

Deep Learning Models for Predicting Parkinson's Freeze of Gait: Accuracy Comparison of LSTM, CNN, TCN, and Transformer

Yichen Wu, Junyu Chen, Hao Su

Abstract—Freezing of gait (FOG) is a disabling Parkinson's symptom. Wearable IMUs enable proactive closed-loop cueing but demand careful choice of model. We frame FOG prediction as near-term forecasting, which predicts onsets within a short horizon from brief IMU windows. We use a public freezing-motion detection dataset (CPGDD) and compare a frequency-domain baseline with eight deep learning models: CNN, LSTM, Transformer, and hybrids under a unified preprocessing and held-out evaluation. Hybrids with CNN front-ends and lightweight temporal layers best balance impending-FOG recall and false-cue rate for closed-loop use. A representative hybrid achieves 90% accuracy and $F1 = 0.94$ for the impending-FOG class (precision = 0.91, recall = 0.97) while remaining compact (5×10^4 parameters). We discuss deployment guidance, including window/label alignment, threshold calibration to a false-cue budget, and subject-robust evaluation, and outline paths toward event-level metrics and on-device latency.

I. INTRODUCTION

Freezing of gait (FOG) is a disabling Parkinson's disease symptom linked to falls and loss of independence. Wearable IMUs enable continuous, privacy-preserving monitoring and open the door to closed-loop cueing, potentially delivering physical stimuli when a freezing of gait episode is imminent, or further usage in exoskeletons. Recent deep-learning (DL) methods learn discriminative patterns directly from raw signals [1]. However, there is limited head-to-head research for near-term freezing of gait forecasting that also addresses cueing (precision/recall and false-cue rates). We present a unified comparison of using frequency-domain as a baseline and eight DL architectures for single-sensor IMU FOG forecasting on a public dataset and using a common preprocessing pipeline in prior human-activity-recognition methodology [2], [3]. Beyond accuracy, a FOG predictor must also balance sensitivity to impending events with a low false-cue rate during normal gait. We therefore benchmark eight models to precision recall trade-offs that matter for closed-loop cueing. Recent work has further motivated this direction and the emphasis on robotic and deployable designs [4]–[6].

II. METHODOLOGY

A. Preprocessing

This study deployed the dataset from Dvorani et al., which consists of inertial measurement unit (IMU) data collected from 16 Parkinson's disease patients. The data was recorded from sensors placed on both feet, each containing a 3D accelerometer and a 3D gyroscope, sampled at a rate of 200 Hz, as shown in Figure ??.

Figure 1. A visual representation of the sensor placement and a sample of the raw time-series data can be seen

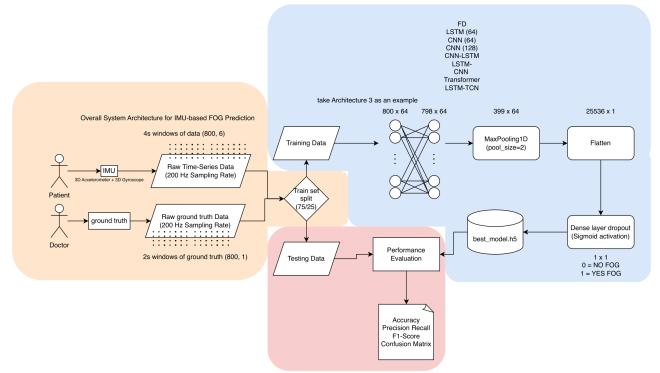


Figure 2. All data are preprocessed as an 800×7 matrix and then processed with the algorithm. The figure used Architecture 2, the CNN baseline algorithm, as an example.

The primary data preparation process, illustrated in Figure 2, was applied consistently across all deep learning experiments to ensure fair comparison. Raw data from each .csv file was loaded, with the header row being programmatically skipped to ensure consistent column value type. The six sensor channels (Acc_X/Y/Z, Gyr_X/Y/Z) were converted to a numeric format. A sliding window technique was then used to segment the continuous data into discrete samples. Each sample consisted of a 4-second input window (800 time steps) used to predict the presence of a Freezing of Gait (FOG) event within a subsequent 2-second prediction window (400 time steps). The window advanced with a step size of 0.5 seconds. A binary label of '1' (Impending FOG) was assigned to an input window if any FOG event was present in its corresponding prediction window; otherwise, a label of '0' was assigned. After generating all samples, the complete dataset was split into a training set (75%) and a testing set (25%). Finally, a standard scalar was fit exclusively on the training data and then used to normalize both the training and testing sets.

B. Architectures

The evaluated model architectures were divided into two categories: foundational baselines and advanced hybrids. The



Figure 3. 4 fundamental algorithms as baseline of the experiment

first group, depicted in Figure 3, established performance benchmarks using fundamental approaches.

