Wearable Sensor-Based System for Real-Time Human Activity Recognition

Mark Shperkin
Department of Computer Science and Engineering
University of South Carolina
Columbia, SC 29208

***Abstract*—**This research originally proposed the development of a wearable motion capture system designed for real-time human activity classification. The system employed a network of 9-degree-of-freedom (9DOF) absolute sensors—comprising gyroscopes, accelerometers, and magnetometers—strategically positioned at key joints across the human body. These sensors, based on BNO055 and interfaced through Arduino MKR IMU and ESP32 boards, were attempted to collect detailed motion data that is used to detect and reconstruct the user's pose in three-dimensional space. The captured data would serve as input for training a machine learning model capable of classifying various human movements, such as walking, running, and jumping. However, due to inconsistent sensor readings (noise, and drift), the hardware data proven unsuitable for model training. Thus, the focus shifted to implementing and evaluation a Spatio-Teporal Graph Convolutional Network (ST-GCN) [2] on standard benchmark NTU RGB+D [4] skeletal dataset, achieving 87.6 % classification accuracy across 49 different activities [5]. This result underscores the promise of ST-GCNs for real-time human action recognition, with potential applications in medical monitoring (e.g., distress-gesture detection in clinical and assisted-living environments), security surveillance in high-risk facilities, and objective sports performance analysis.

Keywords*—Human Activity Recognition, Wearable Motion Capture, 9DOF Sensor Network, Machine Learning Classification, Real-Time Pose Estimation.*

I. INTRODUCTION

Advancements in wearable technology and embedded sensing have led to significant progress in real-time human motion tracking and activity recognition. Inertial Measurement Units (IMUs), which combine accelerometers, gyroscopes, and magnetometers, provide a compact, low-cost, and portable solution for capturing motion data without the need for external reference systems. These devices enable continuous tracking of human pose and orientation, even in environments where optical systems are impractical or unavailable. However, challenges such as sensor noise, drift, and the nonlinear nature of human movement require sophisticated estimation techniques to extract meaningful insights from raw IMU data.

The integration of sensor fusion algorithms like the Extended Kalman Filter (EKF) has improved the reliability of motion tracking by correcting noisy measurements and accommodating nonlinear system dynamics. Recent studies have demonstrated that strategically placed IMUs on the human body—particularly on key joints—can yield highly accurate estimations of joint angles, limb orientation, and displacement. For example, dual-stage EKF models with novel sensor placement strategies have shown up to 30 % improvement in position estimation over traditional dead reckoning methods, reinforcing the importance of both algorithmic and hardware configuration in motion capture systems [1]. In our experiments, the prototype sensor network produced pose streams with excessive noise and drift, rendering them unreliable for downstream model input.

In the realm of activity recognition, machine learning techniques have become increasingly essential. With the proliferation of data-driven approaches, frameworks such as spatial-temporal graph convolutional networks (ST-GCNs) have demonstrated remarkable performance in classifying human actions from skeletal and joint data, particularly in healthcare and sports applications [2]. These models leverage the spatial relationships between joints and their temporal dynamics to learn robust representations of human movement. The capability to identify movements such as walking, running, or jumping in real-time enables transformative applications in athletic performance monitoring, injury prevention, and personalized training program design.

This research originally aimed at building upon these foundational concepts by presenting a wearable motion capture system that employs a network of 9DOF BNO055 sensors on an Arduino MKR IMU platform, wirelessly connected to ESP32 boards. However, accurate estimation challenges prevented full hardware deployment. As a result, a publicly available dataset was used to train a machine learning model for human activity classification. The ultimate goal was to provide an affordable and effective solution for biomechanical analysis, enabling athletes and coaches to optimize performance and minimize injury risk through precise, data-driven insights.

