Skepthical review: *Wearable Step Counting: A Comparative Analysis of Deep Learning and Traditional Methods Highlighting Data Imbalance Challenges*

Summary

The paper studies step counting from wearable accelerometer data in free-living conditions, comparing a traditional vector-magnitude + band-pass + peak-detection baseline against two lightweight deep models (a compact 1D-CNN and a MobileNet-inspired separable 1D-CNN). Data from **39** participants are collected at hip and wrist, at **100 Hz** and **25 Hz**, and evaluated with subject-independent **5**-fold cross-validation (Sec. 2.1, Sec. 2.2.4). Performance is reported via event-level detection metrics (Precision/Recall/F1 with a $\pm$**150 ms** tolerance) and count-level error (MAPE; also MAE is mentioned), alongside efficiency measures (parameter counts and CPU inference time) and nonparametric statistical tests for placement/frequency/demographics (Sec. 2.3.2–2.3.3, Sec. 3.2–3.6). Empirically, the baseline performs moderately on hip data (F1 $\approx$ **0.42–0.44**; MAPE $\approx$ **10–11%**) but poorly on wrist (MAPE $>$ **50%**) (Sec. 3.2, Sec. 3.5). Both CNNs collapse to predicting zero steps across all conditions (F1 $\approx$ **0**; MAPE $\approx$ **100%**) despite low focal-loss values (Sec. 3.3). The manuscript's most valuable potential contribution is as a negative-result/diagnostic case study about sparse-event detection under extreme class imbalance; however, multiple presentation corruptions (notably Sec. 2.1.2/Table 1 and Sec. 2.1.3/Table 3), underspecified labeling and post-processing, and limited ablation/diagnostics currently prevent a definitive attribution of failure to class imbalance (vs. label alignment, target formulation, or thresholding/calibration), and weaken the paper's framing as a "comparative analysis."

Strengths

- Addresses a practically important wearable task (step counting) across meaningful deployment variables: sensor placement (hip vs wrist) and sampling rate (**100 Hz** vs **25 Hz**) (Introduction, Sec. 2.1).
- Uses subject-independent (grouped-by-participant) **5**-fold cross-validation on **39** participants, which is the right protocol to test generalization to unseen individuals (Sec. 2.2.4).
- Includes both event-level and count-level metrics, which capture different failure modes (Sec. 2.3.2, Sec. 3.2–3.3).
- Baseline peak-detection pipeline is reasonably described and serves as a relevant non-ML reference point; results clearly demonstrate strong placement effects (Sec. 2.2.1, Sec. 3.2, Sec. 3.5).
- Profiles model size and CPU inference time, which is important for on-device feasibility discussions (Sec. 2.3.2, Sec. 3.4).

- Openly reports a striking negative result (DL collapse to zero steps despite low loss) that, if rigorously diagnosed, would be instructive for the community (Sec. 3.3, Sec. 4.3–4.4).

Major issues

1. **Corrupted/irrelevant content in the dataset demographics section undermines confidence in the manuscript.** In Sec. 2.1.2, Table 1 (Participant Demographics Summary) contains unrelated text about literature search/retrieval toolkits/RAG systems and spurious citations, interleaved with partial demographic fragments. This makes it impossible to verify cohort composition and invalidates downstream demographic discussion and tests (Sec. 3.6, Sec. 4.2–4.3).

   Recommendation: Rebuild Sec. 2.1.2 and replace Table 1 with a correct demographics table for this study only ($N = 39$), including sex distribution, age summary (mean± SD and/or bins), and any recorded covariates used later (e.g., height/weight/BMI if analyzed). Remove all unrelated retrieval/RAG/toolkit content and spurious citations. Then audit Sec. 3.6 and Sec. 4.2–4.3 so every demographic count/percentage matches the corrected Table 1.

2. **Label generation / target definition is corrupted and underspecified, making the learning problem irreproducible and potentially ill-posed.** Sec. 2.1.3 is truncated and followed by a nonsensical "Table 3. Label Generation" containing repeated numeric strings. It is unclear how step annotations (timestamps) are aligned to accelerometer samples at **100 Hz** vs **25 Hz**; whether labels are single-sample impulses vs intervals; what happens if timestamps fall between samples; how overlapping **2 s** windows are labeled; and how boundary cases (steps near window edges) are treated. Given the models predict per-sample probabilities and the paper's central claim hinges on extreme sparsity/imbalance (Sec. 3.3), this missing detail is critical.

   Recommendation: Rewrite Sec. 2.1.3 as a precise, step-by-step labeling specification: (1) annotation source and format (timestamps, heel-strike definition, per-foot vs per-stride); (2) synchronization/alignment to accelerometer indices for both sampling rates; (3) exact label target used for training (impulse at one index vs pulse width $\pm N$ samples vs Gaussian bump); (4) how labels are generated inside overlapping windows and how duplicates/overlaps are resolved; (5) handling of first/last partial windows and recording boundaries. Delete the corrupted "Table 3" and replace it with a clean schematic (timeline + windowing) and/or a compact rule table.

