-KSS dataset and a comparison of RGB-modality performance with skeleton-modality was conducted across all KS-KSS protocols.

3.6. In-The-Deployment Action Recognition

Although in-the-wild implementations use RGB video data captured under unconstrained real-world conditions, they do not adequately represent real-world deployment scenarios involving streaming inputs and potential distribution shifts between training and inference data. To address these limitations in the context of child action recognition, we designed an action recognition system (ARS) that takes streaming data as input and outputs temporally localized action predictions.

Since multi-person scenarios are expected in deployment, we first apply a YOLO-based human detector (Wang et al., 2022) to localize humans and estimate bounding boxes for each individual. The similarity of bounding boxes across consecutive frames is then used to track all humans throughout the frame sequence. For each detected human, AlphaPose (Fang et al., 2022) is employed to extract 2D pose estimations.

For action recognition, the skeleton sequences of each individual are used as input to the model. Since the input sequence length is a design parameter, we adopt a sliding window approach with a window size of n frames and a stride of d frames during both training and inference. To prevent frame backlog caused by high instantaneous latency, we employed queue-based data structures for efficient frame buffering and utilized multi-threading to parallelize key operations.

Action recognition model was trained using the ST-GCN architecture on the KS dataset for a child clapping versus non-clapping classification task. All 386 child action samples from the clapping class were used, along with 98 child action samples each from the baseball throw, hopscotch, climbing tree, and bouncing on trampoline classes for the non-clapping category. Results on these are given in the supplementary document.

Since real-world deployments often involve out-of-distribution data, we evaluated the action recognition system (ARS), incorporating the previously trained ST-GCN model, on a privately collected child action dataset from a daycare center. The recordings, captured with parental consent, include children playing with toys and interacting with adults in natural settings. A subset of the videos was manually annotated with action classes, such as clapping and high-fiving, when the action persisted for at least one second. This annotated subset is referred to as "Hummingbird-AS" in this paper. The ARS was then used to evaluate classification accuracy and inference latency on Hummingbird-AS.

4. Experiments

This section describes the quantitative experimental procedures, including dataset pre-processing, evaluation methods, and detailed implementation settings.

4.1. Data Pre-processing

Data pre-processing steps were systematically applied to in-the-lab datasets to enhance model performance. Initially, noisy data such as pseudo-skeleton sequences were removed from the NTU datasets. Subsequently, the following standardized steps were applied:

- Fixed the frame size to 300 frames by padding sequences, increasing feature visibility.
- Extracted first-order information by translating each skeleton to have the spine joint as the coordinate system's origin ([0,0,0]).

- Rotated skeletons around the spine joint, ensuring the person initially faces the positive x-axis and the spine is parallel to the z-axis.

Following pre-processing, the NTU-60/120 datasets were split using the standard cross-subject protocol (Shahroudy et al., 2016), while maintaining class balance. For the CWBG dataset, the train-test split was performed as described in Section 3.4.2. The processed data was subsequently stored as TFRecords to optimize training efficiency.

To match the single-person actions in CWBG, one-person-per-sequence architecture was used during transfer learning instead of the default two-person ST-GCN. For NTU-60/120 and NTU-44-B/W, the individual with the most prominent activity was selected, while ambiguous interactions were excluded from NTU-44 and NTU-22.

The Kinetics-skeleton and KS-KSS datasets were pre-processed differently, using a two-stage approach. Primary pre-processing involved:

- Centralization: positioning skeleton coordinates relative to the center point rather than the default upper left corner.
- Match poses: tracking the correct individual throughout the two-person skeleton sequence based on distance metrics.

Secondary pre-processing introduced randomized spatial-temporal augmentations during training, following protocols outlined in (Yan et al., 2018). Specific augmentation configurations were experimentally optimized and documented in supplementary materials.

4.2. Implementation Details

All experiments, except for in-the-deployment tests, were conducted on a PC with an AMD Ryzen 9 3900X 12-Core (3.79GHz) processor and an NVIDIA GeForce RTX 2060-S GPU with 8GB memory. In-the-deployment experiments were conducted on a PC with an Intel i7-8700 6-Core (3.20GHz) processor and an NVIDIA GeForce GTX 1070 GPU with 8GB memory.

All ST-GNN models were trained using categorical cross-entropy loss. Random seeds were not fixed, resulting in variability due to batch shuffling and weight initialization. For CWBG vanilla experiments, repeated hold-out validation was used to achieve stable performance (Section 5.1).

To study the effects of model capacity, we varied ST-GCN hyper-parameters such as the number of layers, number of channels, and the use of batch normalization. Results are provided in Section 5.2.

For both in-the-lab and in-the-wild CAR implementations, several optimizer hyper-parameters were tuned to improve model accuracy. Key hyper-parameters explored included:

- Optimizer: SGD and Adam
- Learning rate scheduler: piecewise constant and exponential decay, with learning rates between 0.1 and 0.0001
- Weight decay: L2 regularization with values of 0.01, 0.001, and 0.0001
- Weight initialization: truncated and untruncated normal distributions with varying parameters
- Label smoothing: smoothing factors of 0, 0.1, and 0.2 (mainly in TFL experiments)

A full list of hyper-parameters is provided in the supplementary material. Final selections, based on accuracy and stability, are detailed in Section 5.3. Additionally, the effect of age and gender on CWBG performance was analyzed using LOOCV, with results reported in Section 5.4.

Feature extraction (FX) experiments initially compared multi-class SVM, multinomial logistic regression, and FFNN classifiers, with results reported in (Mohottala et al., 2022b). Due to FFNN's superior performance, it was used in all FX implementations, with results presented in Section 5.5. Hybrid-frozen and vanilla fine-tuning approaches, previously selected based on preliminary layer-wise analyses (Mohottala et al., 2022b), were also adopted in this study (Section 5.5).

We reproduced benchmark results using the provided hyper-parameters, except for adjustments to batch size and number

of epochs due to GPU memory limitations (batch size set to 4 in most experiments). Vanilla implementations of four additional ST-GNN models mentioned in Section 3.2.2 were also conducted to study the impact of architectural differences on small datasets and potential accuracy improvements. Results are given in Section 5.6.

In-the-wild action recognition experiments using KS-KSS were conducted with the ST-GCN model, with results presented in Section 5.7. Since occlusions in videos can bottleneck pose estimation and impact recognition accuracy, we compared action recognition accuracy against pose-estimation confidence values; results are given in Section 5.8.

A modular pipeline was developed for in-the-deployment experiments, with the action recognition module trained on KS-KSS and evaluated using a privately collected child dataset. Results are detailed in Section 5.9.

4.3. Evaluation Methods

The hold-out method was primarily used for model evaluation, while LOOCV was applied to the CWBG dataset. Cross-subject evaluation was employed for most CWBG and NTU implementations, whereas random splits were mainly used for Kinetics-skeleton and KS-KSS datasets.

Top-1 accuracy served as the main evaluation metric, complemented by confusion matrices for class-wise performance analysis. Box-and-whisker plots compared model performance and confidence across different learning methods and architectures, while bar plots compared overall accuracies. 3D skeleton visualizations were used to interpret individual sample predictions. Additionally, reliability diagrams and t-SNE visualizations were employed to study model calibration and feature/embedding space separability.

5. Results and Discussion

This section presents a comprehensive analysis of experimental results obtained across multiple protocols, datasets, and model configurations. We evaluate the effectiveness of ST-GNN models under various learning strategies, investigate performance trends across different data scenarios, and highlight key insights drawn from in-the-lab, in-the-wild and in-the-deployment implementations.

5.1. ST-GCN Implementations

ST-GCN architecture was exclusively used in the initial CAR implementations and the results are detailed in this subsection.

Vanilla implementations were done with class-wise protocols under cross-subject evaluation method and the results are given in the Table 3. Due to the small dataset size, stochastic nature of initial weights and minibatch shuffling in the training loop, varying results were observed. Repeated model training and validation were conducted with the fixed train-test subsets resulting in model performance that is invariant to initial conditions.

Due to higher inter-class variation, CWBG-dissimilar performance is comparatively better than CWBG-similar. In contrast, increased number of classes in CWBG-full makes it difficult to discriminate action classes, resulting in a lower accuracy. Increased accuracy in CWBG-shared can be attributed to lower number of classes. With CWBG-Random approach, contrary to our expectations, the results were slightly lower than the cross-subject approach with the exception of CWBG-Similar protocol. While the information leakages resulting from the random split like this generally increase the overall model performance, small dataset size along with changes in inter-class and intra-class variations of test set may be affecting it.

CWBG-LOOCV protcol was implemented for each child independently and the final averaged results are given as LOOCV in Table 3 for all four CWBG subset protocols. Since each LOOCV implementation uses 29 out of 30 children for training the model, it can be inferred that each model developed for every child is roughly comparable to one another. This is due to the fact that 93.33% of the data used in any two random LOOCV implementations are identical. Thus averaging the results from all 30 implementations is justifiable.

A box plot of softmax probabilities under the CWBG-D protocol (Figure 6), considering only correctly classified samples, shows similar Top-1 accuracy across cross-subject, random,

Table 3: Vanilla Implementations Top-1 Accuracy

	CWBG-full	CWBG-dissimilar	CWBG-similar	CWBG-shared
Cross subject	45.82 ± 3.4	66.78 ± 2.6	47.29 ± 2.4	81.78 ± 4.1
Random	44.82 ± 2.6	66.22 ± 3.1	53.10 ± 1.1	80.74 ± 1.4
LOOCV	48.93 ± 10.7	67.53 ± 12.4	55.79 ± 13.8	81.13 ± 11.4

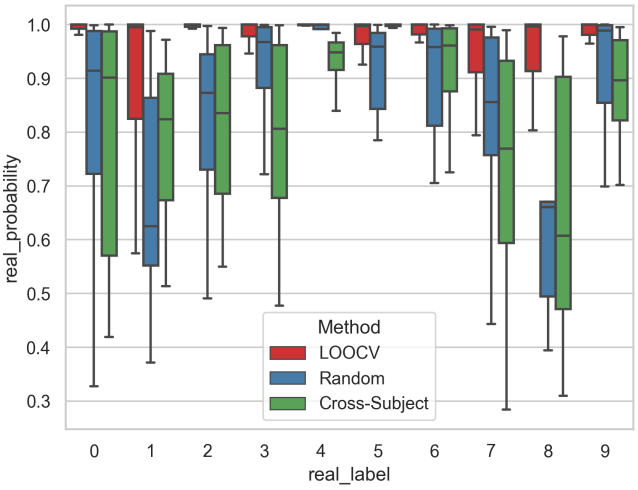


Fig. 6: Probability distribution of vanilla implementations under CWBG-D protocol

and LOOCV approaches, but with higher median confidence for LOOCV, followed by random, and then cross-subject.

5.2. *ST-GCN architectural hyper-parameter tuning*

The original ST-GCN architecture (Yan et al., 2018), utilizes a decoder architecture that features increasing channel numbers and decreasing feature map sizes in a pyramid structure, similar to other popular models like VGG-19 and ResNet. This enables the network to capture progressively more complex and detailed features as it goes deeper into the architecture. The original ST-GCN architecture consists of three blocks, each of which contains 4, 3, and 3 ST-GCN layers, respectively. Implementations on architecture were done by keeping the pyramid structure intact but changing the number of layers, resulting in changes in the network's depth and changing the number of filters in every layer, resulting in changes to the network's width. Four depth-wise configurations (D_d) were used in experiments with (i, j, k) number of layers from each block such that configuration D_{10} is the original ST-GCN architecture with $(4, 3, 3)$, D_7 with $(3, 2, 2)$, D_4 with $(2, 1, 1)$ and D_1 with $(1, 0, 0)$. Under each depth-wise configuration, 8 width-wise configurations were also used as in Table 4. Number of filters were changed in each layer by the same ratio R, $R = \bar{F}_n/F_n$, where n refers to ST-GCN layer number ($1 \leq n \leq d$) while $\bar{F}$ refers to new model filters and F to the original model filters. Implementations were done with CWBG-Full protocol, which contains 968 training samples and the results are given in Table 4. Under D_{10}, D_7 and D_4 depth-wise configurations, results show signs of under-fitting with lower R values, over-fitting with higher R values and good-fit with intermediate R values but under D_1 configuration, over-fitting is not present. Best test accuracy of 53.88% is achieved with D_4 with $R = 1/8$ which results in $\approx 17k$ trainable parameters. As the diameter of the CWBG skeleton graph in Fig. 5 is 10, the experiments were limited to maximum of 10 layers since having a maximum of 10 layers is sufficient for the information to propagate throughout the graph using the graph convolution operation. Lower R values such as $1/32$ result in small number of channels such as 2 in the starting layer resulting in an information bottleneck situation which could also explain the under-fitting scenarios.