II. LITERATURE REVIEW

Yadav and Bleakley introduce a novel two-stage Extended Kalman Filter (EKF) that incorporates a unique sensor placement constraint on a rigid body to improve position estimation using inertial measurement units (IMUs) [1]. Their approach leverages two spatially separated IMUs mounted on a single rigid body, enabling the system to maintain known relative positions and orientations between sensors. In the first stage, orientation is estimated using gyroscope and accelerometer data. The second stage refines position estimates by integrating the fixed inter-sensor distance as a constraint within the Kalman filter's observation model. This dual-stage strategy significantly reduces cumulative drift and outperforms traditional dead reckoning methods, showing up to 30 % improvement in position accuracy across various motion scenarios. This work provides a valuable foundation for wearable motion capture systems, particularly in scenarios lacking external reference systems.

Bennett et al. propose a method to estimate human gait parameters and walking distance using a biomechanical model of the leg, modeled as a two-link robotic manipulator [3]. They place gyroscopes on the thigh and shin, using angular velocity readings in conjunction with forward kinematics and an EKF to track joint angles and displacement. Their experiments demonstrated high accuracy, achieving an average root-mean-square error (RMSE) of just 7 cm, with over 97 % accuracy in walking distance estimation. The simplicity and effectiveness of their EKF-based approach show how meaningful biomechanical data can be extracted from minimal, wearable sensor setups, an idea central to our project's wearable system for classifying human movement.

Ghosh and colleagues propose a spatio-temporal graph convolutional network (STGCN) architecture tailored for human action recognition (HAR) using skeleton data who preserve critical joint feature while cutting compute and memory costs [2]. Their model introduces a novel feature extraction approach by independently processing spatial and temporal information, which helps reduce feature loss and enhances accuracy. This modular design improves flexibility and reduces computational overhead. Achieving 92.2 % accuracy on the NTU-RGBD dataset [4]. The model also performs well on edge devices, making it ideal for real-time applications such as patient monitoring or athletic performance analysis. These insights directly inform the machine learning component of our project, offering a scalable and accurate solution for classifying human motion in real time based on wearable sensor data.

Shahroudy et al. introduce the NTU RGB+D dataset—a large-scale benchmark comprising 56,880 RGB-D video samples and over 4 million frames across 60 distinct action classes (40 daily, 9 health-related, 11 mutual) captured from 40 subjects using Kinect v2 sensors under 80 camera viewpoints [4]. Each sample provides synchronized modalities: RGB frames, depth maps, infrared sequences, and 3D joint coordinates for 25 body landmarks, enabling rich spatio-temporal analysis. A key strength of the 3D skeletal dataset is the data representation, where human joints are modeled as graph vertices, with spatial edges representing anatomical structure and temporal edges linking the same joints across sequential frames. This enables learning Spatial dynamics by paying attention to joint relationships within each frame, while temporal patterns are analyzed across time.

III. Proposed Work

Building on Yadav and Bleakley's two-stage Extended Kalman Filter for dual-IMU position estimation [1] and Bennett el al.'s leg-kinematic EKF for gait parameters [3], we first developed a wearable 9-DOF sensor network to track the user's 3D position. However, both approaches suffer from unbounded drift, twice integrating noisy accelerometer signals causes small errors to accumulate exponentially, making sustained, accurate 3D tracking infeasible. Even after implementing zero-velocity updates and magnetometer-based heading corrections, residual bias and noise persisted, and overall drift remained unacceptably high. As a result, pose reconstructions were unstable and visually unreliable, preventing the creation of a valid dataset or real-time, suit-based pose tracking.

With a view toward real-time human activity recognition on resource-constrained edge device, we focused exclusively on implementing the Spatio-Temporal Graph Convolutional Network (STGCN), an architecture designed to maximize representational power while minimizing parameter count and model complexity [3]. STGCN's core innovation is its parallel, attention-driven feature extractors—a spatial convolutional layer based on adaptive graph convolutions—learns a data depended on adjacency (and thus "attends" to the most informative joint relationships), and a lightweight temporal convolutional layer efficiently captures motion dynamics across time. By decoupling spatial and temporal streams and embedding attention mechanisms directly into the graph operations, STGCN delivers high accuracy with a compact model footprint, making it ideal for deployment on IoMT and other edge platforms.