3. **The diagnosis "deep learning fails due to extreme class imbalance" is plausible but not yet demonstrated; alternative explanations (label misalignment, overly sparse targets, calibration/thresholding/post-processing choices) are not ruled out.** The paper does not report basic imbalance statistics (positive prevalence per sample/window, steps per minute/hour, per-fold ratios), nor does it show diagnostics like probability histograms, confusion matrices, PR

curves/PR-AUC, or example probability traces aligned with ground truth (Sec. 3.3; many figures show only loss). Without these, the observed "zero steps after post-processing" could stem from an overly strict fixed threshold (0.5), peak-finding settings, or attenuation from window-stitching rather than total lack of learned signal.

Recommendation: Augment Sec. 3.3 (and methods in Sec. 2.3.1–2.3.2) with decisive diagnostics: (i) report positive class prevalence (per-sample and per-window) for each condition and per fold; (ii) add sample-level PR curves and PR-AUC (preferred under extreme imbalance) and/or ROC-AUC for raw probabilities before peak detection; (iii) show confusion matrices at representative thresholds; (iv) provide representative reconstructed probability time series for hip and wrist with ground-truth step times overlaid; (v) compare trained models to a trivial "always no-step" predictor in terms of loss and PR-AUC. If PR-AUC is near random and probabilities concentrate near zero, the imbalance-collapse claim is strengthened; if PR-AUC is non-trivial, revisit post-processing/thresholding.

4. **Deep-model post-processing appears fixed and may hard-code failure or mask partial learning.** The pipeline (Sec. 2.3.1) averages overlapping window outputs ("stitching"), then applies \texttt{find_peaks} using a fixed probability threshold (reported as **0.5**) and fixed minimum peak distance (**25** samples at **100 Hz**, **6** at **25 Hz**). Under severe imbalance, probability calibration can be conservative, so meaningful peaks may exist below **0.5**; averaging across misaligned windows can further attenuate peaks. Additionally, it is unclear whether peak parameters were tuned per fold/condition (baseline is tuned, but DL post-processing seems fixed), raising fairness concerns.

   Recommendation: In Sec. 2.3.1 and Sec. 3.3, (i) perform a threshold sweep (and optionally min-distance sweep) on the validation set within each training fold, report F1 vs threshold curves, and choose thresholds per condition in a leakage-free way; (ii) report sensitivity to stitching strategy (mean vs max vs median) and show whether averaging attenuates peaks; (iii) clearly state and justify the min-distance choice with respect to annotation definition (step vs stride) and plausible cadence. If no threshold yields non-trivial F1/PR-AUC, this supports true collapse; otherwise, update the conclusion from "no learning" to "learned signal but mis-calibrated/mis-postprocessed."

5. **Imbalance-mitigation exploration is too narrow to support broad claims about deep learning being unsuitable here.** The CNNs are trained only with one focal-loss setting ($\gamma = 2$, $\alpha = 0.25$) (Sec. 2.2.4), then declared failed. Given the likely much more extreme imbalance than typical object detection, α/γ may be inappropriate; moreover, alternative standard remedies (weighted BCE, balanced sampling, less sparse targets) are not tested.

   Recommendation: Extend Sec. 2.2.4 and Sec. 3.3 with targeted ablations (even small-scale) that directly test the paper's hypothesized failure mechanism: (1) weighted BCE with several positive-class weights; (2) focal loss sweep over α and γ; (3) class-

balanced mini-batches or oversampling windows that contain steps; (4) label smoothing in time (pulse labels $\pm N$ samples or Gaussian bumps) to reduce "single-sample impulse" sparsity; (5) optionally a simpler formulation (window-level "contains step" classification or per-window count regression) as a sanity check that the network can extract gait periodicity. Report results in a compact table per condition (hip/wrist; **100/25 Hz**).

6. **The manuscript framing as a "comparative analysis" is not supported by the current deep-learning results.** Since both CNNs are degenerate across all settings (Sec. 3.3), the paper cannot yet answer the stated trade-off question between lightweight DL and peak detection (accuracy vs efficiency), and risks over-generalizing that lightweight CNNs are inherently unfit for step counting rather than highlighting a specific pipeline/target/imbalance pitfall (Abstract, Sec. 1, Sec. 4.1–4.4).

   Recommendation: Revise the framing in Abstract/Sec. 1/Sec. 4 to match what is actually established. Choose one: (i) Recast as a negative-result diagnostic paper centered on why collapse happens and which fixes do/don't work (preferred if deep models remain weak), or (ii) demonstrate at least one non-trivial DL configuration (via the ablations above) and then present a true comparison vs the baseline including accuracy/efficiency trade-offs. In either case, bound conclusions to the tested conditions and explicitly avoid general statements about deep learning "in general" for step counting.

7. **Core reproducibility details are missing or vague across data processing, training, evaluation aggregation, and efficiency measurement, and some manuscript metadata appears placeholder.** Examples: preprocessing for DL inputs (normalization, gravity removal, axis alignment), batch size and optimizer schedule, number of windows per fold, shuffling/augmentation, initialization, regularization, and exact stitching/boundary handling are not fully specified (Sec. 2.1–2.3). Inference time reporting is ambiguous about what is included (windowing + stitching + peak detection vs forward pass only) and lacks a comparable runtime for the baseline (Sec. 3.4). The unstructured report also notes placeholder author/affiliation text, which is a serious presentation issue.