These results are also explainable from model capacity since higher R values result in over-parameterized models as evident by the large number of trainable parameters, reaching as high as $\approx 76M$ in Table 4 and lower R values result in under-parameterized models, with the smallest number of trainable parameters being 235. Even though the optimal capacity for a model is achieved when number of trainable parameter is similar to number of training samples, comparable models with $\approx$ 1000 parameters in Fig. 7 doesn't give strong evidence for presence of this phenomenon. Instead, results in Fig. 7 and Table 4 indicate that the best accuracy is generally achieved

Table 4: ST-GCN architecture hyper-parameter tuning

R	D_{10}		D_7		D_4		D_1	
	Acc	Param	Acc	Param	Acc	Param	Acc	Param
5	45.85	76M	45.08	50M	47.67	25M	47.41	932k
2	46.89	12M	45.85	8M	48.70	4M	**51.81**	152k
1	48.96	3M	46.11	2M	49.74	1M	50.77	39k
1/2	43.26	770k	51.03	500k	52.07	251k	51.04	10k
1/4	**53.36**	195k	**53.10**	130k	50.25	64k	43.26	3k
1/8	51.55	50k	49.48	33k	**53.88**	17k	45.08	967
1/16	48.45	13k	48.76	9k	45.34	4.6k	32.90	407
1/32	33.67	3.6k	30.56	2.5k	36.26	1.4k	20.47	235

with over-parameterized models. These results are consistent with the existing research done on over-parameterized neural networks and the resultant high generalization accuracy (Zhang et al., 2021a; Allen-Zhu et al., 2018). Furthermore, it is consistant with the original ST-GCN implementation achieving SOTA results on NTU-60 dataset with $\approx 3M$ parameters despite the training set being $\approx 40k$ samples.

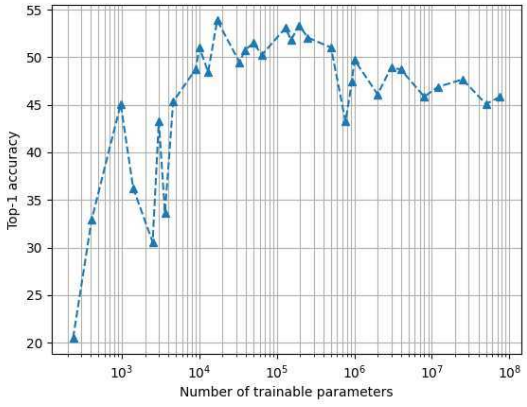


Fig. 7: Model complexity results

5.3. ST-GCN Optimizer hyper-parameter tuning

A comprehensive hyper-parameter tuning process was conducted across various CWBG dataset protocols. In addition to fundamental optimizer parameters such as learning rate and batch size, the experiments also explored the effects of learning rate scheduler configurations, regularization techniques, and weight initialization strategies. Final optimal hyper-parameters obtained across cross-subject protocols are given in Table 6.

Given the small dataset size, Stochastic Gradient Descent (SGD) outperformed Adam. While the piecewise decay scheduler yielded the best results across most protocols, exponential decay was superior for CWBG-Sh. Notably, under the LOOCV protocol, exponential decay consistently delivered the best performance across all implementations. The final hyper-parameters for both LOOCV and random split protocols and detailed descriptions of each hyper-parameter are provided in the supplementary document, and additional details, including model performance, are available via the GitHub repository[2]. Furthermore, performance difference between different hyper-parameter sets for KS-Vanilla protocols are given in the KS protocols based results section in supplementary document.

5.4. Age and Gender Effect

Participants in the CWBG dataset, aged between 3 and 6 years (average 4.4), demonstrate variability in action recognition performance likely influenced by cognitive and motor development stages. Using the LOOCV implementations with the CWBG-D protocol, an analysis was conducted to evaluate the effect of age and gender on accuracy (Table 5). Results indicate a clear trend of increased accuracy with higher age groups,

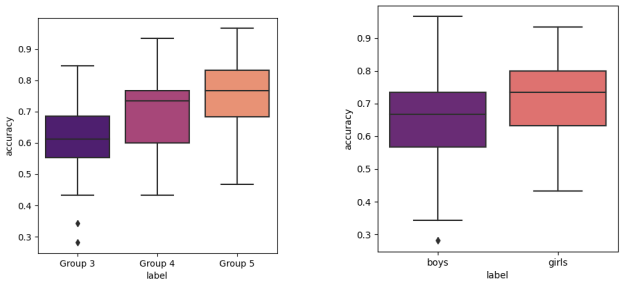
[2] https://github.com/sankamohotttala/ST_GNN_HAR_DEML/tree/main/vanilla_cwbg

Table 5: LOOCV child-wise model performance with CWBG-Dissimilar protocol

Group Details		Boys					Girls					Average
		A	B	C	D	E	A	B	C	D	E	
Age Three (3)	Child ID	26	4	7	16	3	22	21	2	9	24	60.23 ± 12.0
	Accuracy	58.97	62.22	57.78	64.52	37.50	67.71	73.08	46.67	59.14	74.71	
Age Four (4)	Child ID	6	25	20	5	14	10	8	15	12	27	69.10 ± 12.2
	Accuracy	77.78	71.11	79.57	47.78	63.33	58.89	76.67	61.11	84.44	70.00	
Age Five (5)	Child ID	13	17	18	28	1	11	23	29	30	19	75.35 ± 11.3
	Accuracy	62.22	77.42	77.78	56.67	83.91	74.44	81.11	84.44	77.78	77.78	
Average		65.24 ± 13.8					71.20 ± 12.1					

Table 6: Final hyper-parameter values for cross-subject CWBG protocols

Hyper-parameter	CWBG Full	CWBG Dissimilar	CWBG Similar	CWBG Shared
base lr	0.01	0.01	0.01	0.01
batch size	4	4	4	4
epochs	30	30	30	30
Optimizer	SGD	SGD	SGD	SGD
SGD momentum	0.9	0.9	0.9	0.9
SGD nesterov	TRUE	TRUE	TRUE	TRUE
learning rate scheduler	PiecewiseConstDecay	PiecewiseConstDecay	PiecewiseConstDecay	ExponentialDecay
steps	[10, 20]	[10, 20]	[10, 20]	-
iterationNum	924	628	612	312
values	[0.01, 0.001, 0.0001]	[0.01, 0.001, 0.0001]	[0.01, 0.001, 0.0001]	-
stepdecay	-	-	-	TRUE
steps decay	-	-	-	78
decay rate	-	-	-	0.90
Weight decay	l2	l2	l2	l2
Weight decay value	0.0001	0.0001	0.0001	0.0001
Weight initializer (WI)	VarianceScaling	VarianceScaling	VarianceScaling	VarianceScaling
WI scale	2	2	2	2
WI mode	fan out	fan out	fan out	fan out
WI distribution	truncated normal	truncated normal	truncated normal	truncated normal



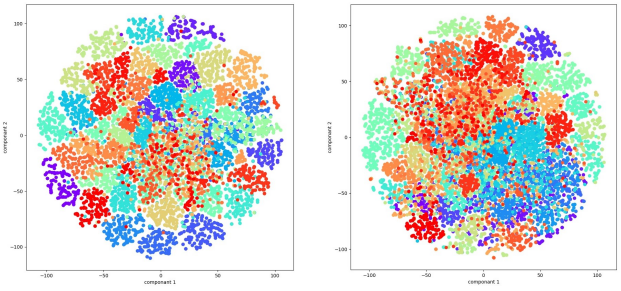
(a) Age-wise results (b) Gender-wise results

Fig. 8: LOOCV results

supported by the age-wise performance distributions depicted in Figure 8.

Additionally, gender-wise analysis revealed consistently higher accuracy in girls compared to boys, suggesting gender-based differences in motor skill development. These findings align with the original CWBG study by Vatavu et al. (Vatavu, 2019), which quantitatively showed a decrease in intra-class variability with increased age. This consistency highlights that age and gender significantly contribute to intra-class variation, influencing the action recognition accuracy of the model.

These findings justify the problem definition (Section 1), emphasizing the importance of developing action recognition models specifically tailored to children, thereby reinforcing the significance and necessity of continued research in this area.



(a) NTU-44-B (b) NTU-44-W

Fig. 9: t-SNE visualization of NTU-44 models

5.5. Transfer Learning Implementations

For the transfer learning (TFL) analysis, only the kinect camera based datasets were used with different protocols. The first stage of TFL experiments was the pre-training of ST-GCN model with different source datasets. Second stage was the use of different TFL methods with target datasets.

Table 7: ST-GCN performance on source datasets

Dataset	Accuracy		Samples	
	Top-1	Top-5	All	Train
NTU-120	65.93	88.26	108998	71103
NTU-60	73.27	92.13	54718	38756
NTU-22	89.59	97.95	19700	12882
NTU-5	90.6	100.00	4490	3086
NTU-44	80.72	95.53	39758	26571
NTU-44-B	85.34	96.39	40180	27141
NTU-44-W	66.43	89.54	40049	23927
NTU-44-FRA	79.02	95.36	39758	26571
NTU-60-FRA	69.12	91.87	54718	38756
NTU-120-FRA	60.52	85.45	108998	71103

5.5.1. Pre-Training with Source Datasets

Initial transfer learning experiments involved training the ST-GCN model on various source datasets, aligning their skeleton structures with the CWBG dataset. Structural modifications led to a noticeable drop in NTU-60 accuracy (73.27%) compared to original ST-GCN implementations (78.19%), highlighting the sensitivity of model performance to skeleton structure changes (Table 7).

Down-sampling source datasets to 10FPS was necessary due to frame rate discrepancies between CWBG (10FPS) and NTU (30FPS). Although this increased the samples-to-features ratio beneficially, it introduced considerable information loss, negatively impacting overall accuracy as evidenced by comparisons between FRA and original datasets (Table 7).

Furthermore, because of the best performing class selection, NTU-44-B contains classes with higher inter-class variation and lower intra-class variation as observed from visualizations based on global average pooling layer output embeddings using t-SNE in Figure 9. This is in contrast to NTU-44-W which results in the opposite explaining the considerable accuracy difference between NTU-44-B and NTU-44-W. Because of the

Table 8: Transfer Learning Results on CWBG dataset (FX: Feature Extraction, FT: Fine-Tuning and HF: Hybrid-Frozen Fine-Tuning)

Source Dataset	CWBG-F			CWBG-D			CWBG-S			CWBG-Sh		
	FX	FT	HF	FX	FT	HF	FX	FT	HF	FX	FT	HF
NTU 120	57.51	58.03	58.55	74.53	76.78	77.53	65.88	66.27	67.45	89.63	91.11	92.59
NTU 60	51.81	52.33	57.25	71.16	69.66	78.28	61.57	63.92	63.92	90.37	92.59	92.59
NTU 22	50.52	52.07	52.07	69.29	67.04	70.79	56.86	58.43	57.65	86.67	85.19	89.63
NTU 5	43.74	42.75	47.41	55.81	58.43	66.67	54.90	55.69	56.47	85.93	86.67	88.89
NTU 44	55.44	56.74	57.51	74.16	75.28	78.28	62.75	65.49	66.27	91.85	91.85	89.63
NTU 44 - B	51.81	53.37	58.81	73.41	75.66	78.28	57.25	59.61	60.39	89.63	90.37	88.89
NTU 44 - W	51.04	50.26	53.63	70.04	71.91	78.65	58.82	63.53	61.18	87.41	90.37	91.85
NTU 44 - FRA	57.77	57.25	56.22	79.10	74.91	77.15	**67.31**	62.35	63.14	94.07	91.11	90.37
NTU 60 - FRA	54.15	55.44	55.44	77.15	76.40	79.03	62.75	66.67	65.49	94.07	91.85	94.07
NTU 120 - FRA	**59.33**	57.51	58.29	**82.24**	76.78	78.28	63.14	60.78	65.49	**94.81**	89.63	91.11

manual class selection, the performance of NTU-44 falls between that of NTU-44-B and NTU-44-W.

5.5.2. *Transfer Learning with Target Datasets*

Transfer learning performance is affected by multiple factors and in this research, effect of several factors were observed using a diverse set of source datasets. With the number of classes selected in each source dataset, dataset diversity and size factors were controlled. Previous research (Li and Hoiem, 2017) suggest higher diversity and larger dataset size increase the transfer learning performance more than any other factor and this can be observed with NTU-{60,120} and NTU-{60,120}-FRA implementations as evident by results in Table 8. Best performance for CWBG-Full, CWBG-Dissimilar and CWBG-Shared were achieved with NTU-120-FRA source dataset and for CWBG-Similar, best performance was achieved with NTU-120 source dataset (Table 8), both datasets equal in size. Furthermore they are x2 larger and x2 diverse than the second largest source datasets, NTU-60 and NTU-60-FRA as detailed in Table 7.

Compared to training from scratch, with feature extraction approach CWBG-Full achieves a 30% increase in Top-1 accuracy, with the accuracy score rising from 45.82% to 59.33%. CWBG-Dissimilar and CWBG-Shared both achieve their best performance also with feature extraction method, accuracy score increasing from 66.78% to 82.24% and from 81.78% to 94.81% respectively, each gaining 23% and 16% increase in accuracy. For CWBG-Similar protocol, best performance is observed with Hybrid-Frozen approach, where accuracy score increased from 47.29% to 67.45%. Similar to what was observed in vanilla implementations, regardless of source dataset used, across all transfer learning approaches (Table 8), CWBG-Dissimilar outperforms CWBG-Similar results.

Curriculum learning inspired class selection approach was evaluated with NTU-44, NTU-44-B and NTU-44-W source datasets. Overall, NTU-44 results in best performance in 8 out of 12 transfer learning implementations while both NTU-44-B and NTU-44-W result in best performance for 2 implementations each. Furthermore, NTU-44-B performs better than NTU-44-W in 7 out of 12 implementations. When feature extraction and vanilla fine-tuning are considered, NTU-44-B performs better than NTU-44-W in 6 out of 8 implementations. Thus, these results corroborate the use of curriculum learning (CL) inspired approach to select classes for the source dataset and the suitability of CL as a data-efficient learning method.