Architecture 1: Frequency-Domain (FD) Algorithm Non-deep-learning baseline. Implementation from Bächlin et al. Calculates Freeze Index (FI) for each 4-second vertical accelerometer window [4]. Where $FI = \text{power}(3-8 \text{ Hz}) / \text{power}(0.5-3 \text{ Hz})$. Detect FOG if FI and total power both exceed thresholds.

Architecture 2: LSTM Baseline (LSTM(64) + Dropout + Dense) Deep learning baseline for sequential modeling. Single LSTM layer (64 units) captures temporal dependencies. Followed by Dropout for regularization and a Dense sigmoid layer for binary classification.

Architecture 3: CNN Baseline (Conv1D(64) + Conv1D(64) + Pooling + Dense) Baseline for feature extraction. Two 1D Convolutional layers (64 filters, kernel size 3) extract motifs. Max Pooling down samples feature maps, flattened, and passed to the Dense classification layer.

Architecture 4: Deeper CNN (Conv1D(128) + Conv1D(128) + Pooling + Dense) Tests hierarchical feature learning. Identical to CNN Baseline, but filters increased from 64 to 128 per Conv1D layer. Higher capacity

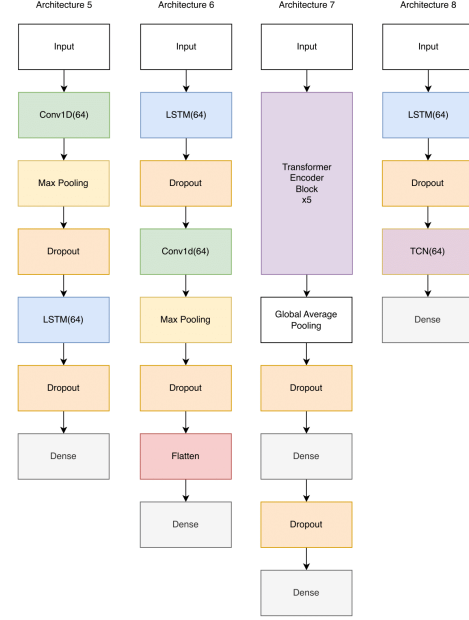


Figure 4. 4 advanced algorithms using advanced or mixed modules that gives better results

to learn complex patterns from input data.

The second group of models, shown in Figure 4, explored more complex hybrid and state-of-the-art architectures.

Architecture 5: Hybrid CNN-LSTM (Conv1D + Pooling + LSTM + Dense) Tests the synergy between convolutional and recurrent layers. Conv1D extracts features; the resulting sequence is fed to LSTM(64) for temporal modeling to make the final prediction.

Architecture 6: Hybrid LSTM-CNN (LSTM(64) + Conv1D + Pooling + Dense) Compared to Experiment 5. Inverted configuration: LSTM(64) with return_sequences processes data. Output hidden states are then analyzed by Conv1D(64) to find spatial patterns.

Architecture 7: Transformer (Multi-Head Attention + Feed Forward) Evaluates state-of-the-art self-attention. Transformer Encoder blocks use Multi-Head Attention for long-range dependencies. Output aggregated using Global Average Pooling before classification.

Architecture 8: Advanced Hybrid LSTM-TCN (LSTM(64) + TCN + Dense) Final exploratory experiment. Replaces CNN with Temporal Convolutional Network (TCN). TCN uses causal and dilated convolutions for a long receptive

field. Tests if a more powerful temporal convolution improves pattern extraction.

III. RESULTS

A. Results

A systematic evaluation of eight distinct architectures was conducted to identify the most effective model for FOG prediction. The comprehensive performance of each model on the held-out test set is detailed in Table I.

Architecture	Recall	Precision	F1-Score	Avg. Accuracy
FD	0.0588	0.9812	0.111	0.173
LSTM (64)	0.9256	0.8732	0.8986	0.8324
CNN (64)	0.942	0.9008	0.921	0.8677
CNN (128)	0.9426	0.8938	0.9176	0.8641
CNN-LSTM	0.9464	0.9142	0.93	0.8825
LSTM-CNN	0.9691	0.9093	0.9382	0.8968
Transformer	0.9616	0.9126	0.9365	0.8917
LSTM-TCN	0.9559	0.8711	0.9115	0.8498

Table I
ACCURACY COMPARISONS OF ALGORITHMS

B. Analysis

Overall analysis. In frequency-domain (FD) detection, it underperforms badly on this forecasting setup (macro F1 = 0.136; micro F1 = 0.111), which reflects a mismatch between fixed near-term onset prediction from foot-mounted IMUs. In contrast, all deep models achieve strong impending-FOG recognition, with the best models reaching F1 = 0.94 for the positive class.