   Recommendation: Add a detailed reproducibility block in Sec. 2 (or Appendix) covering: (i) exact preprocessing for CNN inputs (normalization strategy, filtering if any, gravity handling); (ii) training hyperparameters (batch size, optimizer, LR schedule, epochs/early stopping, seeds, regularization); (iii) number of windows/samples per fold and how validation is drawn within training folds; (iv) exact stitching algorithm and handling of first/last windows; (v) metric aggregation procedure (per recording vs per subject vs per fold) (Sec. 2.3.2–2.3.3); (vi) inference-time protocol including all steps and hardware/software versions, plus baseline runtime on the same setup (Sec. 3.4). Also replace any placeholder author/affiliation text and run an end-to-end manuscript assembly audit to prevent table/section corruption.

Minor issues

1. The evaluation metric set is insufficient for heavily imbalanced detection when the model outputs probabilities. F1 after peak detection can be zero even when ranking quality is non-trivial; conversely, loss can be low even when the detector fails. The manuscript would benefit from probability-level metrics (Sec. 2.3.2, Sec. 3.3).

   Recommendation: Add PR-AUC (Average Precision) as a primary probability-level metric in Sec. 2.3.2, and report it in Sec. 3.3 for each condition (or at least hip **100 Hz/25 Hz**). If possible, also report calibration-aware plots (precision-recall curves; histograms of predicted probabilities for positive vs negative).

2. Baseline peak-detection description mixes relevant and tangential components, and tuning protocol is unclear (Sec. 2.2.1). It is not explicit which parameters are tuned, how, on what objective, and how leakage is prevented.

   Recommendation: In Sec. 2.2.1, separate implemented components used for step counting from out-of-scope elements (e.g., step-length/inertial navigation). Specify a leakage-free tuning procedure within each CV fold (grid/random search ranges; metric optimized; validation subset selection) and report chosen parameter values per condition (possibly as a small table).

3. Cross-validation fold construction is under-described (Sec. 2.1.1, Sec. 2.2.4). It is unclear whether folds are stratified by placement/sampling rate and whether every participant has all conditions.

   Recommendation: Add a concise description of fold construction: confirm group split by participant, note any stratification, and state how missing conditions are handled. Provide seeds or participant IDs per fold (or a deterministic split description) to support replication.

4. MAPE can be heavy-tailed and unstable; the wrist condition shows very large variance, and it is unclear how zero-step segments (if any) are handled (Sec. 2.3.2, Sec. 3.2–3.3).

   Recommendation: Report median and IQR for MAPE (in addition to mean±SD), and include MAE (already mentioned) consistently in the main results tables. Explicitly state how cases with $\textbf{TrueSteps} = 0$ are handled (exclude, epsilon, or alternative like sMAPE).

5. Statistical testing lacks multiple-comparison handling and effect sizes, and is partly uninformative for degenerate DL outputs (Sec. 2.3.3, Sec. 3.5–3.6).

   Recommendation: In Sec. 2.3.3 and Sec. 3.5, report effect sizes (e.g., median difference, rank-biserial correlation) and state whether p-values are corrected (Holm/Bonferroni) or explicitly uncorrected. In Sec. 3.6, either restrict inference to the baseline (where variability exists) or clearly label DL demographic tests as non-informative due to collapse.

6. Figure set is overloaded with near-duplicate training-loss plots (e.g., many figures in Sec. 3.3), while key diagnostic visuals are missing (probability traces, PR curves, confusion matrices).

   Recommendation: Move most per-fold loss histories to an appendix and keep a small representative subset in the main text. Replace freed space with the diagnostics recommended in Major Issues #3–#4 (PR curves, threshold sweeps, probability trace examples).

7. Figure readability and interpretability are uneven: axes/units/aggregation level/sample sizes are often unclear, and reliance on color alone reduces accessibility.

   Recommendation: Standardize figure formatting: label axes with units and specify aggregation level (per window/recording/subject/fold), include n per group, use color-blind-safe palettes with redundant encodings (markers/linestyles), and increase font sizes/resolution.

8. Computational efficiency discussion is incomplete for deployment: baseline runtime is missing (listed as N/A), and it is unclear whether reported CNN inference includes windowing/stitching/peak detection (Sec. 3.4). MobileNet-style depthwise convolutions being slower than CompactCNN in some cases is noted but not explained.

   Recommendation: Report baseline runtime on the same CPU, and define precisely what is timed for DL (forward pass only vs full pipeline). Average runtimes over multiple runs and briefly explain any counterintuitive speed results (e.g., framework overhead for depthwise ops). Optionally report model memory footprint (MB) and whether processing is real-time relative to **25/100 Hz**.

9. Related work positioning is relatively thin and can read as overstating novelty (Sec. 1, Sec. 2.2.2–2.2.3, Sec. 4.4).

   Recommendation: Add a short, structured Related Work subsection summarizing traditional and DL step counting/event detection on inertial data, including how prior work handles imbalance and what differs in your protocol (free-living, subject-independent CV, sparse event labels). Update claims in Sec. 1/Sec. 4.4 accordingly and ensure citations directly support statements.