Comparison of different transfer learning approaches under each CWBG protocol is shown in Figure 12. Source datasets were used in x axis with ascending order of dataset size. Compared to the vanilla implementation results, all protocols gain a

performance increase as a result of transfer learning with source datasets with the exception of NTU-5 dataset. Both feature extraction and vanilla fine-tuning result in an accuracy decrease with NTU-5 source dataset under CWBG-F and CWBG-D protocols. This can be considered due to negative transfer learning because of the low-diversity. Increased source dataset size results in increased accuracy across all protocols. However, in the case of NTU-44 and its variations (NTU-44-{B,W,FRA}), where the differences in dataset size are relatively small, other factors become more significant, leading to unpredictable results.

Furthermore, when considering the overall performance under CWBG protocols, hybrid-frozen approach performs better than vanilla fine-tuning and feature extraction approaches with the exception of CWBG-shared protocol. This behaviour is more pronounced when only the non-FRA datasets were considered thus validating the conclusions drawn from CWBG benchmark results in Table 9. When comparing the overall transfer learning results in Table 9, 10FPS down sampled datasets result in the best performance in all CWBG protocols except CWBG-Similar. These findings indicate that the frame rate has a significant impact on feature representation despite the loss of information in down-sampled datasets.

To summarize the improvement in transfer learning performance, we introduce the TFL enhancement factor (TFL_F).

$$f_o(s_p, t_p) = \left\{ f_i(s_p, t_p) \;\middle|\; i \in \{\text{ FT , FX , HF }\} \right\},$$
$$\text{TFL}_F = \frac{\max\left(f_o(s_p, t_p)\right) - f_v(t_p)}{f_v(t_p)}, \tag{17}$$

where s_p and t_p denote the source and target protocols, $f_i(.)$ the accuracy of TFL method i, and $f_v(.)$ the vanilla training accuracy. TFL_F values for each (s_p, t_p) are shown in Figure 10.

As mentioned in section 4.2, label smoothing (LS) was used as a regularization method and this improved the generalization of models across most of the TFL implementations. Neural networks generally become too confident about their predictions (Müller et al., 2019) and LS mitigates this by adding a small noise to ground-truth labels during training. To estimate model confidence, we utilized reliability diagrams (Figure 11)

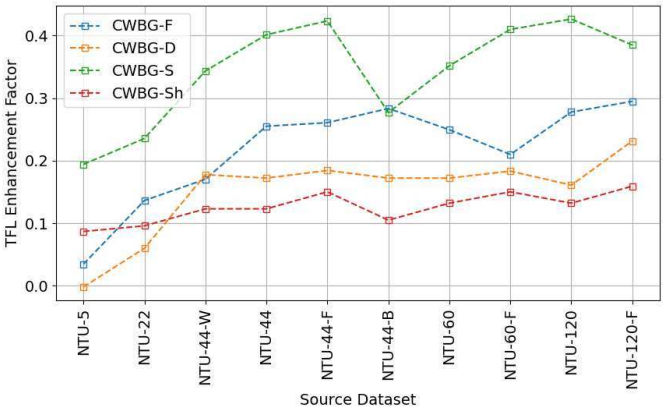


Fig. 10: Accuracy Enhancement Factor of TFL across NTU Datasets

and as illustrated, when $LS = 0.1$, resultant model is a well-calibrated model. Similar behavior can be observed in other TFL approaches as well thus we use $LS = 0.1$ as the optimal smoothing factor in most implementation.

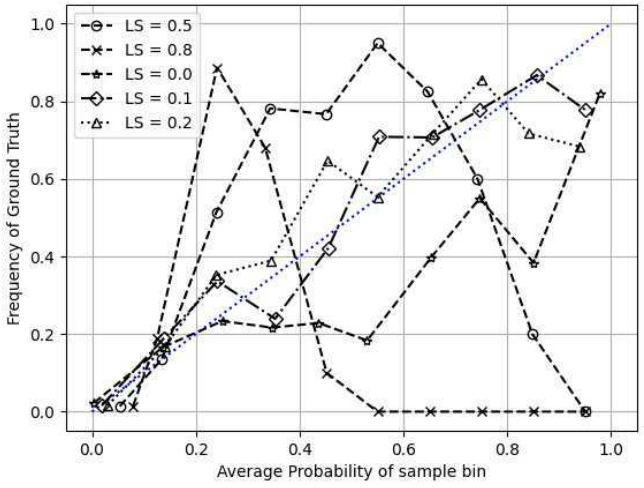


Fig. 11: Label smoothing effect on feature extraction

Despite the similarity of the classes, CWBG-Sh test accuracy on NTU-5 based Model is only 56.29% (compared to NTU-5 test acc of 90.6% (Table 7)). This highlights the significance of our research problem and the importance of children-focused HAR models (i.e., CAR models).

For CWBG-Sh protocol, best performance under feature extraction approach resulted in 94.81% while vanilla fine-tuning and hybrid-frozen approaches resulted in 92.59% and 94.07% respectively. Considering the class-wise accuracy, lowest accuracy resulting in 'Throw ball' class can be attributed to the ambiguity of the class while less ambiguous classes like 'hands

Table 9: Benchmark results of CWBG protocols

Accuracy	CWBG-F	CWBG-D	CWBG-S	CWBG-Sh
All source datasets	59.33	82.24	67.45	94.81
non-FRA datasets (30FPS)	58.81	78.65	67.45	92.59
FRA datasets (10FPS)	59.33	82.24	67.31	94.81

up', 'crouch' and 'clapping' resulted in higher accuracy.

A comparison of child action recognition and adult action recognition can be done between NTU-5 and CWBG-Sh due to the similarity of classes. Top-1 accuracy of 81.78% resulted in CWBG-Sh dataset based ST-GCN model while 90.6% was resulted in NTU-5 dataset based ST-GCN model with skeleton changes as detailed in Section 3.4.3. When original Kinect v2 skeleton structure is used in NTU-5 dataset, accuracy increased to 93.80%. This is to be expected since NTU-5 contains 3086 samples in train set while CWBG-Sh contains only 312 samples in train set, hence a $\times$10 larger dataset usage in adult action recognition implementation. But with transfer learning approaches, contrary to the expectations, CWBG-Sh accuracy increased to 94.81%, exceeding original Kinect v2 NTU-5 accuracy as well as modified Kinect v1 NTU-5 accuracy.

These results conclude that 1) ST-GCN model can be effectively used for child action recognition in the same way it can be used for any standard human action recognition task, 2) despite a $\times$10 smaller dataset, ST-GCN model can be effectively used for transfer learning and 3) contrary to the past literature, transfer learning can be effectively used to improve child action recognition.

5.6. ST-GNN Model Implementations

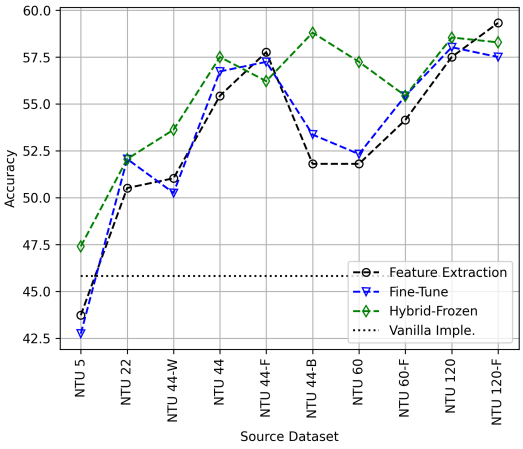
Initial evaluations indicated that limitations in ST-GCN architecture affected the accuracy on the CWBG datasets. To address this, four state-of-the-art ST-GNN models (2s-AGCN, MS-AAGCN, ST-GAT, and RA-GCN) with varying architectural features were explored in Section 3.2.2. Performance on the NTU-60 benchmark dataset are detailed in Table 11.

Vanilla implementations were done with ST-GNN models and the results are given in Table 12 for all CWBG protocol under cross subject evaluation approach. Since ST-GNN models leverage higher order information in datasets in a multi-model/stream approach with an ensemble, experiments were also done on those different streams as well and the results are given in Table 12. Under each CWBG protocol, all experimental results except one show substantial performance gains over ST-GCN thus supporting the hypothesis that the architectural features can be used as data-efficient learning methods.

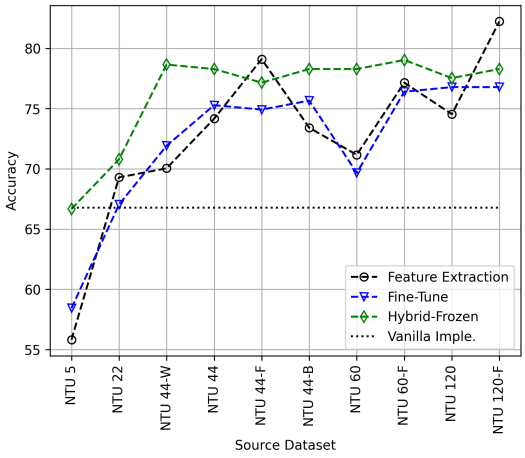
As the results in Table 12 indicate, bone modality perform better than joint modality in 2s-AGCN while in MS-AAGCN, motion modalities are better over static modalities. But the results obtained on NTU benchmark dataset do not agree with this. Thus these findings indicate that performance of modality depends on the dataset and can not be generalized.

RA-GCN results in Table 12 indicate that on average, 3-stream RA-GCN (3s RA-GCN) perform well over other RA-GCN models. We compare the performance of 3s RA-GCN model and ensemble models of other ST-GNNs in Figure 13. When comparing the vanilla ST-GNN models, except for CWBG-Sh protocol, highest accuracy is achieved with MS-AAGCN model. When considering the models with minimal differences in top-1 accuracy, such as MS-AAGCN and ST-GAT, performance with CWBG is unpredictable. These observations indicate that the best ST-GNN model to use with a given dataset, especially when the dataset is small, is not always the best SOTA ST-GNN model and that it depends on the dataset and the task.

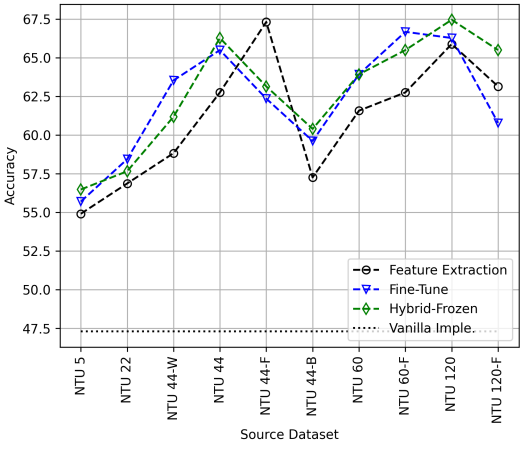
Detailed evaluations of the performance of the ST-GNN models were not possible due to the use of different modalities across ST-GNN models. However, a comparison between the MS-AAGCN and ST-GAT models, which employ the same modalities, was conducted. Figure 14 contains the confusion



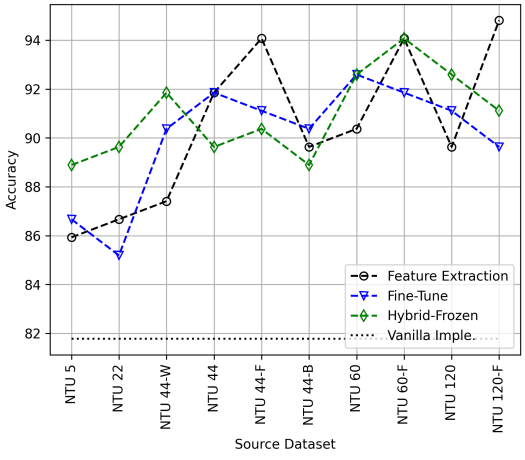
(a) CWBG - full protocol



(b) CWBG - dissimilar protocol



(c) CWBG - similar protocol



(d) CWBG - shared protocol

Fig. 12: Transfer Learning results on CWBG dataset

Table 10: ST-GNN benchmark results

Accuracy	CWBG-F	CWBG-D	CWBG-S	CWBG-Sh
TFL-ST-GCN$_{max}$	59.33	82.24	67.45	94.81
Joint-ST-GNN$_{max}$	59.59	82.02	67.45	96.29
ST-GNN$_{max}$	66.58	87.27	70.20	96.29

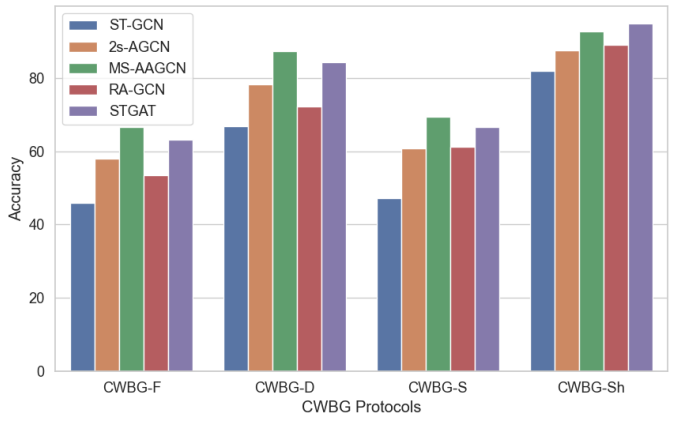
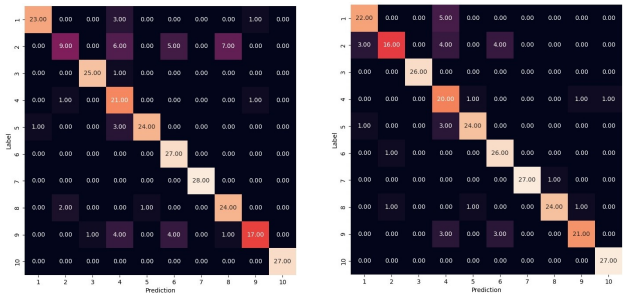


Fig. 13: ST-GNN model performance



(a) ST-GAT (b) MS-AAGCN

Fig. 14: CWBG-D classwise comparison

matrices resulted from the ST-GAT ensemble model and MS-AAGCN ensemble model with CWBG-D dataset. Higher top-1 accuracy resulted from MS-AAGCN (87.27%) compared to ST-GAT (84.26%) can be attributed to the its higher classwise accuracy for 'climb ladder' (ID: 2) and 'throw ball' (ID: 9) classes (Figure 14) since both models result in approximately similar accuracy for all other classes.