The spatial convolution layer is built on a three matrices, Ak, Bk and Ck. Ak, the physical skeleton graph used a predefined binary adjacency matrix $\bar{A}k$, where $\bar{A}ijk=1$ if joints i and j are naturally connected in the human body, and 0 otherwise. To ensure stable, symmetric message-passing, we normalize this as

$$Ak=\Lambda k\text{^}(-21)\ \bar{A}k\ \Lambda k\text{^}(-21),$$

Where Λk is defined as

$$\Lambda iik=j\sum(\bar{A}ijk)+\alpha,$$

With $\alpha = 0.003$ to prevent zero-degree rows. This symmetric normalization balances contributions from high and low degree joints and prevents numerical instabilities. Whereas Ak is fixed by anatomy, $Bk \in R$ with shape of N×N is initialized to zero and learned end to end. In effect, it acts like an attention mask over edges, allowing the network to strengthen or weaken particular joint to joint connections in a class and layer specific manner, thereby capturing correlations that go beyond the physical skeleton. Finnlay, Ck dynamically adapts to each input sample by measuring feature similarity in an embedded gaussian space. Concretely, the incoming feature map fin ∈ R^(Cin×T×N) is first projected via two separate 1×1 convolutions θk, ϕk into an embedding of size Cem, then reshaped to N×(Cem T) and (CemT)×N. their dot=product produces an unnormalized affinity matrix, in the shape of N×N, Which is normalized with a softmax over each row to yield Ck. This lets the model capture context-dependent relationships. The adaptive adjacency is then simply Ak+Bk+Ck, followed by Wk who is a 1×1 convolution and added a residual branch closes the loop. Please see figure 1. By fusing fixed, learned, and dynamic graphs in this way, the spatial convolution layer is able to extract far richer, more discriminative joint features than a purely learned adjacency could achieve.

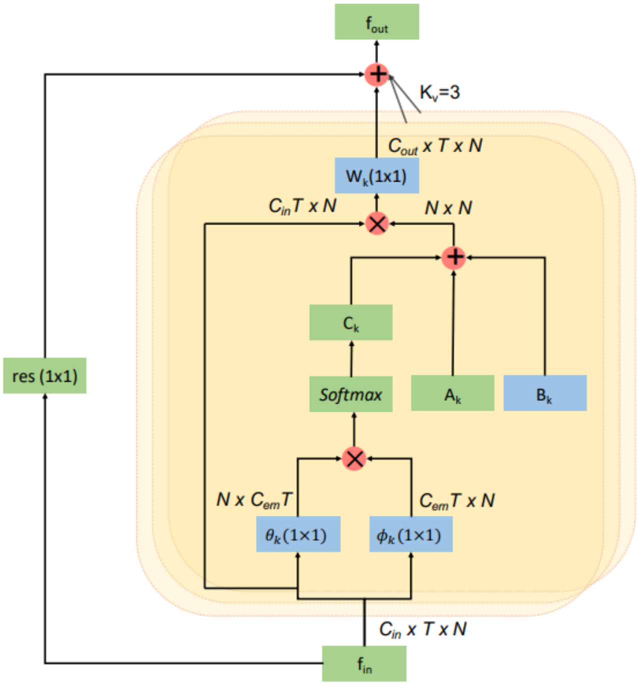


Figure 1. architecture of an adaptive graph convolutional layer [2].

The temporal convolution layer operated purely along the time axis. It takes the features of shape C × T × N where C is the number of channels, T the number of frames, and N and number of joints, we apply a 2D convolution with kernel size Kt × 1 (spanning Kt frames but only a single joint at a time). This preserves the joint dimension while capturing temporal patterns.

