Human Activity Recognition Using Multi-Sensor Accelerometry Data: A Deep Learning Approach

Andrea Manocchio
andrea.manocchio@studenti.unipd.it

Giovanni Passaro
giovanni.passaro@studenti.unipd.it

September 15, 2025

Abstract

Human Activity Recognition (HAR) has become increasingly important in healthcare monitoring, fitness tracking, and assistive technologies. This study presents a comparison of deep learning architectures for multi-sensor accelerometry-based activity recognition. We evaluate Convolutional Neural Networks (CNNs), and hybrid CNN-LSTM models on a dataset containing synchronized accelerometer data from four body locations. Our methodology addresses critical challenges in time-series classification including class imbalance, data leakage prevention, and overfitting mitigation. Results demonstrate that CNN architectures with appropriate regularization achieve optimal performance (75-82% accuracy) while maintaining computational efficiency.

1 Introduction

1.1 Background and Motivation

Human Activity Recognition (HAR) represents a fundamental challenge in ubiquitous computing, with applications spanning healthcare monitoring, fitness tracking, elderly care, and rehabilitation. The proliferation of wearable sensors, particularly accelerometers, has made continuous activity monitoring feasible and accessible. However, accurately classifying complex human movements from raw sensor data remains challenging due to inter-individual variations, sensor noise, and the temporal nature of human activities.

Traditional HAR approaches relied heavily on hand-crafted features extracted from accelerometer signals, such as statistical measures, frequency domain characteristics, and time-domain patterns. While effective for simple activities, these methods struggle with complex, multi-phase activities and require extensive domain expertise for feature engineering.

1.2 Research Objectives

This study addresses the following research questions:

1. **Architecture Comparison**: How do different deep learning architectures (CNN, LSTM, Hybrid) perform on multi-sensor accelerometry data?
2. **Data Balancing Impact**: What is the effect of various class balancing techniques on model performance in imbalanced HAR datasets?
3. **Overfitting Prevention**: How can we prevent data leakage and overfitting in time-series HAR applications?

1.3 Contributions

Our main contributions include:

- **Novel Architecture Design**: Custom deep learning architectures optimized for multi-sensor accelerometry data
- **Comprehensive Balancing Analysis**: Systematic evaluation of data balancing techniques for imbalanced HAR datasets
- **Overfitting Prevention Framework**: Participant-aware data splitting methodology to prevent temporal data leakage

2 Related Work

2.1 Traditional HAR Approaches

Early HAR systems predominantly used statistical and frequency-domain features extracted from accelerometer data. Bao and Intille (2004) [1] demonstrated the effectiveness of decision trees using mean, variance, and correlation features. Ravi et al. (2005) [5] explored various machine learning algorithms including SVMs and k-NN for activity classification.

2.2 Deep Learning in HAR

The introduction of deep learning to HAR began with basic feedforward networks. Yang et al. (2015) [6] first applied CNNs to accelerometer data, treating sensor readings as 1D signals. Hammerla et al. (2016) [3] conducted the first comprehensive comparison of deep learning architectures for HAR, including CNNs, LSTMs, and hybrid approaches.

3 Processing Pipeline

3.1 Data Cleaning

- **Missing Value Treatment**: Forward-fill interpolation for sparse missing values
- **Outlier Detection**: Statistical outlier removal using z-score thresholding ($z > 3$), to remove potential data coming from incorrect sensor placement or malfunction
- **Signal Smoothing**: Moving average filtering to reduce sensor noise

3.2 Normalization

We applied standardization (z-score normalization) to each accelerometer axis independently:

$$x_{\text{normalized}} = \frac{x - \mu}{\sigma} \tag{1}$$

This approach ensures consistent scale across different sensors while preserving relative signal characteristics.

3.3 Windowing Strategy

- **Window Size**: 25 samples (0.25 seconds at 100 Hz)
- **Overlap**: 10% to minimize redundancy while maintaining temporal continuity
- **Validation**: Windows containing multiple activities were excluded to ensure label consistency

3.4 Limitations

Data collected from the study were 31 right-handed participants and one ambidextrous. Future works should look to obtain a more diverse knowledge base.

