Beyond Simulated Benchmarks: Evaluating Motion Representations for Fall Detection Under Real-World Data Scarcity

Timilehin B. Aderinola[1(✉)], Ilaria D'Ascanio[3], Luca Palmerini[3], Lorenzo Chiari[3], Jochen Klenk[4], Clemens Becker[4,5], Brian Caulfield[2], and Georgiana Ifrim[1]

[1] School of Computer Science, University College Dublin (UCD), Ireland
[2] UCD School of Public Health, Physiotherapy and Sports Science, Ireland
{timilehin.aderinola,b.caulfield,georgiana.ifrim}@ucd.ie
[3] Department of Electrical, Electronic and Information Engineering "Guglielmo Marconi", University of Bologna, Italy
{luca.palmerini,ilaria.dascanio2,lorenzo.chiari}@unibo.it
[4] Department of Clinical Gerontology, Robert Bosch Hospital, Stuttgart, Germany
{jochen.klenk,clemens.becker}@rbk.de
[5] Digital Geriatrics Unit, Heidelberg University Hospital, Heidelberg, Germany

Abstract. Falls are a major health concern for older adults, and wearable sensors have been widely explored for detecting falls and enabling timely intervention. However, real-world falls are extremely rare: collecting 100 of them requires an estimated 100,000 days of monitoring, resulting in severely limited labelled data for training machine learning models. Consequently, many approaches rely on simulated datasets, often reporting high laboratory performance but limited real-world generalisation. We present a systematic evaluation of motion representations for wearable fall detection under real-world data scarcity. Using accelerometer signals, we compare interval-based, kernel-based, symbolic, and foundation model representations. As an interpretable baseline, we additionally investigate a lightweight symbolic representation that converts short motion segments into symbolic sentences augmented with physically-grounded impact descriptors. Experiments use FallAllD, a simulated falls dataset, and FARSEEING, a clinically verified real-world falls dataset. Through cross-validation, controlled data scarcity, and cross-dataset transfer, we examine how representation choices affect robustness under realistic deployment. Our results reveal that highly parameterised kernel and foundation models excel on simulated data but degrade severely under both data scarcity and domain shift. Although the interval-based representation achieves the strongest absolute real-world performance, augmenting a symbolic representation with physically-grounded impact descriptors yields the smallest degradation under domain shift and retains detection sensitivity under extreme scarcity, albeit at lower precision. These findings highlight the importance of evaluating beyond simulated benchmarks and show that representation choice is critical for deployable fall detection given the scarcity of real-world data.

1 Introduction

As the proportion of older adults increases worldwide, so does the demand for technologies supporting independent living and remote health monitoring. Among the health risks facing this population, falls are among the most significant, threatening physical safety, long-term independence, and psychological well-being [24]. Falls are the second leading cause of unintentional injury-related death globally, with an estimated 684,000 fatalities [27] and over 37 million cases requiring medical attention annually [4]. A particular concern is not only the event itself but the "long lie": the period during which a person remains on the ground without assistance, which is associated with greater risk of complications and mortality when help is delayed. Fall detection systems have therefore become an important component of geriatric care and remote monitoring [19]. Such systems typically acquire movement during everyday activity, capturing both activities of daily living (ADLs) and falls, followed by signal preprocessing, feature extraction, and classification [14].

Many existing fall detection systems emphasise achieving high accuracy using threshold-based heuristics or supervised machine learning models [19]. In controlled laboratory settings, these approaches often report accuracies exceeding 95%, which can create the impression that fall detection technology is mature and ready for large-scale deployment [17]. However, when evaluated outside controlled laboratory settings, a substantial performance gap often emerges [2]. A key reason for this discrepancy is the extreme scarcity of real-world fall data. Because falls are accidental events, collecting sufficient real-world examples is inherently challenging. It has been estimated that capturing 100 real-world falls may require approximately 100,000 days (around 300 person-years) of monitored activity [11]. As a result, much of existing research relies on simulated falls performed by young, healthy participants in controlled laboratory environments. While simulated datasets are useful for initial model development, they often fail to reflect the complexity of falls occurring naturally in daily life, leaving a gap between laboratory performance and real-world deployment.

In many fall detection studies, improvements are primarily pursued through increasingly complex model architectures. However, the scarcity of real-world fall data fundamentally limits the ability of such models to learn robust patterns, and high in-domain accuracy offers no guarantee of robustness when models are transferred to real-world conditions. As a result, the choice of motion representation becomes a critical factor: different representations may capture fall dynamics more effectively when only a small number of fall events are available, and may differ in how well they generalise across datasets. Despite this, relatively little work has systematically examined how representation choices influence robustness under data scarcity or their ability to generalise from simulated or private datasets to real-world scenarios.