To complete the STGCN block, we process the input via the spatial and temporal convolution layers in parallel streams. Their outputs are then batch-normalized and ReLU activated, concatenated channel wise, and passed through a 1 × 1 convolution to fuse and reduce the combined feature dimensionality. Finally, to improve performance and ensure robust gradient flow, a residual connection is added to the block. Please see figure 2.

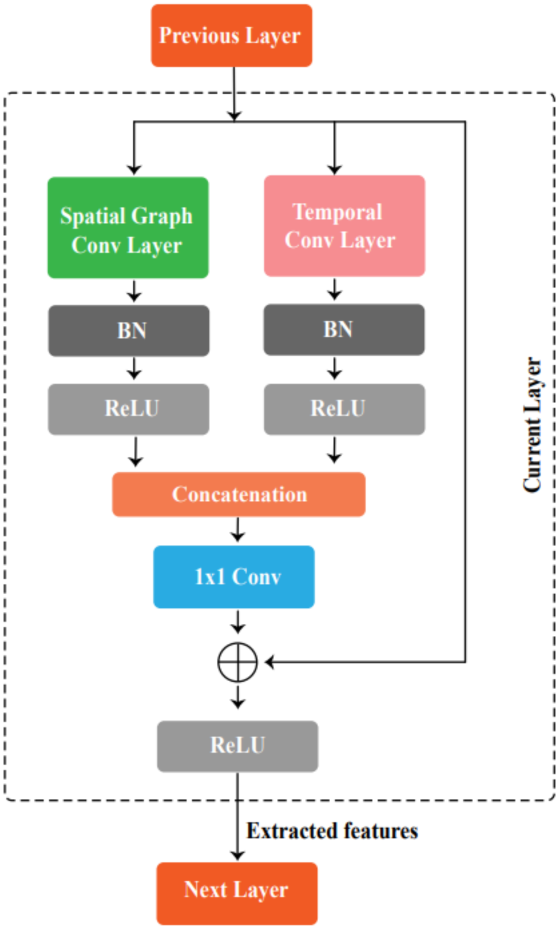


Figure 2. A spatio-temporal graph convolutional block of STGCN [2].

The complete architecture comprises of ten sequential spatial-temporal convolutional blocks. Starting with an initial batch normalization layer to stable the training, and end with a fully connected layer with a softmax function that generates the class probability distribution for action prediction. Please see figure 3.

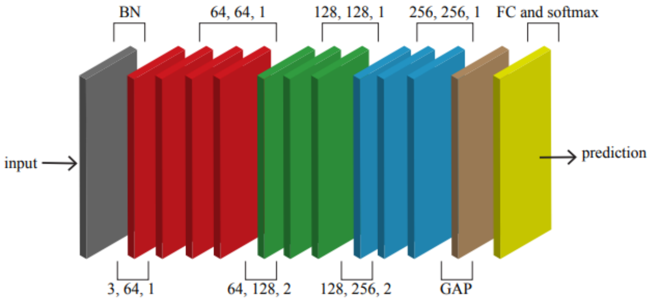


Figure 3. The architecture of STGCN [2].

Training was carried out in PyTorch using stochastic gradient descent with Nesterov Momentum of 0.9 and a weight decay of 1 × 10^-4. The initial learning rate was set to 0.01 and was reduced by a factor of ten at epochs 30 and 40. the model was trained for 50 epochs with a batch size of 16, and cross entropy loss was used for gradient backpropagation. To accommodate the fixed frame input requirement, any sequence shorten than 300 frames was temporally replicated until this length was reached. Finally, an 80/20 split between training and validation dataset yielded a peak validation accuracy of 87.6 % by epoch 49 [5].