10. Ethics/data provenance are not stated (Sec. 2.1.1).

    Recommendation: Add a brief statement on dataset provenance (public vs private), consent/IRB approval, anonymization, and (if possible) data/code availability to support replication.

Very minor issues

1. Section numbering/formatting is inconsistent and sometimes corrupted (e.g., mixed heading styles and numeric chains like "1.3.1.1…" in Sec. 2.1.3), reducing readability.

Recommendation: Standardize headings to a single hierarchy (**2**, **2.1**, **2.1.1**, ...) and remove corrupted numbering artifacts throughout.

2. Equation rendering issues: Vector Magnitude formula appears garbled ("p VM = x^2 + y^2 + z^2"), and some metric definitions lack edge-case conventions (e.g., F1 when Precision+Recall= **0**) (Sec. 2.2.1, Sec. 2.3.2).

   Recommendation: Fix typesetting for VM explicitly as $VM = \sqrt{x^2 + y^2 + z^2}$ (or clearly state if sqrt is intentionally omitted). State conventions for zero-denominator cases for Precision/Recall/F1 (commonly define as **0**).

3. Terminology/unit formatting is inconsistent (e.g., "25Hz" vs "25 Hz", "hipworn" vs "hip-worn"), with scattered typographical glitches and broken line breaks.

   Recommendation: Run a consistent style pass: unify SI spacing (e.g., **100 Hz**), hyphenation ("hip-worn", "wrist-worn"), and remove broken line breaks across the manuscript.

4. CNN architecture description lacks enough tensor-shape detail to verify the claim that outputs match input length; padding/stride choices are not fully specified (Sec. 2.2.2–2.2.3).

   Recommendation: For each CNN, explicitly state padding/stride for every convolution and provide a small layer-by-layer shape table for both $N = 200$ (**100 Hz**) and $N = 50$ (**25 Hz**) windows.

5. Some citations appear duplicated/mismatched (duplicate years; citations embedded in the corrupted Table 1), reducing trust in bibliographic accuracy.

   Recommendation: Audit the bibliography and in-text citations: remove duplicates, ensure years match reference entries, and eliminate any references that entered the manuscript erroneously via corrupted sections/tables.

6. Peak-distance choice for post-processing is not clearly justified with respect to cadence and annotation definition (step vs stride), especially across **100 Hz** vs **25 Hz** (Sec. 2.3.1).

   Recommendation: Add a brief justification for minimum peak distance in seconds (not only samples), tie it to plausible cadence ranges, and clarify whether labels represent steps (left+right) or strides.

7. MAPE definition uses **TrueSteps** in the denominator but the manuscript does not explicitly state that evaluated segments always have **TrueSteps** > 0 (Sec. 2.3.2).

   Recommendation: Explicitly state whether any evaluated recordings/segments have zero steps; if yes, define handling (exclude, epsilon, or use sMAPE/MAE).

Key statements and references

- ⚠ **Traditional signal processing techniques, particularly peak detection algorithms, have historically been used for step counting because they are interpretable and computationally lightweight, and they tend to perform well on well-behaved gait signals from hip-worn sensors but degrade substantially in more dynamic or noisy scenarios such as wrist-worn devices where signal patterns are more complex.**
- *Reference(s):* Abadleh et al., 2018, Chen and Pan, 2024
- *Justification:* Both papers note that classical PDR/step-counting commonly relies on threshold/peak detection (Abadleh et al., 2018; Chen and Pan, 2024) and that such methods can miscount under noise or varied motions (e.g., hand shaking/irrelevant movements; difficulty setting thresholds; degraded performance on more complex trajectories or unconstrained device placements). However, the papers do not explicitly state the historical motivation of interpretability and computational lightness, nor do they specifically compare hip-worn versus wrist-worn performance. Thus the claim is only partially supported.

- ⚠ **Recent work on deep learning for inertial sensing and step counting has shown that architectures such as attention-based LSTMs and other deep models can learn complex patterns directly from raw sensor data and potentially overcome limitations of fixed-rule algorithms, but their robustness on resource-constrained wearables and under severe class imbalance in sparse physiological event detection remains largely unexplored.**

- *Reference(s):* Khan and Abedi, 2022, Chen and Pan, 2024
- *Justification:* Supported that deep models (including attention-based LSTMs) learn patterns from raw inertial data and alleviate limitations of fixed-rule/windowed methods: Khan and Abedi, 2022 show an attention-based many-to-one LSTM that learns step patterns and outperforms threshold/peak and other approaches without per-step labels, and Chen and Pan, 2024 survey many deep inertial models replacing/augmenting classical rules. The claim that robustness on resource-constrained wearables is largely unexplored is partly consistent with Chen and Pan, 2024, who highlight efficiency/deployment as an open challenge despite initial lightweight/on-device works (e.g., TinyOdom). However, robustness under severe class imbalance in sparse physiological event detection is not addressed in either paper, so that part is unsupported.