In this work, using a realistic streaming evaluation pipeline, we systematically evaluate motion representations for wearable fall detection. We compare interval-based, kernel-based, symbolic, and foundation-model representations under realistic constraints of limited real-world fall data and cross-dataset transfer

between simulated and real-world datasets, with particular attention to how representations behave when moving from simulated to real-world falls. **The main contributions of this work are as follows:**

1. We present a systematic comparison of interval-based, kernel-based, symbolic, and foundation-model representations for wearable fall detection, using a unified streaming evaluation pipeline with subject-wise splits.
2. We characterise representation robustness under extreme data scarcity and quantify the simulation-to-reality gap through cross-dataset transfer from simulated falls to clinically verified real-world falls.
3. As an interpretable case study, we implement FallLM, a lightweight symbolic representation augmenting SAX tokens with impact descriptors, and use its transparency to diagnose failure modes at the token level.
4. We release our code[6] to support reproducible benchmarking of wearable fall detection and related time series event detection tasks.

Our experiments use accelerometer data from the FARSEEING real-world falls dataset [11] and the publicly available FallAllD dataset [22], which contains simulated falls performed in controlled laboratory settings. To ensure realistic evaluation, we adopt a streaming event-detection protocol with subject-wise splits. During training, samples are generated using fixed-size overlapping windows extracted from the continuous signals. During testing, signals are processed in a continuous streaming setting, beginning at arbitrary points independent of the fall impact event. For each incoming window, the trained model estimates the probability of a fall, which is then thresholded to produce fall predictions.

The rest of this paper is organised as follows. Section 2 reviews related work on fall detection and time series representation. Section 3 describes the datasets, preprocessing steps, and representation methods evaluated in this study. Section 4 presents the experimental setup and evaluation protocols. Section 5 reports the results and analysis, which are discussed in Section 6. Finally, Section 7 concludes the paper and outlines directions for future work.

2 Related Work

2.1 Wearable Sensor-Based Fall Detection

Wearable and ambient sensing technologies have both been explored for automatic fall detection. Ambient devices, such as cameras [3] or environmental sensors [26], can provide rich contextual information but are limited to fixed monitoring spaces and may raise privacy concerns in sensitive environments. Consequently, wearable inertial measurement units (IMUs), particularly accelerometers, have been widely adopted due to their low cost, portability, and ability to continuously monitor individuals in free-living environments [16,29].

Wearable fall detection approaches are typically categorised as threshold-based, machine learning (ML)-based, or hybrid methods [21]. Threshold-based techniques

[6] https://github.com/mlgig/fall-lm

[12] detect falls using predefined rules based on signals such as acceleration magnitude or orientation changes. Although computationally efficient, these methods are limited by high false alarm rates because many activities of daily living (ADLs) exhibit motion patterns similar to falls. Hence, many studies have explored ML-based approaches that involve manual feature extraction from raw motion signals and fall detection with classifiers such as decision trees, support vector machines, and neural networks. More recently, deep learning models [13,28] have also been applied to automatically learn representations of motion signals. In hybrid systems [20], thresholds are often used to pre-filter candidate fall events before applying a learned classifier for final decision making. While a wide range of machine learning models have been proposed for fall detection, relatively little work has systematically examined how different time series representations influence robustness under data scarcity or cross-dataset transfer.

2.2 Time Series Representations for Wearable Motion Data

A critical step for fall detection systems is to transform raw sensor signals into compact and discriminative feature spaces that facilitate classification while remaining robust to noise, data leakage and subject variability. Kernel-based approaches have recently gained popularity for time series classification due to their strong performance and computational efficiency. Methods from the ROCKET family [5] transform time series using a large number of convolutional kernels. Variants such as MiniRocket [6] and MultiRocket [25] improve efficiency through deterministic kernel selection and optimised pooling operations, while maintaining competitive accuracy on large time series benchmarks.

Interval-based representations summarise signals using statistical features extracted from temporal subsequences. For example, QUANT [7] represents time series through quantiles computed over hierarchical intervals, capturing distributional characteristics of motion signals at different temporal locations.

Symbolic approaches transform real-valued signals into sequences of discrete tokens using segmentation and discretization techniques such as Symbolic Aggregate Approximation (SAX). Methods such as WEASEL [23] and MrSQM [18] represent time series as collections of symbolic patterns.

More recently, foundation models for time series have been proposed to learn generalisable representations through large-scale pre-training. Transformer-based models such as Mantis [8] use contrastive learning objectives to produce embeddings that can be adapted to downstream tasks such as time series classification.

Despite the diversity of these representation paradigms, many highly accurate methods produce opaque feature spaces that lack clinical interpretability. Furthermore, relatively little work has systematically examined how these representations influence robustness under extreme data scarcity or their ability to generalise from simulated to real-world fall datasets. In this study, we evaluate representative methods from these families, including MiniRocket (kernel-based), QUANT (interval-based), WEASEL (symbolic), MrSQM (symbolic), and the foundation model Mantis. To complement these opaque representations, we additionally include FallLM, a lightweight symbolic representation that encodes motion as

sequences of discrete, human-readable tokens augmented with impact descriptors, allowing its decisions to be inspected directly at the token level.