4 Signals and features

We utilized the PhysioNet Accelerometry Dataset [4], containing labeled accelerometer data from 32 healthy adults performing various activities. The dataset provides:

- **Participants**: 32 adults (13 male, 19 female, ages 23-52)
- **Sensors**: 4 synchronized ActiGraph GT3X+ accelerometers
- **Locations**: Left wrist, left hip, left ankle, right ankle

- **Sampling Rate**: 100 Hz
- **Activities**: Walking, ascending stairs, descending stairs, driving
- **Additional labels**: The dataset also contains sound data labeled as Clapping and Non-Study activity, they were not included because of non informative value
- **Duration**: 9-13 minutes walking, 18-30 minutes driving per participant

4.1 Data Characteristics and Challenges

4.1.1 Class Imbalance

The dataset exhibits severe class imbalance as shown in Table 1:

Table 1: Class Distribution in the Dataset

Activity	Samples	Percentage
Driving	59,821	64.4%
Walking	24,992	26.9%
Descending stairs	3,979	4.3%
Ascending stairs	4,166	4.5%

4.2 Participant-Aware Data Splitting

Traditional random splitting can place the same participant's data in both training and test sets, leading to temporal data leakage and inflated performance metrics. Given the origin of the data, and the restricted sample of population, we decided to go for a participant aware split. This approach ensures true generalization to unseen individuals, providing realistic performance estimates for deployment scenarios.

4.3 Inter-participant Variability

Significant variations in movement patterns between individuals necessitate robust generalization strategies.

4.4 Removed Classes

Two dataset classes were removed from the study, since "Clapping" was just an experimental marker, which had every participant clap 3 times at the beginning and the end of the clips to ease the data processing.

5 Learning Architecture

5.1 Architecture Design

5.1.1 Convolutional Neural Network (CNN)

The CNN model consists of two convolutional blocks followed by dense classification layers. The first block uses 16 filters with a kernel size of 5 to capture broader temporal patterns, while the second block employs 32 filters with a smaller kernel size of 3 for finer detail extraction. The progressive filter increase pushes the model to learn hierarchical features, from simpler to more complex. Each convolutional layer is followed by batch normalization for training stability, max pooling for dimensionality reduction, and progressive dropout (0.4 to 0.7) for regularization. Global max pooling extracts the most significant features across the entire sequence, feeding into a compact 32-unit dense layer before the final 4-class softmax output.

5.1.2 Hybrid CNN-LSTM

We wanted to exploit the temporal information using LSTM, a first experiment used 2 stacked layers but ended up being too resource demanding to complete training. This made us go for an hybrid architecture, only using one LSTM layer to allow sequence modeling after the local pattern detection achieved in the convolution steps.

The hybrid model combines convolutional feature extraction with sequential modeling in a two-stage approach. The initial CNN stage uses 16 filters with a 3-sample kernel to identify local temporal patterns in the accelerometer data, followed by batch normalization, max pooling for dimensionality reduction, and 0.4 dropout. The extracted CNN features are then fed to an LSTM layer with 16 units that models temporal dependencies and long-term sequential patterns, incorporating both standard dropout (0.3) and recurrent dropout (0.3) to prevent overfitting in the recurrent connections. The sequence output is processed through a compact 16-unit dense layer with heavy dropout (0.7) before the final 4-class softmax classification.

Design Rationale:

- **CNN Feature Extraction**: Identify local temporal patterns
- **LSTM Sequence Modeling**: Model temporal dependencies in CNN features
- **Balanced Complexity**: Optimal trade-off between performance and efficiency

5.2 Data Balancing Techniques

5.2.1 Class Over/Undersampling

Due to the same resources constraints, we adopted a computationally efficient SMOTE variant specifically optimized for multi-sensor accelerometer time series. Unlike traditional k-nearest neighbor SMOTE which requires expensive distance calculations across flattened feature vectors, our approach uses simple random pair selection within each minority class. For each synthetic sample needed, the algorithm randomly selects two existing samples from the same activity class and performs linear interpolation with a random weight (Uniform(0,1)): synthetic sample = sample1 + × (sample2 - sample1). This preserves the temporal structure and realistic accelerometer signal characteristics while being computationally efficient with O(n) complexity versus O(n^2) for traditional k-NN SMOTE.

Our approach was a combined augmentation and undersampling, in order to minimize the generated images to match the other classes magnitude.

5.2.2 Class Weights

Class weighting addresses class imbalance during training without modifying the actual dataset. The technique assigns inverse-frequency weights to each class based on their representation in the training data. We used class weights for a comparison with SMOTE.