- ✔ **The peak-detection baseline implemented in this study follows prior work by first detecting peaks in accelerometer magnitude using dynamic windows, then applying amplitude and temporal thresholds to filter valid steps, and finally estimating step length from vertical acceleration patterns between consecutive valid peaks, reflecting established model- and data-driven pedestrian inertial navigation approaches.**

- *Reference(s):* Abadleh et al., 2018, Klein, 2024, Wei, 2024
- *Justification:* Abadleh et al., 2018 describe detecting peaks from accelerometer magnitude using dynamic time windows and a thresholded Peak Vector to select valid peaks. Klein, 2024 details standard model-based PDR peak detection with a magnitude threshold and a minimum step period (temporal constraint) and reviews step-length models (e.g., Weinberg). Wei, 2024 implements a baseline that enforces amplitude and minimum-time thresholds for valid peaks and estimates step length via the Weinberg formula using vertical acceleration max–min between consecutive valid peaks. Klein, 2024 also situates these within established model- and data-driven PDR frameworks. Collectively, these directly support the stated pipeline.

- **✔ The compact 1D-CNN architecture used here is consistent with prior deep learning designs for time-series and inertial data, employing stacked 1D convolutional layers with batch normalization and ReLU activations to learn features from raw accelerometer windows, as in previous CNN-based activity and event classification systems for wearable or industrial accelerometer data.**

- *Reference(s):* Shengwei and Jianjie, 2018, Yampolsky et al., 2025, Renault et al., 2025
- *Justification:* Renault et al., 2025 explicitly describe a 1D-CNN composed of stacked 1D convolutional layers followed by batch normalization and ReLU activations, operating on windowed tri-axial accelerometer inputs for industrial event classification. Yampolsky et al., 2025 corroborate that CNN-based architectures with 1D convolutions and ReLU are standard for inertial time-series classification (HAR/SLR), reinforcing consistency with prior designs. Shengwei and Jianjie, 2018 is unrelated but does not contradict. Thus, the statement is directly supported.

- **✖ The training protocol for the deep learning models adopts practices from prior GPU-based deep learning work, including the Adam optimizer with an initial learning rate of 0.001, a ReduceLROnPlateau learning-rate scheduler, early stopping based on validation loss, and the use of Focal Loss with parameters γ=2 and α=0.25 to mitigate severe class imbalance in line with recommendations on loss functions and metrics for imbalanced deep learning tasks.**

- *Reference(s):* Müller and Kramer, 2019, Liu et al., 2021, Terven et al., 2025
- *Justification:* None of the attached papers document using Adam with $lr = 0.001$, a ReduceLROnPlateau scheduler, or early stopping. Müller and Kramer, 2019 reports training with Tversky loss and a starting learning rate of $1e{-}4$, with no mention of Adam, schedulers, or early stopping. Liu et al., 2021 focuses on GPU-accelerated optimizer evaluation and does not specify such training practices. Terven et al., 2025 discusses Focal Loss for class imbalance in general but does not prescribe $\gamma = 2$ and $\alpha = 0.25$ nor tie them to the stated training protocol. Hence the claimed protocol is not supported by the provided sources.

- ✖ **The evaluation framework and statistical analysis in this study follow established best practices for machine learning assessment, using subject-independent k-fold cross-validation to avoid subject leakage, event-level metrics such as F1-score and MAPE for performance fitness, and non-parametric tests (Wilcoxon signed-rank, Mann–Whitney U, Kruskal–Wallis) to quantify the effects of sensor location, sampling frequency, and demographics, consistent with recent guidance on cross-validation and evaluation of ML systems, especially in clinical and spatially structured domains.**
- *Reference(s):* Dehghani, 2019, Gorriz, 2024, Ferrer et al., 2024
- *Justification:* Dehghani, 2019 argues standard k-fold CV overestimates performance in HAR and recommends subject-based CV (leave-one-subject-out), reporting only F1; it does not use MAPE, non-parametric tests, or analyze effects of sensor location/sampling frequency/demographics. Gorriz, 2024 critiques k-fold CV and proposes CUBV; no Wilcoxon/Mann–Whitney/Kruskal–Wallis or MAPE. Ferrer et al., 2024 give general evaluation guidance (speaker-independence, careful splits, bootstrap CIs), not the specific protocol claimed. Thus the stated evaluation framework (subject-independent k-fold, F1+MAPE, non-parametric tests for sensor/frequency/demographics) is not supported by these papers.

Mathematical consistency audit

This section audits **symbolic/analytic** mathematical consistency (algebra, derivations, dimensional/unit checks, definition consistency).

Maths relevance: light

The paper contains a small number of mathematical definitions and conversions (windowing/overlap arithmetic, vector magnitude, peak-distance time conversions, and standard classification/counting metrics). There are no extended derivations. The main analytic gaps are missing layer-shape/padding details for the CNNs (despite claims about output length) and missing well-posedness conditions for MAPE (division by **TrueSteps**).