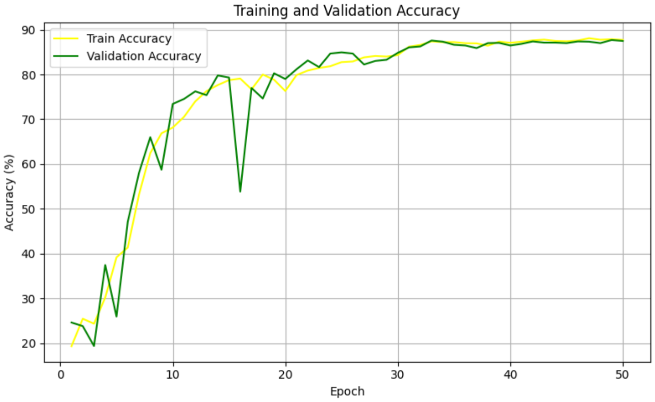


Figure 4. training and validation accuracy over 50 epochs [5].

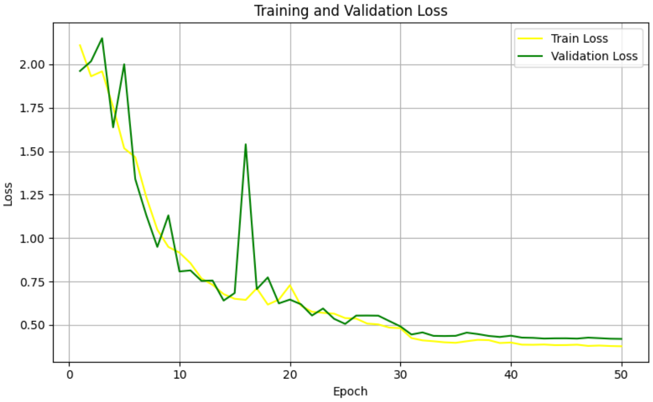
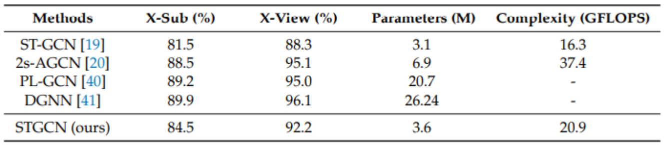


Figure 5. training and validation loss over 50 epochs [5].

IV. Results

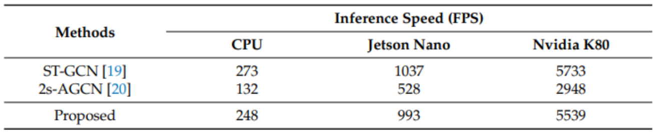
The experimental results demonstrate that STGCN achieves strong recognition performance while drastically reducing both model size and computational cost [2] On the NTU RGB-D benchmark [4], STGCN attains 84.5 % cross-subject (X-sub) and 92.2 % cross-view (X-view) accuracy, matching or closely approaching the accuracy of heavier GCN variants such as 2s-AGCN's 88.5 % / 95.1 %. These findings align with our own validation accuracy of 87.6 % [5].



Methods	X-Sub (%)	X-View (%)	Parameters (M)	Complexity (GFLOPS)
ST-GCN [19]	81.5	88.3	3.1	16.3
2s-AGCN [20]	88.5	95.1	6.9	37.4
PL-GCN [40]	89.2	95.0	20.7	-
DGNN [41]	89.9	96.1	26.24	-
STGCN (ours)	84.5	92.2	3.6	20.9

Table 1. Comparison of STGNC with state-of-the-art GCN based methods on the NTU_RGB-D dataset [2].

Crucially, this lean design translates to real-time feasibility on edge hardware, on an NVIDIA Jetson Nano, STGCN processes 993 frames per second, more than sufficient for live, real-time deployment in resource constrained IoMT settings.



Methods	Inference Speed (FPS)		
	CPU	Jetson Nano	Nvidia K80
ST-GCN [19]	273	1037	5733
2s-AGCN [20]	132	528	2948
Proposed	248	993	5539

Table 2. Comparisons of STGCN with state-of-the-art GCN based methods in terms of inference speed [2].