A modality independent comparison of all ST-GNN models with the exception of RA-GCN was done by only considering the joint modality and the results are given in Figure 15. Higher accuracy of 2s-AGCN compared to ST-GCN (Figure 15) can be attributed to the use of an adaptive graph in the spatial graph convolution in the model architecture instead of only using the static global graph based on skeleton structure. Compared to 2s-AGCN, MS-AAGCN also result in higher accuracy across all CWBG protocols and this can be attributed to the improved adaptive graph as well as the use of spatial, temporal and channel attention modules after the graph convolution block. Superior performance of joint modality based ST-GAT over ST-GCN show that graph attention network outperforms graph convolution networks and the equivalent performance of ST-GAT and MS-AAGCN models validate the effectiveness of attention mechanism and show that attention incorporated GNN layers performs better than attention usage after the GCN layers.

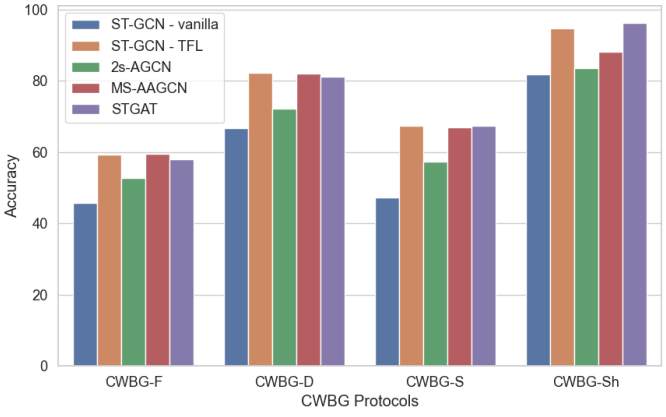


Fig. 15: ST-GNN Joint modality performance

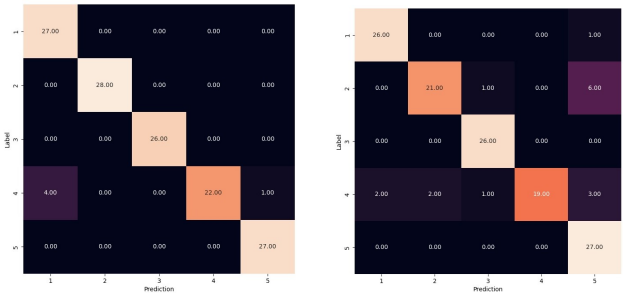
A classwise accuracy comparison of these two models were done with CWBG-Sh protocol (Figure 16) because of the considerable difference in top-1 accuracy. With ST-GAT, misclassification only happens with 'throw ball' class and four out of five misclassified samples were labeled as 'hands up' and three of them come from a single participant (ID: 30). Visualization of sequences show the participant throwing the ball over the head thus resembling the hands-up action movement. Figure 17 illustrate this with several intermediate frames visualized as 3D skeletons. With MS-AAGCN, misclassification is much more varied but it is more pronounced with 'jump' and 'throw ball' classes. Contrary to the joint modality, the ensemble MS-AAGCN model result in similar confusion matrix to ST-GAT in figure 16.

Furthermore, top-1 accuracy of best transfer learning based ST-GCN model with each protocol as well as best joint modality based ST-GNN are given in table 10. Comparison between best transfer learning results of ST-GCN and the joint-modality ST-GNN results shows a similarity in performance. But when considering all the modalities and the ensemble models, ST-GNN models such as ST-GAT and MS-AAGCN outperforms

the transfer learning approaches.

These results conclude that,

1. While best state-of-the-art (SOTA) models tend to outperform other SOTA models when used on relatively large datasets, they does not always perform well with small datasets. Thus, SOTA models cannot be generalized as data-efficient methods,
2. As results indicated, transfer learning (TFL) methods perform on par with architecturally improved vanilla models thus both are potential directions as data-efficient methods,
3. Higher-order information based (i.e., Multi-Modality) ensemble models is a practically useful data-efficient method.



(a) ST-GAT (b) MS-AAGCN

Fig. 16: CWBG-Sh joint modality classwise comparison

5.7. *In-the-wild data based implementations*

Initial vanilla implementations using the KS and KSS protocols demonstrated improved accuracy when transitioning from the KSS to the KS protocol, as shown under Vanilla method in Table 13. This improvement likely resulted from the increased number of samples per class in KS dataset. Best overall results from different transfer learning methods for OpenPose based skeleton modality is shown in Table 13 under TFL_{best}. Detailed results for TFL can be found in the supplementary document. Furthermore, RGB based implementations for KS-KSS can be found under $\text{TFL}_{\text{Pre-train}}$ where the pre-training of LRCN model is done using ImageNet dataset.

TFL of skeleton modality was done using kinetics-skeleton dataset as the source dataset and the considerable accuracy increase from vanilla implementation to TFL proves the effectiveness of TFL despite the noisiness of the kinetics-skeleton dataset as evident in the from the confidence value based experiments in section 5.8. This improvement can be attributed to the high diversity and large dataset size of kinetics-skeleton. When comparing the TFL based skeleton modality model with the pre-trained RGB modality based model, both show on par performance validating that skeleton modality based models can be effectively used with RGB datasets when combined with a suitable pose-estimation model. Furthermore, to the best of our knowledge, this is the first time the kinetics-skeleton has been successfully used in a transfer learning approach. Detailed results are provided in the supplementary document. Additional results, including those obtained with the final hyper-parameters as well as selected experiments with alternative hyper-parameter settings, are available in the GitHub repository[3].

These results highlight that,

1. large and diverse datasets such as kinetics-skeleton can be used in transfer learning despite the in-the-wild nature of the dataset (i.e., noisiness),
2. TFL methods can be successfully used with RGB based in-the-wild data with appropriate pose-estimation models thus showing the practicality of the TFL based HAR models.

5.8. *Accuracy vs Confidence comparison*

Initial analysis was done using the classwise sets ($\mathcal{S}^{k_y}$) of kinetic-skeleton dataset as described in Section 3.5.1 and the results are plotted in Figure 18 where independent variable (x axis) represents the classwise average confidence of the most active person ($\bar{c}_0^y$) and the dependent variable (y axis) represents the average accuracy.

Visualized distribution of all 400 classes in Figure 18 implies there is no strong correlation but if the average confidence values is close to zero, then there is a higher chance of resulting in a low accuracy. Quantitative analysis resulted in 0.533 for Pear-

[3] https://github.com/sankamohotttala/ST_GNN_HAR_DEML/tree/main/kss_ks

Table 11: ST-GNN implementations with NTU-60

	ST-GCN	2s-AGCN	MS-AAGCN	RA-GCN	ST-GAT
Original	81.5% (Yan et al., 2018)	88.5% (Shi et al., 2019)	90.0% (Shi et al., 2020)	87.3% (Song et al., 2021)	92.8% (Hu et al., 2022)
Reproduced	78.2%	86.1%	88.7%	84.4%	90.4%

Table 12: ST-GNN Vanilla Implementations Results

ST-GNN		CWBG-F	CWBG-D	CWBG-S	CWBG-Sh
Model	Modality				
ST-GCN	Joint	45.82	66.78	47.29	81.78
2s-AGCN	Joint	52.85	72.28	57.25	83.70
	Bone	56.48	72.66	59.22	84.44
	Ensemble	58.03	78.28	60.78	87.41
MS-AAGCN	Joint	59.59	82.02	67.06	88.15
	Bone	63.73	81.65	63.14	87.41
	J-Motion	62.18	79.03	**70.20**	93.33
	B-Motion	61.66	82.77	66.26	90.37
	Ensemble	**66.58**	**87.27**	69.41	92.59
RA-GCN	2 streams	48.70	70.41	61.96	78.52
	3 streams	53.37	72.28	61.18	88.89
	4 streams	49.48	75.66	57.25	91.11
ST-GAT	Joint	58.03	81.27	67.45	**96.29**
	Bone	61.39	75.28	62.35	89.62
	J-Motion	60.36	80.90	64.70	93.33
	B-Motion	54.40	79.03	60.78	86.67
	Ensemble	63.22	84.26	66.67	94.81

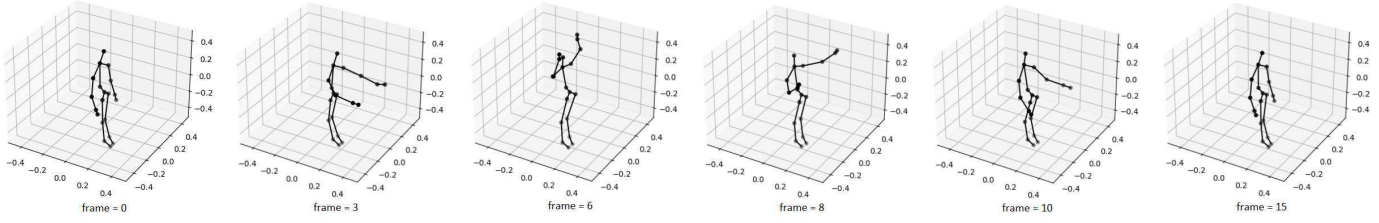


Fig. 17: Visualization of a misclassified sample in CWBG-Sh protocol

Table 13: Implementation results for KS-KSS datasets

Method	Modality	Dataset	Full	Balance	Large	Small
Vanilla	Skeleton	KS	75.29	77.83	69.32	69.23
		KSS	60.68	64.88	59.43	86.95
TFL_{best}	Skeleton	KS	84.3	83.38	86.03	76.92
		KSS	81.26	89.85	87.92	86.95
$TFL_{Pre\text{-}train}$	RGB	KS	86.62	88.64	87.02	73.07
		KSS	82.57	86.23	79.72	78.26

son correlation indicating only a moderate positive relationship and 0.564 for Spearman's correlation. Analysis on other implementation also resulted in similar results and conclusions.

For the KS-KSS dataset, classwise accuracy and the average confidence of two most active persons $(\bar{c}_0^y, \bar{c}_1^y)$ are given in Table 14. 'Position' refers to the place each class take when all 400 classes are ordered in descending order in terms of classwise accuracy.

Higher accuracy and position attained by classes with indices 7, 6, 0, and 2 (Table 14) can be explained as a result of motion-oriented nature of those actions. When considering the all 400 classes in Figure 18, all these four classes are above average. Relatively bad performance of other 4 classes is difficult to attribute to a single cause. Considering 4 and 5 classes, it may be due to truncation/occlusion present in the videos as evidenced by the confidence values but same reasoning is not true for the low performance of 1 and 3 classes.

Since classwise approach doesn't capture the variance within classes, we introduce an approach that is analogous to reliability diagrams as detailed in Section 3.5.1. Since most of the kinetics-400 classes are single person activities, we only considered the most active person in each data sequence and the resultant plot is given in Figure 19. Furthermore, result in Figure 19 also suggest a linear relationship between confidence value and accuracy thus giving quantitative evidence for model performance dependency on pose estimation process and the presence of occlusion and truncation.

Softmax probability based results for the most active person in Figure 20 show that higher confidence values result in higher probability values thus demonstrating a relationship between

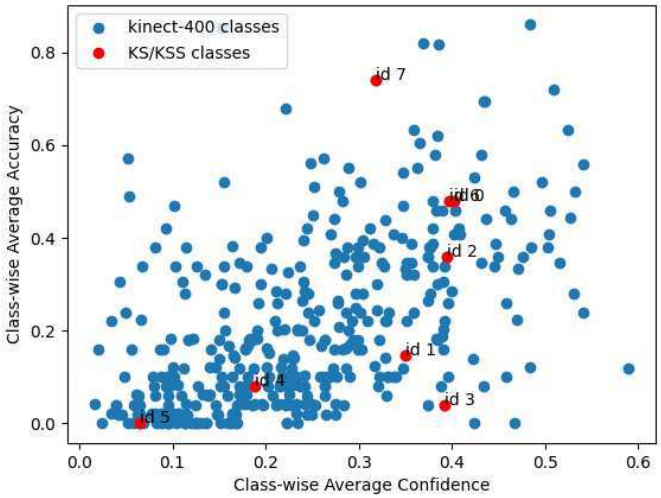


Fig. 18: Classwise accuracy vs confidence comparison

feature confidence values and predicted label confidence values. Moreover, these results corroborate the conclusion we derived from normalize accuracy based implementations.

These results conclude that,

1. both pose-estimation accuracy and the motion-oriented nature contribute to the model's overall accuracy,
2. confidence value from pose-estimation show a linear correlation with ST-GCN model accuracy thus highlighting the importance of utilizing improved pose-estimation models when used in real-world scenarios.