Checked items

1. ✔ **Window size conversion (2 s to samples)** (Sec. 2.1.3 (Data segmentation and labeling), p.3)
    - **Claim:** A **2 s** window corresponds to **200** samples at **100 Hz** and **50** samples at **25 Hz**.
    - **Checks:** algebra/arithmetic consistency, definition consistency
    - **Verdict:** PASS; confidence: high; impact: minor
    - **Assumptions/inputs:** Sampling frequency is uniform within each condition., Samples are taken at exactly **100 Hz** or **25 Hz**.
    - **Notes:** $2\ \text{s} \times 100\ \text{samples/s} = 200$; $2\ \text{s} \times 25\ \text{samples/s} = 50$.

2. ✔ **Overlap-to-stride conversion (90% overlap)** (Sec. 2.1.3 (Overlap), p.3)
    - **Claim:** With 90% overlap, the stride is 20 samples (100 Hz) and 5 samples (25 Hz).
    - **Checks:** algebra/arithmetic consistency
    - **Verdict:** PASS; confidence: high; impact: minor
    - **Assumptions/inputs:** \text{Stride} = \text{window_length} \times (1 - \text{overlap_fraction}).; Window lengths are 200 and 50 samples respectively.
    - **Notes:** 10% of 200 is 20; 10% of 50 is 5.
3. ✔ **Label-vector length consistency with windows** (Sec. 2.1.3 (Label Generation), p.3)
    - **Claim:** Each window has a corresponding binary target vector of the same length as the window.
    - **Checks:** definition consistency
    - **Verdict:** PASS; confidence: medium; impact: moderate
    - **Assumptions/inputs:** Ground-truth annotations can be represented at the same sample rate as the accelerometer data., The label is defined per time index within the window.
    - **Notes:** Conceptually consistent: a per-sample target for per-sample prediction. However, later CNN output-length claims depend on convolution padding details (checked separately).
4. ⚠ **Vector magnitude (VM) formula** (Sec. 2.2.1, bullet 1 (Vector Magnitude Calculation), p.4)
    - **Claim:** VM is computed from tri-axial acceleration as the Euclidean norm.
    - **Checks:** notation correctness, dimensional consistency
    - **Verdict:** UNCERTAIN; confidence: medium; impact: minor
    - **Assumptions/inputs:** x, y, z are acceleration components in consistent units.
    - **Notes:** The text appears garbled as “p VM = x^2 + y^2 + z^2”; it is unclear from the provided rendering whether the square root is present. The intended definition is presumably $VM = \sqrt{x^2 + y^2 + z^2}$, which is dimensionally consistent.
5. ✔ **Band-pass filter cutoff units** (Sec. 2.2.1, bullet 2 (Band-Pass Filtering), p.4)
    - **Claim:** A band-pass filter with cutoff frequencies 0.5 Hz and 3 Hz is applied to the VM signal.
    - **Checks:** units/dimensional consistency
    - **Verdict:** PASS; confidence: high; impact: minor

- **Assumptions/inputs:** VM is a time series indexed in seconds with known sampling frequency.
- **Notes:** Cutoffs specified in Hz are consistent with filtering a time series signal.

6. ⚠ **CNN output length equals input length (CompactCNN)** (Sec. 2.2.2 (Head description), p.4)
   - **Claim:** The final 1D convolution with kernel size 1 and sigmoid produces an output vector of the same length as the input window.
   - **Checks:** tensor-shape/notation consistency
   - **Verdict:** UNCERTAIN; confidence: medium; impact: moderate
   - **Assumptions/inputs:** Earlier convolution layers preserve temporal length.
   - **Notes:** With kernel size 5 in preceding convolutions, output length equals input length only if padding/stride are chosen appropriately (e.g., stride= 1 and same padding). The paper does not specify padding/stride, so the stated same-length guarantee cannot be verified symbolically.

7. ⚠ **CNN output length equals input length (MobileNet-inspired)** (Sec. 2.2.3 (Head description), p.5)
   - **Claim:** The MobileNet-inspired model also outputs a probability time series of the same length as the input window.
   - **Checks:** tensor-shape/notation consistency
   - **Verdict:** UNCERTAIN; confidence: medium; impact: moderate
   - **Assumptions/inputs:** Depthwise convolution blocks preserve temporal length.
   - **Notes:** As above, kernel size 5 depthwise convolutions typically shrink length unless padding is specified. Missing padding/stride details block verification.

8. ✔ **Peak-distance to time conversion** (Sec. 2.3.1, item 2 (Peak Detection), p.5)
   - **Claim:** Minimum distance 25 samples at 100 Hz corresponds to ~ 0.25 s; 6 samples at 25 Hz corresponds to ~ 0.24 s.
   - **Checks:** units/dimensional consistency, algebra/arithmetic consistency
   - **Verdict:** PASS; confidence: high; impact: minor
   - **Assumptions/inputs:** Time per sample is $1/fs$.
   - **Notes:** $25/100 = 0.25$ s; $6/25 = 0.24$ s.

9. ✔ **Precision/Recall/F1 formulas** (Sec. 2.3.2 (Step Detection Performance), p.6)
   - **Claim:** Precision $= \frac{TP}{TP+FP}$, Recall $= \frac{TP}{TP+FN}$, $F1 = \frac{2PR}{P+R}$.
   - **Checks:** algebra correctness, definition consistency
   - **Verdict:** PASS; confidence: high; impact: minor

- **Assumptions/inputs:** TP, FP, FN are nonnegative integers from a matching procedure.
- **Notes:** Formulas are algebraically standard and mutually consistent. Edge case $P + R = 0$ is not specified (handled separately as a clarity issue).