Class imbalance and “false-cue” control. The held-out test split is heavily skewed toward FOG windows (81% positive). Under this skew, a model can achieve high accuracy while still triggering excessive false cues. Per-class metrics show why architecture choice matters: the LSTM model attains Impending-FOG recall = 0.93 but suffers low No-FOG recall = 0.43, which implies frequent false cues. Hybrids and the Transformer improve specificity to approximately 0.58–0.61 while maintaining 0.94 in Impending-FOG recall, meaningfully fewer false cues without sacrificing sensitivity.

Analysis of capacity and return. Scaling CNN filters from 64→128 yields minimal change (Impending-FOG F1 stays 0.92) while doubling parameters (2.56 M→5.15 M). By contrast, as shown below with compact and accurate hybrids indicate diminishing returns from wider convolution alone and a clear benefit.

- **LSTM-CNN** (56 k params): Acc = 0.90, Impending-FOG precision/recall/F1 = 0.91/0.97/0.94; No-FOG recall = 0.59.
- **Transformer** (112 k params): Acc = 0.89, 0.91/0.96/0.94; No-FOG recall = 0.58.
- **CNN→LSTM** (34 k params): Acc = 0.90, 0.91/0.97/0.94; No-FOG recall = 0.59.

Ordering matters. Swapping the order of temporal and spatial modules changes what is learned.

- **LSTM→CNN** first aligns the phase and captures the slow-varying gait context across the full 4s window. On

the other hand, the downstream Conv1D then learns local motifs over the stabilized hidden sequence, improving recall at a modest parameter cost.

- **CNN→LSTM** extracts short-range kinematic patterns first and lets the LSTM integrate them over time. It achieves a similar Impending-FOG F1 with slightly lower specificity than LSTM→CNN in our split.

Transformer performance and interpretation. The Transformer matches the top hybrids’ Impending-FOG F1 while remaining lightweight. Its multi-head attention aggregates cross-axis couplings (acc/gyro) and variable-duration precursors without hand-tuned receptive fields, which likely explains its robustness across subjects and trials. The small feed-forward head and global pooling keep memory/latency low.

Operational guidance. For closed-loop cueing, hybrids and the Transformer provide (i) 0.94 Impending-FOG F1 (ii) improved specificity (0.58–0.61) (iii) 0.12 M parameters. Therefore, these are suitable for on-device inference. Calibrating the decision threshold on per-subject thresholding to a fixed false-cue budget, event-level evaluation, and verifying end-to-end latency meets cueing deadlines.

IV. CONCLUSION

We conducted a unified, deployable comparison for IMU-based FOG forecasting. The hybrid architecture delivered the best balance for closed-loop use. Beyond model selection, our results suggest three priorities: (i) align windows and labels to forecast onsets (rather than detect ongoing episodes), (ii) calibrate thresholds to a predefined false-cue budget, (iii) account for model capacity and compute to enable on-device use. Methodologically, we recommend subject-aware evaluation (e.g., LOSO or grouped CV) and reporting event-level metrics (onset latency, event-F1) alongside window-level scores to reflect real-world usefulness. Future work will broaden daily-living conditions, explore sensor-placement differences, and quantify end-to-end online latency on target hardware.

REFERENCES

- [1] T.-H. Huang, S. Zhang, S. Yu, M. MacLean, J. Zhu, A. Di Lallo, C. Jia, T. C. Bulea, M. Zheng, and H. Su, “Modeling and stiffness-based continuous torque control of lightweight quasi-direct-drive knee exoskeletons for versatile walking assistance,” *IEEE Transactions on Robotics*, vol. 38, no. 3, pp. 1442–1459, 2022.
- [2] S. Luo, M. Jiang, S. Zhang, J. Zhu, S. Yu, I. D. Silva, T. Wang, E. Rouse, B. Zhou, H. Yuk, X. Zhou, and H. Su, “Experiment-free exoskeleton assistance via learning in simulation,” *Nature*, vol. 630, pp. 353–359, 2024.
- [3] S. Zhang, J. Zhu, T.-H. Huang, S. Yu, J. S. Huang, I. Lopez-Sanchez, T. Devine, M. Abdelhady, M. Zheng, T. C. Bulea, and H. Su, “Actuator optimization and deep learning-based control of pediatric knee exoskeleton for community-based mobility assistance,” *Mechatronics*, vol. 97, p. 103109, 2024.
- [4] M. Bächlin, M. Plotnik, D. Roggen, I. Maidan, J. M. Hausdorff, N. Giladi, and G. Tröster, “Wearable assistant for Parkinson’s disease patients with the freezing of gait symptom,” *IEEE transactions on information technology in biomedicine: a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 14, no. 2, pp. 436–446, Mar. 2010.