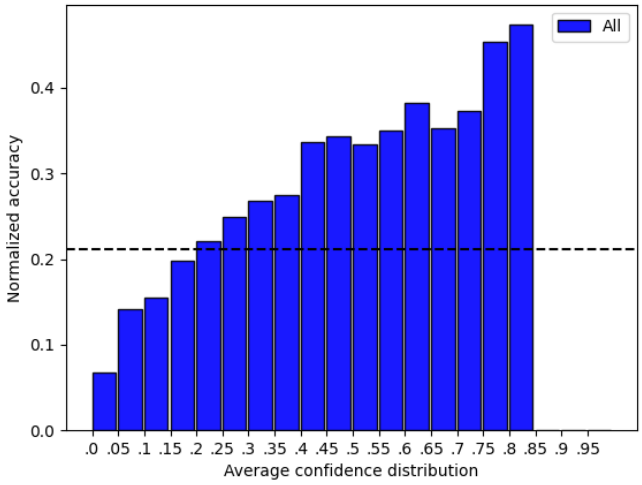


Fig. 19: Normalized accuracy based comparison for most active person ($p = 0$)

Table 14: Accuracy and Confidence Comparison for KS-KSS dataset

Class Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Accuracy	48%	15%	36%	4%	8%	0%	48%	74%
Position	32(2)	210(5)	79(4)	321(7)	276(6)	373(8)	34(3)	5(1)
Confidence ($\bar{c}_0^y, \bar{c}_1^y$)	.40/.12	.35/.18	.39/.11	.39/.16	.19/.03	.06/.01	.40/.12	.32/.05

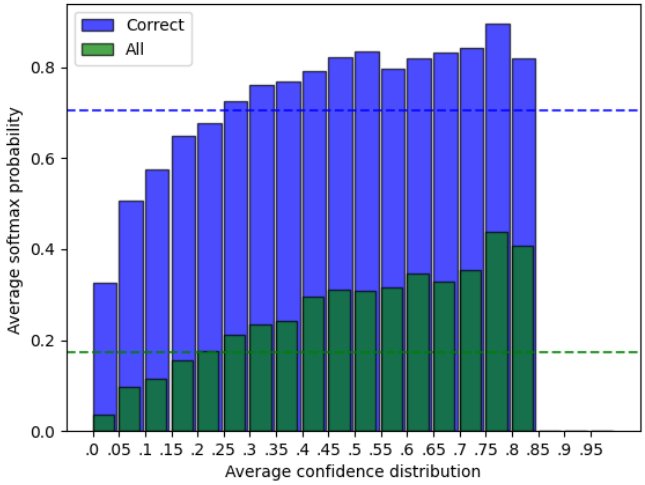


Fig. 20: Softmax probability based comparison for most active person ($p = 0$)

5.9. *In-The-Deployment Implementation Results*

Based on preliminary results, a sliding window of size $n = 30$ frames and a stride of $d = 1$ frame was used for training the CAR model on the KS-KSS dataset for clapping task recognition, as detailed in Section 3.6. The trained ST-GCN model achieved a test accuracy of 91.25% on the KS clapping vs non-clapping subset. The corresponding accuracy and loss plots, along with the confusion matrix, are provided in the supplementary document.

The Action Recognition System (ARS) was employed for inference on the Hummingbird-AS dataset to simulate an in-the-deployment scenario, where the streaming input comprised sequential video frames. Clapping-annotated video data was used to evaluate the system's performance and estimate accuracy. For a representative input video (referred to as "Chenidu_GP02"), the instantaneous prediction probabilities are illustrated in Figure 21. Instantaneous predictions were computed for each frame based on a sliding window containing 14 past frames and 15 future frames, resulting in predictions beginning from frame #15. To enhance robustness during action determination, the ARS utilized a sigmoid probability threshold of 0.5 rather than relying on softmax outputs. Specifically, if the predicted probabilities for both classes were below 0.5, the system classified the action as non-clapping. If not, then the class with higher probability was selected as prediction. This thresholding approach helps mitigate uncertain predictions in streaming scenarios.

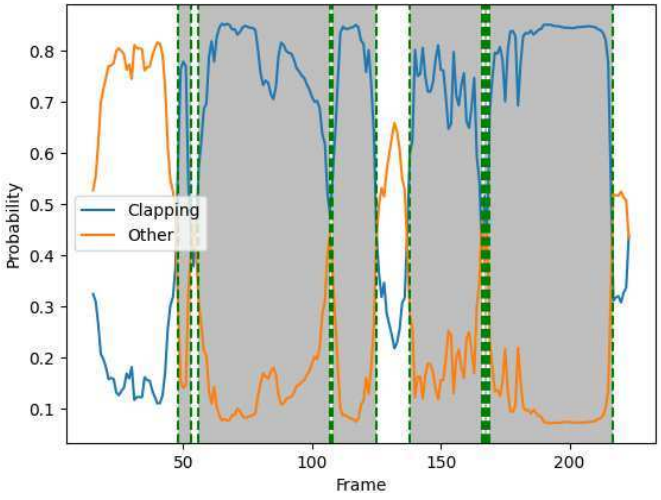


Fig. 21: Probability distribution for the streaming data

The shaded regions in Figure 21 represent the periods during which the system predicted clapping. To estimate the overall accuracy, the cumulative duration of these predicted clapping periods was divided by the total video length. For the "Chenidu_GP02" sample, this streaming-based evaluation resulted in an estimated accuracy of 74.64%. Considering that the dataset is out-of-distribution and no specific optimizations were applied, this level of performance is deemed acceptable.

The current ARS, when deployed in the in-the-deployment setup, outputs the instantaneous probability distribution, instantaneous latency distribution, and on-frame detection and esti-

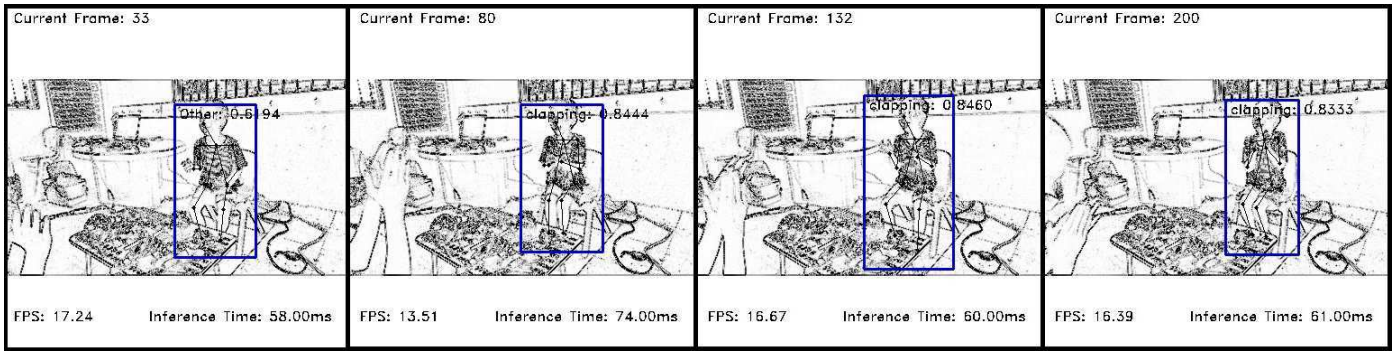


Fig. 22: On-frame outputs with in-the-deployment setup

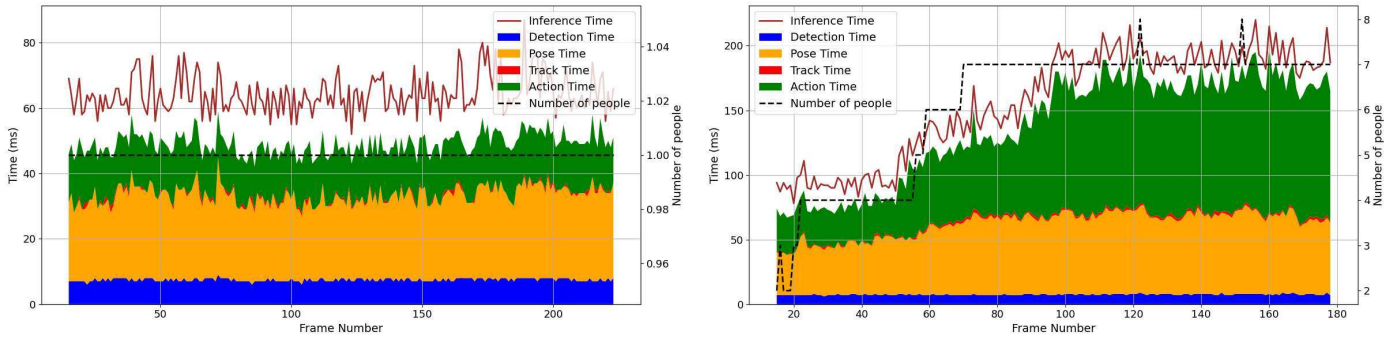
mation results. Sample on-frame outputs for selected time instances are shown in Figure 22. To address ethical considerations, the output videos were anonymized using a Laplacian filter, which acts as an edge detector. Figure 22 presents four such frames, corresponding to frame numbers #33, #80, #132, and #200 from the "Chenidu_GP02" sample.

As shown, latency and related results are also displayed on the frames. The child is highlighted with a blue bounding box, while the skeleton is overlaid in black. If additional individuals are present, they are similarly annotated with black bounding boxes and corresponding skeletons. Additionally, the predicted probability for the selected action class is displayed on each frame.

The distributions of latency across different modules in the ARS, as well as their variation over time for two sample recordings from Hummingbird-AS, are presented in Figure 23. For the "Chenidu_GP02" sample (Figure 23a), pose estimation accounts for the largest share of latency, followed by action recognition, with only a single individual present in the recording. In contrast, for the "Buwaneka" sample (Figure 23b), the number of people varies from 2 to 7, resulting in increased latency for both pose estimation and action recognition. The rate-limiting nature of the action recognition module becomes more pronounced as the number of people increases. Meanwhile, the latencies of the human detection and person tracking modules remain relatively constant, with the efficiency of confidence matching leading to negligible tracking time in both scenarios. In addition to these four ARS modules, other computational tasks also contribute significantly to the overall inference time.

Latency analysis was conducted across 18 annotated child action samples, with the results summarized in Table 15. These findings further corroborate the observation regarding the rate-limiting behavior of the action recognition module in overall latency. When analyzing the latency patterns with respect to the number of people in the frame, the inference time was found to increase approximately linearly due to the higher number of pose estimations and skeleton sequence inputs fed into the ST-GCN model. As the number of people increases, both the pose estimation time and the action recognition time proportionally increase, whereas the human detection and tracking times remain largely unaffected. Although pose estimation takes longer than action recognition when only one or two individuals are present, action recognition becomes the dominant contributor to latency as the number of people grows. Therefore, for achieving real-time performance in this ARS setup, it is essential to address the rate-limiting behavior of the action recognition module.

The overall inference time is approximately 87 ms, corresponding to an average frame rate of around 11 FPS. Real-time performance with a 30 FPS streaming input can still be achieved by adjusting the stride to $d = 3$, which provides a 100 ms window for inference per frame. This adjustment is particularly effective when there are only 1–3 people present in the video frames. As the number of people increases, real-time performance can still be maintained by further increasing the stride. Additionally, further optimization of the action recognition module could contribute to reducing latency. For comparison, our initial in-the-lab CWBG experiments achieved an



(a) For sample video: "Chenidu_GP02"

(b) For sample video: "Buwaneka"

Fig. 23: Inference time distributions for two recordings from Hummingbird-AS

inference latency of approximately 8 ms when only one person was present, more than two times faster than the current setup. This difference in the ST-GCN model's inference speed can be attributed to difference in the training environments, as detailed in Section 4.2. Furthermore, since these performance results (i.e., latency) are independent of the specific action types, they can be generalized to broader human action recognition tasks in in-the-deployment setups.

These results lead to the following conclusions:

1. Streaming input data and out-of-distribution data can be effectively utilized by ST-GCN-based action recognition systems (ARS) for child action recognition (CAR) tasks, achieving both good accuracy and low latency suitable for real-world deployment,
2. As the number of people in the sliding window increases, real-world human action recognition (HAR) systems must optimize the inference time of the action recognition module to maintain real-time frame rates.

6. Conclusion

This paper presented state-of-the-art ST-GNN based implementations for child action recognition (CAR). Transfer learning experiments with ST-GCN models demonstrate that, contrary to past research, adult-based datasets can serve as effective source datasets to improve performance on child datasets. Results further suggest that source dataset diversity and size have a greater impact than either source-target similarity or the source model's Top-1 accuracy.

Moreover, transfer learning with ST-GCN achieves comparable performance to other joint-modality ST-GNN models. Comparisons between CWBG-Sh and NTU-5 protocols reveal that (1) transfer learning-based CAR models can outperform adult action recognition models with similar classes, and (2) transfer learning can surpass vanilla training on datasets 10 times larger. These findings validate transfer learning as a data-efficient strategy for low-resource settings, across both in-the-lab and in-the-wild datasets.

Our analysis highlights that state-of-the-art performance on large benchmark datasets does not always generalize to smaller datasets, as seen with ST-GNN implementations. While learnable graph rewiring enhances CAR accuracy, higher graph expressivity (e.g., moving from GCN to GAT) does not necessarily yield better performance, particularly as the number of classes increases. RA-GCN results further indicate that lower CAR accuracy is not solely due to dataset noise such as occlusion or truncation. Additionally, the superior performance of multi-stream models, observed in general HAR tasks, is evident with ST-GNN models as well. These results emphasize the need for CAR-specific architectural innovations focused on data efficiency, rather than relying on benchmark performance indicators.