We have conducted our own inference speed results and concluded that using NVIDIA 4070 Super has an average sample latency of 0.009 seconds and a throughput of 102.8 samples a second, while, on Intel i7-14700F CPU average latency per sample is 0.14 seconds with a throughput of 6.8 samples per second. We take this into consideration where each sample has 300 frames. This experiment reinforces the ability of the STGCN architecture to be deployed on edge devise.

Device	**Average latency/batch**	**Average latency/sample**	**Throughput**
GPU	0.155592s (±0.015316)	0.009725s	102.8 samples/s
CPU	2.353588s (±0.036606)	0.147099s	6.8 samples/s

Table 3. Comparison of GPU and CPU inference experiment [5].

Together, these findings confirm that the adaptive spatial temporal architecture delivers competitive accuracy at a fraction of size and cost, making it well suited for on device inference.

V. Conclusion and Future Work

Although we initially explored two data-fusion strategies, a two-state Kalman filter applied to dual IMU streams [1] and an extended Kalman filter [3], their performance fell short of expectations. This was likely due both to the high noise levels in our acquired IMU measurements and to ambiguities and missing details in the published methodologies. In contrast, the novel skeleton-based human action recognition architecture presented here employs spatial-temporal graph convolutions to capture joint dependencies directly from the graph-structured data [2]. The model achieves competitive accuracy against state-of-the-art baselines while maintaining a lightweight footprint suitable for edge deployment: it requires less memory, demands lower computational resources, and eliminates extensive preprocessing overhead. These findings emphasize the promise of graph-based neural networks for efficient, on-device biomechanical analysis.

Looking ahead, there are several exciting directions in which the STGCN architecture can be extended [2]. In the medical domain, the model could be adapted for real-time patient monitoring in clinical and assisted-living settings—detecting sudden falls, distress gestures, or other emergencies and automatically triggering alerts to caregivers. In parallel, deploying STGCN in high-security environments (airports, stadiums, border checkpoints, and critical infrastructure) could enable real-time classification and flagging of anomalous behaviors indicative of malicious intent, supporting proactive threat interception.

Beyond safety and security, STGCN's powerful spatial-temporal skeleton encoding lends itself to richer analytics across sports and performance arts. For example, in gymnastics, diving, or dance, the network could evaluate movement quality, provide automated skill validation, and even augment or partially replace human judges. More generally, by fine-tuning to diverse joint-motion patterns, STGCN could serve as a versatile skeleton-based analytics engine—from athletic performance assessment to ergonomics monitoring in industrial settings—opening up new possibilities for data-driven evaluation and feedback.

VI. Summary of Contributions

All research, development, and implementation duties for this project were undertaken solely by Mark Shperkin, who managed every aspect—from system design and sensor integration to machine-learning model development and evaluation. Although the final outcomes did not fully meet the original project goals, Mark Shperkin takes full responsibility and has distilled key lessons to guide future research.

References

[1] N. Yadav and C. Bleakley, "Two Stage Kalman Filtering for Position Estimation Using Dual Inertial Measurement Units," *Proc. of UCD Complex and Adaptive Systems Laboratory*, University College Dublin, Ireland.

[2] D. K. Ghosh et al., "A Spatio-Temporal Graph Convolutional Network Model for Internet of Medical Things (IoMT)," *Sensors*, vol. 22, no. 8438, 2022. doi: 10.3390/s22218438

[3] T. Bennett, R. Jafari, and N. Gans, "An Extended Kalman Filter to Estimate Human Gait Parameters and Walking Distance," *Proc. American Control Conf.*, 2013, pp. 752–757. doi:10.1109/ACC.2013.6579926

[4] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: a large-scale dataset for 3D human activity analysis," 2016. https://openaccess.thecvf.com/content_cvpr_2016/html/Shahroudy_NTU_RGBD_A_CVPR_2016_paper.html

[5] Mark Shperkin GitHub Repository for this project. https://github.com/markshperkin/HAR-STGCN