10. ⚠ **MAPE definition** (Sec. 2.3.2 (MAPE formula), p.6)
    - **Claim:** MAPE $= (1/n) \sum_i \left| \frac{\text{TrueSteps}_i - \text{PredictedSteps}_i}{\text{TrueSteps}_i} \right| \times 100\%$.
    - **Checks:** algebra correctness, well-posedness/domain constraints
    - **Verdict:** UNCERTAIN; confidence: medium; impact: moderate
    - **Assumptions/inputs:** n is the number of evaluated recordings/participants in the aggregation., $\textbf{TrueSteps}_i$ is nonzero (not stated explicitly).
    - **Notes:** Algebraic form matches the textual definition of absolute percentage error averaged over i. However, the denominator requires $\textbf{TrueSteps}_i \neq 0$; the paper does not state how zero-true-step cases are handled.

Limitations

- The provided paper contains very few explicit equations and no multi-step derivations; the audit is therefore limited to checking stated formulas, unit conversions, and shape/definition consistency.
- Key implementation-defining mathematical details (e.g., convolution padding/stride and exact tensor shapes) are not specified, preventing verification of some claims about output length purely from the paper text.

Numerical results audit

This section audits **numerical/empirical** consistency: reported metrics, experimental design, baseline comparisons, statistical evidence, leakage risks, and reproducibility.

18 numeric checks were run: 14 PASS, 2 FAIL (both only floating-point precision on overlap/stride), and 2 UNCERTAIN (claims requiring per-condition Table 3 parameter entries not included in the provided inputs). No substantive arithmetic/unit inconsistencies were detected within the verified items.

Checked items

1. ✔ **C1_table1_sex_total** (Page 3, Table 1 (Participant Demographics Summary, $N = 39$))
    - **Claim:** Sex (Female/Male) is reported as 21/18 with $N = 39$.
    - **Checks:** parts_to_total
    - **Verdict:** PASS
    - **Notes:** female + male == N_{total}
2. ✔ **C2_table1_age_ranges_total** (Page 3, Table 1 (Participant Demographics Summary, $N = 39$))

- **Claim:** Age Range counts are reported as $(18\text{-}25) = 15$, $(26\text{-}40) = 14$, $(41+) = 10$ with $N = 39$.
- **Checks:** parts_to_total
- **Verdict:** PASS
- **Notes:** age_18_25 + age_26_40 + age_41_plus == N_{total}

3. ✔ **C3_window_size_100hz** (Page 3, Section 2.1.3 (Data segmentation and labeling))
    - **Claim:** A **2 s** window corresponds to **200** samples at **100 Hz**.
    - **Checks:** unit_consistency_recompute
    - **Verdict:** PASS
    - **Notes:** window_duration × sampling_rate == reported_samples
4. ✔ **C4_window_size_25hz** (Page 3, Section 2.1.3 (Data segmentation and labeling))
    - **Claim:** A **2 s** window corresponds to **50** samples at **25 Hz**.
    - **Checks:** unit_consistency_recompute
    - **Verdict:** PASS
    - **Notes:** window_duration × sampling_rate == reported_samples
5. ✖ **C5_overlap_stride_100hz** (Page 3, Section 2.1.3 (Overlap))
    - **Claim:** With **90%** overlap and a **200**-sample window (**100 Hz**), stride is **20** samples.
    - **Checks:** percentage_to_quantity
    - **Verdict:** FAIL
    - **Notes:** reported_stride == window_samples $\times(\mathbf{1 - overlap})$
6. ✖ **C6_overlap_stride_25hz** (Page 3, Section 2.1.3 (Overlap))
    - **Claim:** With **90%** overlap and a **50**-sample window (**25 Hz**), stride is **5** samples.
    - **Checks:** percentage_to_quantity
    - **Verdict:** FAIL
    - **Notes:** reported_stride == window_samples $\times(\mathbf{1 - overlap})$
7. ✔ **C7_cv_split_counts** (Page 3, Section 2.1.4 (Data splitting))
    - **Claim:** In **5**-fold CV with **39** participants, approximately **31** are used for training and approximately **8** for testing each fold.
    - **Checks:** integer_partition_feasibility
    - **Verdict:** PASS
    - **Notes:** Sum check diff= **0.0**; test-size closeness $|8.0 - 7.8| = 0.20000000000000018$.