Age-related correlations with CAR performance suggest that motor skill development significantly affects action recognition, indicating that increasing dataset size or improving data efficiency alone may not fully address the challenges in CAR. Further research employing deep learning explainability tools

Time Metric (ms)	All	1	2	3	4	5	6	7	8
Inference time	86.96	62.37	83.83	103.99	118.72	122.35	132.98	175.92	191.01
Detection time	7.16	7.10	7.20	7.27	7.13	7.04	6.94	7.20	8.00
Pose time	33.14	25.65	32.64	38.35	40.93	44.80	49.14	59.45	66.51
Track time	0.65	0.42	0.67	0.79	0.88	1.02	1.05	1.46	1.00
Action time	31.84	15.32	29.77	42.86	54.30	55.06	61.22	91.43	96.01

Table 15: Average inference time (ms) distribution with respect to number of people in the sliding window

could provide deeper insights into these effects.

Normalized accuracy and confidence analyses show that, for in-the-wild activities, limitations in pose estimation significantly bottleneck CAR model performance. Thus, improving pose estimation is critical for child action recognition in low-resource settings. The comparable performance between RGB- and skeleton-based models suggests that ST-GNN approaches can be highly effective when supported by better pose estimation.

In-the-deployment evaluations show that ST-GCN-based Action Recognition Systems (ARS) can handle streaming and out-of-distribution child action data with good accuracy and latency. Inference time increases linearly with more people, with action recognition becoming the rate-limiting stage. Real-time performance can be maintained by optimizing the action module and adjusting the sliding window stride. These results highlight the need for system-level adaptations for skeleton-based action recognition in real-world deployments.

These findings open several directions for future work. First, extending the experimental studies to systematically assess how data quality, quantity, diversity, and source-target similarity impact model performance across broader human action recognition (HAR) tasks would be valuable. Second, investigating graph rewiring and expressivity as data-efficient strategies for small datasets could further improve performance. Third, a detailed analysis of the effects of occlusion, truncation, and pose jitter, using multiple pose estimation models (e.g., AlphaPose, Detectron2), could enhance system robustness. Exploring advanced learning methods, such as self-supervised and curriculum learning, and replacing heuristic hyperparameter tuning with evolutionary approaches like genetic algorithms, offer promising directions for improving generalization. Finally, achieving real-time 30 FPS deployment will require targeted optimizations to rate-limiting components, particularly ST-GNN architectures and pose estimation pipelines.

CRediT author statement

Sanka Mohottala: Conceptualization, Methodology, Formal Analysis, Investigations, Visualizations, Writing - Original Draft, Review & Editing, Revision - Experiments, Revision - Writing. **Asiri Gawesha**: Visualizations, Writing - Review & Editing, Revision - Writing. **Pradeepa Samarashinghe**: Project administration, Funding acquisition, Writing - Review & Editing. **Dharshana Kasthurirthna**: Supervision, Writing - Review & Editing, Revision - Writing. **Charith Abhayaratne**: Supervision, Writing - Review & Editing, Revision - Writing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Pradeepa Samarashinghe reports financial support was provided by accelerating Higher Education Expansion and Development (AHEAD) Operation of the Ministry of Higher Education of Sri Lanka funded by the World Bank. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The datasets analyzed in this study consist of publicly available resources and one private dataset, detailed as follows:

- NTU RGB+D (Shahroudy et al., 2016) and NTU RGB+D 120 (Liu et al., 2019) in https://rose1.ntu.edu.sg/dataset/actionRecognition/,
- CWBG (Mohottala et al., 2022b) dataset in http://www.eed.usv.ro/~vatavu/projects/DissimilarityConsensus/,
- kinetics-skeleton (Yan et al., 2018) in https://github.com/yysijie/st-gcn,
- KS-KSS (Mohottala et al., 2022a) in https://github.com/sankamohotttala/KS-KSS-Dataset,
- Hummingbird-AS: The authors do not have the permission to share.

Acknowledgment

This research was supported by the Accelerating Higher Education Expansion and Development (AHEAD) Operation of the Ministry of Higher Education of Sri Lanka funded by the World Bank (https://ahead.lk/result-area-3/).

References

Aflalo, A., Bagon, S., Kashti, T., Eldar, Y., 2023. DeepCut: Unsupervised Segmentation using Graph Neural Networks Clustering. URL: http://arxiv.org/abs/2212.05853, doi:10.48550/arXiv.2212.05853. arXiv:2212.05853 [cs].

Allen-Zhu, Z., Li, Y., Liang, Y., 2018. Learning and generalization in overparameterized neural networks, going beyond two layers. CoRR abs/1811.04918. URL: http://arxiv.org/abs/1811.04918, arXiv:1811.04918.

Aloba, A., 2019. Tailoring Motion Recognition Systems to Children's Motions, in: 2019 International Conference on Multimodal Interaction, Association for Computing Machinery, New York, NY, USA. pp. 457–462. URL: https://doi.org/10.1145/3340555.3356092, doi:10.1145/3340555.3356092.

Aloba, A., Anthony, L., 2021. Characterizing Children's Motion Qualities: Implications for the Design of Motion Applications for Children, in: Proceedings of the 2021 International Conference on Multimodal Interaction, Association for Computing Machinery, New York, NY, USA. pp. 229–238. URL: https://dl.acm.org/doi/10.1145/3462244.3479941, doi:10.1145/3462244.3479941.

Aloba, A., Luc, A., Woodward, J., Dong, Y., Zhang, R., Jain, E., Anthony, L., 2019. Quantifying Differences Between Child and Adult Motion Based on Gait Features, in: Antona, M., Stephanidis, C. (Eds.), Universal Access in Human-Computer Interaction. Multimodality and Assistive Environments, Springer International Publishing, Cham. pp. 385–402. doi:10.1007/978-3-030-23563-5_31.

Aloba, A., Woodward, J., Anthony, L., 2020. FilterJoint: Toward an Understanding of Whole-Body Gesture Articulation, in: Proceedings of the 2020 International Conference on Multimodal Interaction, ACM, Virtual Event Netherlands. pp. 213–221. URL: https://dl.acm.org/doi/10.1145/3382507.3418822, doi:10.1145/3382507.3418822.

Amemiya, Y., Suzuki, S., Sato, M., 2020. Enhancement of child gross-motor action recognition by motional time-series images conversion, in: 2020 IEEE/SICE International Symposium on System Integration (SII), IEEE. pp. 225–230.

Attali, H., Buscaldi, D., Pernelle, N., 2024. Rewiring Techniques to Mitigate Oversquashing and Oversmoothing in GNNs: A Survey. URL: http://arxiv.org/abs/2411.17429, doi:10.48550/arXiv.2411.17429. arXiv:2411.17429 [cs].

Awais, M., Long, X., Yin, B., Chen, C., Akbarzadeh, S., Abbasi, S.F., Irfan, M., Lu, C., Wang, X., Wang, L., Chen, W., 2020. Can pre-trained convolutional neural networks be directly used as a feature extractor for video-based neonatal sleep and wake classification? BMC Research Notes 13, 507. URL: https://doi.org/10.1186/s13104-020-05343-4, doi:10.1186/s13104-020-05343-4.

Batatia, I., Batzner, S., Kovács, D.P., Musaelian, A., Simm, G.N.C., Drautz, R., Ortner, C., Kozinsky, B., Csányi, G., 2025. The design space of E(3)-equivariant atom-centred interatomic potentials. Nature Machine Intelligence 7, 56–67. URL: https://www.nature.com/articles/s42256-024-00956-x, doi:10.1038/s42256-024-00956-x. publisher: Nature Publishing Group.

Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., Grundmann, M., 2020. Blazepose: On-device real-time body pose tracking. arXiv preprint arXiv:2006.10204 .

Bobick, A., Davis, J., 2001. The recognition of human movement using temporal templates. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, 257–267. URL: https://ieeexplore.ieee.org/document/910878, doi:10.1109/34.910878. conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

Bruna, J., Zaremba, W., Szlam, A., LeCun, Y., 2014. Spectral Networks and Locally Connected Networks on Graphs. URL: http://arxiv.org/abs/1312.6203, doi:10.48550/arXiv.1312.6203. arXiv:1312.6203 [cs].

Cao, Z., Hidalgo, G., Simon, T., Wei, S.E., Sheikh, Y., 2019. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. IEEE transactions on pattern analysis and machine intelligence 43, 172–186.

Cao, Z., Simon, T., Wei, S.E., Sheikh, Y., 2017. Realtime multi-person 2d pose estimation using part affinity fields, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7291–7299.

Carreira, J., Noland, E., Banki-Horvath, A., Hillier, C., Zisserman, A., 2018. A short note about kinetics-600. arXiv preprint arXiv:1808.01340 .

Carreira, J., Zisserman, A., 2017. Quo vadis, action recognition? a new model

and the kinetics dataset, in: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6299–6308.

Chen, S., Zheng, D., Ding, C., Huan, C., Ji, Y., Liu, H., 2023. TANGO: rethinking quantization for graph neural network training on GPUs, in: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, Association for Computing Machinery, New York, NY, USA. pp. 1–14. URL: https://dl.acm.org/doi/10.1145/3581784.3607037, doi:10.1145/3581784.3607037.

Chen, T., Fang, C., Shen, X., Zhu, Y., Chen, Z., Luo, J., 2021. Anatomy-aware 3d human pose estimation with bone-based pose decomposition. IEEE Transactions on Circuits and Systems for Video Technology 32, 198–209.

Chi, S., Chi, H.G., Huang, Q., Ramani, K., 2025. Infogcn++: Learning representation by predicting the future for online skeleton-based action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 47, 514–528. doi:10.1109/TPAMI.2024.3466212.

Defferrard, M., Bresson, X., Vandergheynst, P., 2017. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. URL: http://arxiv.org/abs/1606.09375, doi:10.48550/arXiv.1606.09375. arXiv:1606.09375 [cs].

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, Ieee. pp. 248–255.

Deng, Z., Gao, Q., Ju, Z., Yu, X., 2023. Skeleton-based multifeatures and multistream network for real-time action recognition. IEEE Sensors Journal 23, 7397–7409. doi:10.1109/JSEN.2023.3246133.

Dhekane, S.G., Ploetz, T., 2024. Transfer learning in human activity recognition: A survey. arXiv:2401.10185.

Donahue, J., Hendricks, L.A., Rohrbach, M., Venugopalan, S., Guadarrama, S., Saenko, K., Darrell, T., 2017. Long-term recurrent convolutional networks for visual recognition and description. IEEE Transactions on Pattern Analysis and Machine Intelligence 39, 677–691. doi:10.1109/TPAMI.2016.2599174.

Dong, Y., Aristidou, A., Shamir, A., Mahler, M., Jain, E., 2020. Adult2child: Motion style transfer using cyclegans, in: Motion, Interaction and Games, Association for Computing Machinery, New York, NY, USA. URL: https://doi.org/10.1145/3424636.3426909, doi:10.1145/3424636.3426909.

Ehrig, C., Sonnleitner, B., Neumann, U., Cleophas, C., Forestier, G., 2024. The impact of data set similarity and diversity on transfer learning success in time series forecasting. URL: http://arxiv.org/abs/2404.06198, doi:10.48550/arXiv.2404.06198. arXiv:2404.06198 [cs].

Fang, H.S., Li, J., Tang, H., Xu, C., Zhu, H., Xiu, Y., Li, Y.L., Lu, C., 2022. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time. IEEE transactions on pattern analysis and machine intelligence 45, 7157–7173.

Farina, F., Slade, E., 2021. Data efficiency in graph networks through equivariance. URL: http://arxiv.org/abs/2106.13786, doi:10.48550/arXiv.2106.13786. arXiv:2106.13786 [cs].

Feichtenhofer, C., 2020. X3D: Expanding Architectures for Efficient Video Recognition. URL: http://arxiv.org/abs/2004.04730, doi:10.48550/arXiv.2004.04730. arXiv:2004.04730 [cs].

Feng, M., Meunier, J., 2022. Skeleton Graph-Neural-Network-Based Human Action Recognition: A Survey. Sensors 22, 2091. URL: https://www.mdpi.com/1424-8220/22/6/2091, doi:10.3390/s22062091. number: 6 Publisher: Multidisciplinary Digital Publishing Institute.

Fukunaga, K., Hayes, R., 1989. Effects of sample size in classifier design. IEEE Transactions on Pattern Analysis and Machine Intelligence 11, 873–885. URL: https://ieeexplore.ieee.org/document/31448, doi:10.1109/34.31448. conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

Gallicchio, C., Micheli, A., 2010. Graph Echo State Networks, in: The 2010 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. URL: https://ieeexplore.ieee.org/document/5596796, doi:10.1109/IJCNN.2010.5596796. iSSN: 2161-4407.

George, A.M., Banerjee, D., Dey, S., Mukherjee, A., Balamurali, P., 2020. A reservoir-based convolutional spiking neural network for gesture recognition from dvs input, in: 2020 International Joint Conference on Neural Networks (IJCNN), IEEE. pp. 1–9.

Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E., 2017. Neural Message Passing for Quantum Chemistry. URL: http://arxiv.org/abs/1704.01212, doi:10.48550/arXiv.1704.01212. arXiv:1704.01212 [cs].