5.3 Overfitting Prevention

5.3.1 Regularization Strategies

- **Dropout**: Progressive rates (0.4 → 0.7) from input to output
- **Batch Normalization**: Stabilize training and reduce internal covariate shift
- **Early Stopping**: Monitor validation loss with patience=5
- **Learning Rate Reduction**: Decay factor=0.5 when validation loss plateaus

6 Results and Analysis

6.1 Overall Performance Comparison

Table 2 presents the comprehensive performance comparison across different methods:

Table 2: Overall Performance Comparison

Balancing	Architecture	Accuracy	F1-Score	Time (s)	Parameters
SMOTE	**CNN**	**0.9060**	**0.8939**	**659.97**	**12,847**
Class Weights	CNN	0.6845	0.6732	578.48	12,847
SMOTE	Hybrid	0.8023	0.7837	705.05	15,203
Class Weights	Hybrid	0.7283	0.7661	2668.30	15,203

6.2 Architecture Analysis

6.2.1 CNN Performance

- **Strengths**: Excellent at capturing local temporal patterns in accelerometer data
- **Efficiency**: Fastest training time due to parallel computation capabilities
- **Robustness**: Best generalization to unseen participants
- **Limitation**: May miss long-term temporal dependencies

6.2.2 Hybrid CNN-LSTM Performance

- **Moderate Performance**: Balanced approach between CNNs and LSTMs
- **Increased Complexity**: More parameters without proportional performance gain
- **Training Time**: 30% longer than CNN

6.3 Balancing Technique Analysis

6.3.1 SMOTE Results

- **Best Overall Performance**: 90.60% accuracy with CNN
- **Effective Minority Class Recognition**: Significant improvement in stair climbing detection
- **Computational Efficiency**: Lightweight implementation suitable for large datasets
- **Synthetic Data Quality**: Generated samples maintain realistic accelerometer patterns

7 Conclusions

This study establishes convolutional neural networks (CNNs), when combined with lightweight SMOTE balancing and participant-aware evaluation, as the most effective and practical approach for multi-sensor accelerometry-based human activity recognition (HAR). The proposed architecture achieves state-of-the-art accuracy while remaining computationally efficient and suitable for real-world deployment on constrained devices.

Three central insights emerge: (1) CNNs outperform recurrent and hybrid models by efficiently capturing local temporal patterns, generalizing robustly across unseen participants, and enabling resource-conscious deployment; (2) a streamlined SMOTE strategy effectively mitigates class imbalance, substantially improving minority activity recognition with minimal computational cost; and (3) participant-aware data splitting is indispensable, revealing overfitting that traditional random splits obscure and ensuring that reported metrics reflect realistic deployment performance.

These findings carry important implications. In healthcare, they support reliable patient monitoring, fall detection, and chronic disease management; in fitness and wellness, they enable accurate activity tracking and behavioral feedback; and in assistive technologies, they facilitate smart home integration and independent living support. At the same time, limitations such as restricted activity scope, controlled environments, and limited demographic diversity highlight the need for broader datasets, advanced sensor fusion, and cross-dataset validation.

Looking forward, future progress will depend on three fronts: (i) methodological refinement through advanced evaluation metrics, temporal consistency checks, and robustness analyses; (ii) application-driven models tailored for clinical, rehabilitation, and smart environment contexts; and (iii) technological innovation in edge optimization, incremental learning, and explainable AI to ensure both efficiency and interpretability.

In sum, this work contributes a robust methodological template for HAR research, providing both a benchmark and a roadmap. By combining architectural simplicity, methodological rigor, and practical deployment considerations, it demonstrates that effective, generalizable, and resource-efficient HAR systems are within reach, paving the way for the next generation of health, fitness, and assistive applications.

References

[1] Bao, L., & Intille, S. S. (2004). Activity recognition from user-annotated acceleration data. In *Pervasive Computing* (pp. 1-17). Springer.

[2] Chen, K., Zhang, D., Yao, L., Guo, B., Yu, Z., & Liu, Y. (2018). Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities. *ACM Computing Surveys*, 54(4), 1-40.

[3] Hammerla, N. Y., Halloran, S., & Plötz, T. (2016). Deep, convolutional, and recurrent models for human activity recognition using wearables. *Proceedings of the 25th International Joint Conference on Artificial Intelligence*.

[4] Karas, M., Urbanek, J., Crainiceanu, C., Harezlak, J., & Fadel, W. (2021). Labeled raw accelerometry data captured during walking, stair climbing and driving (version 1.0.0). *PhysioNet*. DOI: 10.13026/51h0-a262.

[5] Ravi, N., Dandekar, N., Mysore, P., & Littman, M. L. (2005). Activity recognition from accelerometer data. In *Proceedings of the 17th Conference on Innovative Applications of Artificial Intelligence* (pp. 1541-1546).

[6] Yang, J., Nguyen, M. N., San, P. P., Li, X. L., & Krishnaswamy, S. (2015). Deep convolutional neural networks on multichannel time series for human activity recognition. In *Proceedings of the 24th International Conference on Artificial Intelligence* (pp. 3995-4001).