8. ✔ **C8_peak_distance_100hz_seconds** (Page 5, Section 2.3.1 (Peak Detection post-processing))
    - **Claim:** Minimum distance **25** samples for **100 Hz** corresponds to approximately **0.25 s** and a maximum of **4 steps/second**.
    - **Checks:** unit_conversion
    - **Verdict:** PASS
    - **Notes:** Time diff_abs= **0.0**, Steps/s diff_abs= **0.0**.
9. ✔ **C9_peak_distance_25hz_seconds** (Page 5, Section 2.3.1 (Peak Detection post-processing))
    - **Claim:** Minimum distance **6** samples for **25 Hz** represents approximately **0.24 s**.
    - **Checks:** unit_conversion
    - **Verdict:** PASS
    - **Notes:** min_distance_samples/sampling_rate ≈ reported_seconds
10. ✔ **C10_matching_tolerance_vs_sampling** (Page 6, Section 2.3.2 (Performance metrics))
    - **Claim:** Predicted steps were matched to ground-truth within **±150 ms**. Compare this tolerance to sampling periods at **100 Hz** and **25 Hz**.
    - **Checks:** unit_consistency_recompute
    - **Verdict:** PASS
    - **Notes:** Computed implied sample tolerance from **±150 ms**; no explicit equality claim to validate. Implied tolerances: **15** samples (**100 Hz**) and **3.75** samples (**25 Hz**).
11. ✔ **C11_table2_equal_means_hip** (Page 3, Table 2 (Data Recording and Annotation Summary))
    - **Claim:** Hip **100 Hz** and Hip **25 Hz** have identical mean±SD: duration **55.4 ± 8.1 min** and steps **2810 ± 954**.
    - **Checks:** repeated_constants_match
    - **Verdict:** PASS
    - **Notes:** All corresponding Hip_100Hz vs Hip_25Hz summary stats match exactly (mean and SD).
12. ✔ **C12_table2_equal_means_wrist** (Page 3, Table 2 (Data Recording and Annotation Summary))
    - **Claim:** Wrist **100 Hz** and Wrist **25 Hz** have identical mean±SD: duration **54.9 ± 8.9 min** and steps **2785 ± 961**.
    - **Checks:** repeated_constants_match
    - **Verdict:** PASS

- **Notes:** All corresponding Wrist_100Hz vs Wrist_25Hz summary stats match exactly (mean and SD).

13. ✔ **C13_table3_baseline_mape_matches_text_hip** (Page 7 (text in Section 3.2) and Page 8, Table 3)
    - **Claim:** Text says baseline hip MAPE is approximately **10–11%**; Table 3 reports Hip_100Hz **10.41%** and Hip_25Hz **11.45%**.
    - **Checks:** range_check_vs_reported_values
    - **Verdict:** PASS
    - **Notes:** Checked table means against textual range with abs slack.
14. ✔ **C14_table3_baseline_wrist_mape_over_50** (Page 7 (text in Section 3.2) and Page 8, Table 3)
    - **Claim:** Text says baseline wrist MAPE surged to over **50%** for Wrist_100Hz (**57.62%**) and Wrist_25Hz (**54.69%**).
    - **Checks:** threshold_check
    - **Verdict:** PASS
    - **Notes:** Margins above threshold: [**7.619999999999997, 4.689999999999998**]
15. ✔ **C15_table3_deeplearning_mape_near_100** (Page 7-8 (text in Section 3.3) and Page 8, Table 3)
    - **Claim:** Text claims deep learning MAPE is consistently **99.98–100.00%** (or very close). Table 3 shows CompactCNN Hip_100Hz $\mathbf{99.98 \pm 0.11}$ and many $\mathbf{100.00 \pm 0.00}$ entries.
    - **Checks:** range_check_vs_reported_values
    - **Verdict:** PASS
    - **Notes:** Inclusive range check for provided deep learning MAPE means.
16. ⚠ **C16_table3_parameters_consistent_across_conditions** (Page 8, Table 3)
    - **Claim:** CompactCNN parameters are **52,705** in all conditions; MobileNetCNN parameters are **12,252** in all conditions.
    - **Checks:** repeated_constants_match
    - **Verdict:** UNCERTAIN
    - **Notes:** Cannot verify 'in all conditions' because per-condition parameter entries are not provided in PAYLOAD.
17. ⚠ **C17_table3_baseline_parameters_zero** (Page 8, Table 3)
    - **Claim:** Baseline parameter count is listed as **0** for all conditions.
    - **Checks:** repeated_constants_match
    - **Verdict:** UNCERTAIN
    - **Notes:** Cannot verify 'for all conditions' because per-condition baseline parameter entries are not provided in PAYLOAD.

18. ✔ **C18_inference_time_ratio_compact_hip_100_vs_25** (Page 11 (text in Section 3.4.2) and Page 8, Table 3)
    - **Claim:** CompactCNN inference time: **37.57 s** (Hip_100Hz) vs **7.00 s** (Hip_25Hz).
    - **Checks:** ratio_computation
    - **Verdict:** PASS
    - **Notes:** Computed ratio $37.57/7.00 = 5.367142857...$, compared to expected $\sim 4\times$ with wide relative tolerance.

Limitations

- Only parsed text from the PDF was available; no access to underlying datasets, per-fold outputs, or code used to generate tables/statistics.
- Values embedded in plots (Figures 1–21) cannot be numerically extracted without reading plot pixels, which is out of scope per instructions.
- Several checks can only assess internal arithmetic/units/repetition consistency, not the scientific correctness of reported results.
- Some 'in all conditions' claims about Table 3 parameter counts could not be verified because per-condition parameter entries were not provided in the supplied inputs (yielding UNCERTAIN verdicts for those checks).