Gori, M., Monfardini, G., Scarselli, F., 2005. A new model for learning in graph domains, in: Proceedings. 2005 IEEE international joint conference on neural networks, pp. 729–734.

Goto, J., Kidokoro, T., Ogura, T., Suzuki, S., 2013. Activity recognition system for watching over infant children, in: 2013 IEEE RO-MAN, pp. 473–477. doi:10.1109/ROMAN.2013.6628549.

Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al., 2017. The" something something" video database for learning and evaluating visual common sense, in: Proceedings of the IEEE international conference on computer vision, pp. 5842–5850.

Hamilton, W.L., 2020. Graph Representation Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, Springer International Publishing, Cham. URL: https://link.springer.com/10.1007/978-3-031-01588-5, doi:10.1007/978-3-031-01588-5.

Hamilton, W.L., Ying, R., Leskovec, J., 2018. Inductive Representation Learning on Large Graphs. URL: http://arxiv.org/abs/1706.02216, doi:10.48550/arXiv.1706.02216. arXiv:1706.02216 [cs].

Hara, K., Kataoka, H., Satoh, Y., 2018. Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet? URL: http://arxiv.org/abs/1711.09577, doi:10.48550/arXiv.1711.09577. arXiv:1711.09577 [cs].

Hu, L., Liu, S., Feng, W., 2022. Spatial Temporal Graph Attention Network for Skeleton-Based Action Recognition. URL: http:

//arxiv.org/abs/2208.08599, doi:10.48550/arXiv.2208.08599. arXiv:2208.08599 [cs].

Hu, W., Xie, D., Fu, Z., Zeng, W., Maybank, S., 2007. Semantic-based surveillance video retrieval. IEEE Transactions on image processing 16, 1168–1181.

Huan, C., Song, S.L., Liu, Y., Zhang, H., Liu, H., He, C., Chen, K., Jiang, J., Wu, Y., 2023. T-gcn: A sampling based streaming graph neural network system with hybrid architecture, in: Proceedings of the International Conference on Parallel Architectures and Compilation Techniques, Association for Computing Machinery, New York, NY, USA. p. 69–82. URL: https://doi.org/10.1145/3559009.3569648, doi:10.1145/3559009.3569648.

Huang, X., Luan, L., Hatamimajoumerd, E., Wan, M., Kakhaki, P.D., Obeid, R., Ostadabbas, S., 2023a. Posture-based Infant Action Recognition in the Wild with Very Limited Data, in: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 4912–4921. URL: https://ieeexplore.ieee.org/document/10208410, doi:10.1109/CVPRW59228.2023.00519. iSSN: 2160-7516.

Huang, Z., Zhang, S., Pan, L., Qing, Z., Zhang, Y., Liu, Z., Jr, M.H.A., 2023b. Temporally-Adaptive Models for Efficient Video Understanding. URL: http://arxiv.org/abs/2308.05787, doi:10.48550/arXiv.2308.05787. arXiv:2308.05787 [cs].

Hussein, M.E., Torki, M., Gowayyed, M.A., El-Saban, M., 2013. Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations, in: Proceedings of the Twenty-Third international joint conference on Artificial Intelligence, AAAI Press, Beijing, China. pp. 2466–2472.

Huynh-The, T., Hua, C.H., Kim, D.S., 2020. Encoding Pose Features to Images With Data Augmentation for 3-D Action Recognition. IEEE Transactions on Industrial Informatics 16, 3100–3111. URL: https://ieeexplore.ieee.org/document/8691567, doi:10.1109/TII.2019.2910876. conference Name: IEEE Transactions on Industrial Informatics.

Jain, E., Anthony, L., Aloba, A., Castonguay, A., Cuba, I., Shaw, A., Woodward, J., 2016a. Is the motion of a child perceivably different from the motion of an adult? ACM Trans. Appl. Percept. 13. URL: https://doi.org/10.1145/2947616, doi:10.1145/2947616.

Jain, E., Anthony, L., Aloba, A., Castonguay, A., Cuba, I., Shaw, A., Woodward, J., 2016b. Is the Motion of a Child Perceivably Different from the Motion of an Adult? ACM Trans. Appl. Percept. 13, 22:1–22:17. URL: https://doi.org/10.1145/2947616, doi:10.1145/2947616.

Jain, S., Salman, H., Khaddaj, A., Wong, E., Park, S.M., Madry, A., 2022. A Data-Based Perspective on Transfer Learning. URL: http://arxiv.org/abs/2207.05739, doi:10.48550/arXiv.2207.05739. arXiv:2207.05739 [cs].

Johansson, G., 1973. Visual perception of biological motion and a model for its analysis. Perception & psychophysics 14, 201–211. doi:https://doi.org/10.3758/BF03212378.

Kawashima, T., Kawanishi, Y., Ide, I., Murase, H., Deguchi, D., Aizawa, T., Kawade, M., 2017. Action recognition from extremely low-resolution thermal image sequence, in: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE. pp. 1–6.

Kayhan, O.S., Gemert, J.C.v., 2020. On Translation Invariance in CNNs: Convolutional Layers can Exploit Absolute Spatial Location. URL: http://arxiv.org/abs/2003.07064, doi:10.48550/arXiv.2003.07064. arXiv:2003.07064 [cs].

Keskes, O., Noumeir, R., 2021. Vision-based fall detection using st-gcn. IEEE Access 9, 28224–28236. doi:10.1109/ACCESS.2021.3058219.

Kim, H.H., Kim, J.Y., Jang, B.K., Lee, J.H., Kim, J.H., Lee, D.H., Yang, H.M., Choi, Y.J., Sung, M.J., Kang, T.J., Kim, E., Oh, Y.S., Lim, J., Hong, S.B., Ahn, K., Park, C.L., Kwon, S.M., Park, Y.R., 2023. Multiview child motor development dataset for AI-driven assessment of child development. GigaScience 12, giad039. URL: https://doi.org/10.1093/gigascience/giad039, doi:10.1093/gigascience/giad039.

Kipf, T.N., Welling, M., 2016. Semi-Supervised Classification with Graph Convolutional Networks URL: https://arxiv.org/abs/1609.02907, doi:10.48550/ARXIV.1609.02907. publisher: arXiv Version Number: 4.

Kooverjee, N., James, S., Zyl, T.v., 2022. Investigating Transfer Learning in Graph Neural Networks. URL: http://arxiv.org/abs/2202.00740, doi:10.48550/arXiv.2202.00740. arXiv:2202.00740 [cs].

Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T., 2011. Hmdb: A large video database for human motion recognition, in: 2011 International Conference on Computer Vision, pp. 2556–2563. doi:10.1109/ICCV.2011.6126543.

Laptev, I., 2005. On Space-Time Interest Points. International Journal of Computer Vision 64, 107–123. URL: https://doi.org/10.1007/s11263-005-1838-7, doi:10.1007/s11263-005-1838-7.

Lemaignan, S., Edmunds, C.E.R., Senft, E., Belpaeme, T., 2018. The PInSoRo dataset: Supporting the data-driven study of child-child and child-robot social dynamics. PLOS ONE 13, e0205999. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0205999, doi:10.1371/journal.pone.0205999. publisher: Public Library of Science.

Li, C., Zhong, Q., Xie, D., Pu, S., 2018. Co-occurrence Feature Learning from Skeleton Data for Action Recognition and Detection with Hierarchical Aggregation. URL: http://arxiv.org/abs/1804.06055, doi:10.48550/arXiv.1804.06055. arXiv:1804.06055 [cs].

Li, Z., Hoiem, D., 2017. Learning without forgetting. IEEE transactions on pattern analysis and machine intelligence 40, 2935–2947.

Lin, W., Shinde, T.S., Dai, W., Liu, M., He, X., Tiwari, A.K., Xiong, H., 2020. Adaptive lossless compression of skeleton sequences. Signal Processing: Image Communication 80, 115659. URL: https://www.sciencedirect.com/science/article/pii/S0923596519306034, doi:10.1016/j.image.2019.115659.

Liu, C., Hu, Y., Li, Y., Song, S., Liu, J., 2017a. Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding. arXiv preprint arXiv:1703.07475 .

Liu, H., Liu, Y., Ren, M., Wang, H., Wang, Y., Sun, Z., 2025. Revealing

key details to see differences: A novel prototypical perspective for skeleton-based action recognition. URL: https://arxiv.org/abs/2411.18941, arXiv:2411.18941.

Liu, H., Shu, N., Tang, Q., Zhang, W., 2018a. Computational Model Based on Neural Network of Visual Cortex for Human Action Recognition. IEEE Transactions on Neural Networks and Learning Systems 29, 1427–1440. URL: https://ieeexplore.ieee.org/document/7874146, doi:10.1109/TNNLS.2017.2669522. conference Name: IEEE Transactions on Neural Networks and Learning Systems.

Liu, J., Akhtar, N., Mian, A., 2018b. Learning Human Pose Models from Synthesized Data for Robust RGB-D Action Recognition. URL: http://arxiv.org/abs/1707.00823, doi:10.48550/arXiv.1707.00823. arXiv:1707.00823 [cs].

Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.Y., Kot, A.C., 2019. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. IEEE transactions on pattern analysis and machine intelligence 42, 2684–2701.

Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.Y., Kot, A.C., 2020. NTU RGB+D 120: A Large-Scale Benchmark for 3D Human Activity Understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence 42, 2684–2701. URL: http://arxiv.org/abs/1905.04757, doi:10.1109/TPAMI.2019.2916873. arXiv:1905.04757 [cs].

Liu, M., Liu, H., Chen, C., 2017b. Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition 68, 346–362. URL: https://www.sciencedirect.com/science/article/pii/S0031320317300936, doi:10.1016/j.patcog.2017.02.030.

Lu, M., Hu, Y., Lu, X., 2020. Driver action recognition using deformable and dilated faster r-cnn with optimized region proposals. Applied Intelligence 50, 1100–1111. doi:https://doi.org/10.1007/s10489-019-01603-4.

Maqueda, A.I., del Blanco, C.R., Jaureguizar, F., García, N., 2015. Human-action recognition module for the new generation of augmented reality applications, in: 2015 International Symposium on Consumer Electronics (ISCE), IEEE. pp. 1–2.

Martin, P.E., Benois-Pineau, J., Péteri, R., Morlier, J., 2018. Sport action recognition with siamese spatio-temporal cnns: Application to table tennis, in: 2018 International conference on content-based multimedia indexing (CBMI), IEEE. pp. 1–6.

Mohottala, S., Abeygunawardana, S., Samarasinghe, P., Kasthurirathna, D., Abhayaratne, C., 2022a. 2D Pose Estimation based Child Action Recognition, in: TENCON 2022 - 2022 IEEE Region 10 Conference (TENCON), IEEE, Hong Kong, Hong Kong. pp. 1–7. URL: https://ieeexplore.ieee.org/document/9977799/, doi:10.1109/TENCON55691.2022.9977799.

Mohottala, S., Samarasinghe, P., Kasthurirathna, D., Abhayaratne, C., 2022b. Graph neural network based child activity recognition, in: 2022 IEEE International Conference on Industrial Technology (ICIT), pp. 1–8. doi:10.1109/ICIT48603.2022.10002799.

Monfort, M., Andonian, A., Zhou, B., Ramakrishnan, K., Bargal, S.A., Yan, T., Brown, L., Fan, Q., Gutfreund, D., Vondrick, C., et al., 2019. Moments in time dataset: one million videos for event understanding. IEEE transactions on pattern analysis and machine intelligence 42, 502–508.

Moon, S., Kim, M., Qin, Z., Liu, Y., Kim, D., 2023. Anonymization for skeleton action recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 15028–15036.

Müller, R., Kornblith, S., Hinton, G.E., 2019. When does label smoothing help? Advances in neural information processing systems 32.

Ng, J.Y.H., Choi, J., Neumann, J., Davis, L.S., 2018. Actionflownet: Learning motion representation for action recognition, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE. pp. 1616–1624.

Noor, N., Jametoni, F., Kim, J., Hong, H., Park, I.K., 2024. Efficient skeleton-based action recognition for real-time embedded systems, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 5889–5897. doi:10.1109/CVPRW63382.2024.00596.

Olalere, F., Brouwers, V., Doyran, M., Poppe, R., Salah, A.A., 2021a. Video-Based Sports Activity Recognition for Children, in: 2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 1563–1570. URL: https://ieeexplore.ieee.org/document/9689651. iSSN: 2640-0103.

Olalere, F., Brouwers, V., Doyran, M., Poppe, R., Salah, A.A., 2021b. Video-based sports activity recognition for children, in: 2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), IEEE. pp. 1563–1570.

Piergiovanni, A., Kuo, W., Angelova, A., 2023. Rethinking Video ViTs: Sparse Video Tubes for Joint Image and Video Learning, in: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2214–2224. URL: https://ieeexplore.ieee.org/document/10203937, doi:10.1109/CVPR52729.2023.00220. iSSN: 2575-7075.

Potapczynski, A., Qiu, S., Finzi, M., Ferri, C., Chen, Z., Goldblum, M., Bruss, B., Sa, C.D., Wilson, A.G., 2024. Searching for Efficient Linear Layers over a Continuous Space of Structured Matrices. URL: http://arxiv.org/abs/2410.02117, doi:10.48550/arXiv.2410.02117. arXiv:2410.02117 [cs].

Qin, Z., Liu, Y., Perera, M., Anwar, S., Gedeon, T., Ji, P., Kim, D., 2022. Anubis: Review and benchmark skeleton-based action recognition methods with a new dataset. arXiv preprint arXiv:2205.02071 .

Rajagopalan, S., Dhall, A., Goecke, R., 2013. Self-stimulatory behaviours in the wild for autism diagnosis, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 755–761.

Ramesha, M.D.D., Kavindi, M., Somawansa, R.P., Yadav, A., Samarasinghe, P., Wedasinghe, N., Jayasinghearachchi, V., 2022. Children's behavior analysis through smart toys, in: TENCON 2022 - 2022 IEEE Region 10 Conference (TENCON), pp. 1–6. doi:10.1109/TENCON55691.2022.9978116.

Rehg, J., Abowd, G., Rozga, A., Romero, M., Clements, M., Sclaroff, S., Essa, I., Ousley, O., Li, Y., Kim, C., Rao, H., Kim, J.,

Lo Presti, L., Zhang, J., Lantsman, D., Bidwell, J., Ye, Z., 2013. Decoding Children's Social Behavior, pp. 3414–3421. URL: https://openaccess.thecvf.com/content_cvpr_2013/html/Rehg_Decoding_Childrens_Social_2013_CVPR_paper.html.

Rodomagoulakis, I., Kardaris, N., Pitsikalis, V., Mavroudi, E., Katsamanis, A., Tsiami, A., Maragos, P., 2016. Multimodal human action recognition in assistive human-robot interaction, in: 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE. pp. 2702–2706.

Romero, D.W., 2024. The Good, The Efficient and the Inductive Biases: Exploring Efficiency in Deep Learning Through the Use of Inductive Biases. Ph.D. thesis. URL: http://arxiv.org/abs/2411.09827, doi:10.5463/thesis.738. pages: HFDRN95020240910 arXiv:2411.09827 [cs].

Roy, D., Komini, V., Girdzijauskas, S., 2022. Out-of-distribution in human activity recognition, in: 2022 Swedish Artificial Intelligence Society Workshop (SAIS), pp. 1–10. doi:10.1109/SAIS55783.2022.9833052.

Sárándi, I., Linder, T., Arras, K.O., Leibe, B., 2020. Metrabs: metric-scale truncation-robust heatmaps for absolute 3d human pose estimation. IEEE Transactions on Biometrics, Behavior, and Identity Science 3, 16–30.

Scarselli, F., Gori, M., Ah Chung Tsoi, Hagenbuchner, M., Monfardini, G., 2009. The Graph Neural Network Model. IEEE Transactions on Neural Networks 20, 61–80. URL: http://ieeexplore.ieee.org/document/4700287/, doi:10.1109/TNN.2008.2005605.

Sciortino, G., Farinella, G.M., Battiato, S., Leo, M., Distante, C., 2017. On the Estimation of Children's Poses, in: Battiato, S., Gallo, G., Schettini, R., Stanco, F. (Eds.), Image Analysis and Processing - ICIAP 2017, Springer International Publishing, Cham. pp. 410–421.

Shahroudy, A., Liu, J., Ng, T.T., Wang, G., 2016. Ntu rgb+ d: A large scale dataset for 3d human activity analysis, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1010–1019.

Shi, L., Zhang, Y., Cheng, J., Lu, H., 2019. Two-stream adaptive graph convolutional networks for skeleton-based action recognition, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 12026–12035.

Shi, L., Zhang, Y., Lu, H., Cheng, J., 2020. Skeleton-based action recognition with multi-stream adaptive graph convolutional networks. IEEE Transactions on Image Processing 29, 9532–9545.

Shi, W., Li, D., Wen, Y., Yang, W., 2023. Occlusion-Aware Graph Neural Networks for Skeleton Action Recognition. IEEE Transactions on Industrial Informatics 19, 10288–10298. URL: https://ieeexplore.ieee.org/abstract/document/10029897?casa_token=hboIWXZ1vKsAAAAA:wpojI1c5Yvy2_QtlY1X5h55_yO3T9lTgKv8THN_aWUL9HCPoz_FNC4rVWXltMgiNfphKs6rQPaX2Sw, doi:10.1109/TII.2022.3229140. conference Name: IEEE Transactions on Industrial Informatics.

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A., 2011. Real-time human pose recognition in parts from single depth images, in: CVPR 2011, Ieee. pp. 1297–1304.

Simonyan, K., Zisserman, A., 2014. Two-Stream Convolutional Networks for Action Recognition in Videos. URL: http://arxiv.org/abs/1406.2199, doi:10.48550/arXiv.1406.2199. arXiv:1406.2199 [cs].

Song, Y.F., Zhang, Z., Shan, C., Wang, L., 2021. Richly Activated Graph Convolutional Network for Robust Skeleton-based Action Recognition. IEEE Transactions on Circuits and Systems for Video Technology 31, 1915–1925. URL: http://arxiv.org/abs/2008.03791, doi:10.1109/TCSVT.2020.3015051. arXiv:2008.03791 [cs].

Soomro, K., Zamir, A.R., Shah, M., 2012. Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 .

Srivastava, S., Sharma, G., 2024. OmniVec2 - A Novel Transformer Based Network for Large Scale Multimodal and Multitask Learning, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 27402–27414. URL: https://ieeexplore.ieee.org/document/10655590, doi:10.1109/CVPR52733.2024.02588. iSSN: 2575-7075.

Sun, Z., Ke, Q., Rahmani, H., Bennamoun, M., Wang, G., Liu, J., 2022. Human action recognition from various data modalities: A review. IEEE transactions on pattern analysis and machine intelligence 45, 3200–3225.

Suzuki, S., Amemiya, Y., Sato, M., 2019. Enhancement of gross-motor action recognition for children by cnn with openpose, in: IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society, IEEE. pp. 5382–5387.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C., 2018. A survey on deep transfer learning, in: International conference on artificial neural networks, Springer. pp. 270–279.

Tsiami, A., Koutras, P., Efthymiou, N., Filntisis, P.P., Potamianos, G., Maragos, P., 2018. Multi3: Multi-Sensory Perception System for Multi-Modal Child Interaction with Multiple Robots, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 4585–4592. URL: https://ieeexplore.ieee.org/document/8461210, doi:10.1109/ICRA.2018.8461210. iSSN: 2577-087X.

Vatavu, R.D., 2019. The dissimilarity-consensus approach to agreement analysis in gesture elicitation studies, in: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA. p. 1–13. URL: https://doi.org/10.1145/3290605.3300454, doi:10.1145/3290605.3300454.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y., 2017. Graph attention networks. arXiv preprint arXiv:1710.10903 .

Vishwakarma, S., Agrawal, A., 2013. A survey on activity recognition and behavior understanding in video surveillance. The Visual Computer 29, 983–1009.

Wang, B., Chen, J., Li, C., Zhou, S., Shi, Q., Gao, Y., Feng, Y., Chen, C., Wang, C., 2024a. Distributionally Robust Graph-based Recommendation System. URL: http://arxiv.org/abs/2402.12994, doi:10.48550/arXiv.2402.12994. arXiv:2402.12994 [cs].

Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M., 2022. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696 .

Wang, H., Kläser, A., Schmid, C., Liu, C.L., 2011. Action recognition by dense trajectories, in: CVPR 2011, pp. 3169–3176. URL: https://ieeexplore.ieee.org/document/5995407, doi:10.1109/CVPR.2011.5995407. iSSN: 1063-6919.

Wang, L., Qiao, Y., Tang, X., 2015. Action Recognition with Trajectory-Pooled Deep-Convolutional Descriptors, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4305–4314. URL: http://arxiv.org/abs/1505.04868, doi:10.1109/CVPR.2015.7299059. arXiv:1505.04868 [cs].

Wang, M., Zhang, Y.D., Cui, G., 2019a. Human motion recognition exploiting radar with stacked recurrent neural network. Digital Signal Processing 87, 125–131.

Wang, X., Miao, Z., Zhang, R., Hao, S., 2019b. I3d-lstm: A new model for human action recognition, in: IOP Conference Series: Materials Science and Engineering, IOP Publishing. p. 032035.

Wang, Y., Li, K., Li, X., Yu, J., He, Y., Wang, C., Chen, G., Pei, B., Yan, Z., Zheng, R., Xu, J., Wang, Z., Shi, Y., Jiang, T., Li, S., Zhang, H., Huang, Y., Qiao, Y., Wang, Y., Wang, L., 2024b. InternVideo2: Scaling Foundation Models for Multimodal Video Understanding. URL: http://arxiv.org/abs/2403.15377, doi:10.48550/arXiv.2403.15377. arXiv:2403.15377 [cs].

Wang, Y., Yasunaga, M., Ren, H., Wada, S., Leskovec, J., 2023. VQA-gnn: Reasoning with Multimodal Knowledge via Graph Neural Networks for Visual Question Answering. URL: http://arxiv.org/abs/2205.11501, doi:10.48550/arXiv.2205.11501. arXiv:2205.11501 [cs].

Wu, H., Xin, L., 2025. Cluster-hgnn: Deep local features clustering for few-shot image classification with hybrid graph neural networks. IEEE Access 13, 30965–30975. doi:10.1109/ACCESS.2025.3538610.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Yu, P.S., 2021. A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning Systems 32, 4–24. URL: http://arxiv.org/abs/1901.00596, doi:10.1109/TNNLS.2020.2978386. arXiv:1901.00596 [cs].

Xie, J., Meng, Y., Zhao, Y., Nguyen, A., Yang, X., Zheng, Y., 2024. Dynamic semantic-based spatial graph convolution network for skeleton-based human action recognition, in: Proceedings of the AAAI conference on artificial intelligence, pp. 6225–6233.

Xu, D., Xiao, X., Wang, X., Wang, J., 2016. Human action recognition based on Kinect and PSO-SVM by representing 3D skeletons as points in lie group, in: 2016 International Conference on Audio, Language and Image Processing (ICALIP), pp. 568–573. URL: https://ieeexplore.ieee.org/document/7846646, doi:10.1109/ICALIP.2016.7846646.

Xu, K., Hu, W., Leskovec, J., Jegelka, S., 2019. How Powerful are Graph Neural Networks? URL: http://arxiv.org/abs/1810.00826, doi:10.48550/arXiv.1810.00826. arXiv:1810.00826 [cs].

Yan, S., Xiong, Y., Lin, D., 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition, in: Thirty-second AAAI conference on artificial intelligence.

Yaras, C., Wang, P., Balzano, L., Qu, Q., 2024. Compressible Dynamics in Deep Overparameterized Low-Rank Learning & Adaptation. URL: http://arxiv.org/abs/2406.04112, doi:10.48550/arXiv.2406.04112. arXiv:2406.04112 [cs].

Yu, S., Cheng, Y., Su, S., Cai, G., Li, S., 2017. Stratified pooling based deep convolutional neural networks for human action recognition. Multimedia Tools and Applications 76, 13367–13382. URL: https://doi.org/10.1007/s11042-016-3768-5, doi:10.1007/s11042-016-3768-5.

Zahan, S., Gilani, Z., Hassan, G.M., Mian, A., 2023a. Human gesture and gait analysis for autism detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3328–3337.

Zahan, S., Gilani, Z., Hassan, G.M., Mian, A., 2023b. Human Gesture and Gait Analysis for Autism Detection, pp. 3328–3337. URL: https://openaccess.thecvf.com/content/CVPR2023W/FGAHI/html/Zahan_Human_Gesture_and_Gait_Analysis_for_Autism_Detection_CVPRW_2023_paper.html.

Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O., 2021a. Understanding deep learning (still) requires rethinking generalization. Communications of the ACM 64, 107–115.

Zhang, E., Lepori, M.A., Pavlick, E., 2024. Instilling Inductive Biases with Subnetworks. URL: http://arxiv.org/abs/2310.10899, doi:10.48550/arXiv.2310.10899. arXiv:2310.10899 [cs].

Zhang, J., Li, W., Ogunbona, P., Xu, D., 2019. Recent advances in transfer learning for cross-dataset visual recognition: A problem-oriented perspective. ACM Computing Surveys (CSUR) 52, 1–38.

Zhang, Y., Tian, Y., Wu, P., Chen, D., 2021b. Application of skeleton data and long short-term memory in action recognition of children with autism spectrum disorder. Sensors 21. doi:10.3390/s21020411.

Zheng, Y., Huang, H., Wang, X., Yan, X., Xu, L., 2024. Spatio-Temporal Fusion for Human Action Recognition via Joint Trajectory Graph. Proceedings of the AAAI Conference on Artificial Intelligence 38, 7579–7587. URL: https://ojs.aaai.org/index.php/AAAI/article/view/28590, doi:10.1609/aaai.v38i7.28590. number: 7.

Zhou, Q., Yu, S., Wu, X., Gao, Q., Li, C., Xu, Y., 2009. HMMs-based human action recognition for an intelligent household surveillance robot, in: 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 2295–2300. URL: https://ieeexplore.ieee.org/document/5420459, doi:10.1109/ROBIO.2009.5420459.

Zhou, Y., Yan, X., Cheng, Z.Q., Yan, Y., Dai, Q., Hua, X.S., 2024. Blockgcn: Redefine topology awareness for skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2049–2058.

Zhu, Y., Xu, W., Zhang, J., Du, Y., Zhang, J., Liu, Q., Yang, C., Wu, S., 2022. A Survey on Graph Structure Learning: Progress and Opportunities. URL: http://arxiv.org/abs/2103.03036, doi:10.48550/arXiv.2103.03036. arXiv:2103.03036 [cs].