3 Materials and Methods

3.1 Datasets

FARSEEING [11] is a large collection of clinically verified real-world falls recorded using wearable inertial sensors. It contains 208 falls from 92 participants (mean age 76.1 ± 12.6 years). Each recording spans 20 minutes of continuous sensor data with the fall near the midpoint. We use the subset of 150 falls from 54 participants, where the sensors were placed at the fifth lumbar (L5) position and recorded at 100 Hz. The L5 placement provides a stable representation of whole-body motion while reducing noise introduced by limb movements.

FallAllD [22] is a publicly available collection of simulated falls and ADLs performed in controlled settings: 35 fall types and 44 ADLs from 15 participants, recorded at 238 Hz (we resample to 100Hz to match FARSEEING). We use 465 simulated falls from 14 participants with waist-mounted sensors, the placement most comparable to FARSEEING's L5.

3.2 Data Preprocessing

Signal Aggregation. The tri-axial acceleration signals are converted into a single acceleration magnitude signal, computed as $M = \sqrt{Acc_x^2 + Acc_y^2 + Acc_z^2}$, where Acc_x, Acc_y, and Acc_z are the anterior–posterior, medial–lateral, and vertical components. Using magnitude reduces sensitivity to sensor orientation and enables consistent processing across devices and placements [9].

Training Data Segmentation. Fall samples are extracted using three-second windows beginning one second before the annotated impact, comprising a one-second pre-impact phase, the impact, and a one-second post-impact phase. This compact window captures the essential dynamics of the fall while remaining suitable for real-time detection. ADL (negative) samples are extracted from fall-free portions using one-second-step sliding windows, retaining only windows whose maximum magnitude exceeds 1.4 g; this removes low-intensity movement while preserving dynamic activities such as walking or turning [20].

3.3 Streaming Fall Detection Evaluation

Since FARSEEING only provides binary impact labels for each timepoint, we pose fall detection as binary classification (fall vs. ADL). To simulate real-world deployment, fall detection is evaluated in a streaming setting where the continuous accelerometer signal is processed sequentially without prior knowledge of fall events, following a recently proposed streaming evaluation protocol [1]. The signal is analysed using sliding windows with a one-second step size. For each

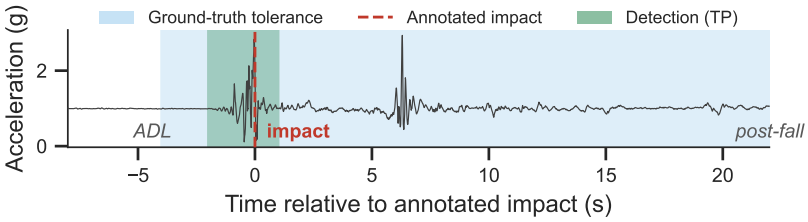

Fig. 1. A 30-second FARSEEING fall segment showing pre-fall activity, the annotated impact (dashed), and the post-fall lying period. Here, the detection window overlaps the ground-truth tolerance interval (shaded), counting as a true positive. This example is illustrative and not indicative of overall detection precision.

Table 1. Representation methods evaluated in this study.

Method	Type	Representation
WEASEL 2.0 [23]	Symbolic	Symbolic Fourier Approximation (SFA)
MrSQM [18]	Symbolic	Symbolic Aggregate Approximation (SAX)
FallLM	Symbolic	SAX + impact-level tokens
MiniRocket [6]	Kernel-based	Random convolutional kernels
QUANT [7]	Interval-based	Quantile features
Mantis [8]	Foundation model	Transformer embeddings

window, the trained model estimates the probability of a fall event. To account for the temporal context of fall events, detection is evaluated within a tolerance interval surrounding the annotated impact point. This interval includes both the motion leading to the fall and the immediate recovery period following impact. A detection is considered correct if the predicted fall window overlaps with the ground-truth interval (see Fig. 1). Overlap between predicted windows and ground truth is measured using the Intersection over Union, defined as $IoU(d, R) = |d \cap R|/|d \cup R|$, where d denotes the detected window and R the ground-truth interval. A detection with $IoU(d, R) > 0$ is counted as a true positive, while detections without overlap are considered false positives. If no detection overlaps with the fall interval, the event is counted as a false negative. To avoid registering repeated firings of the sliding window as separate detections, a debounce step retains only the first alarm within a detection episode.

3.4 Motion Representations

Each evaluated representation (Table 1) transforms the acceleration magnitude into a feature space that a classifier then uses. We briefly describe each below.

Symbolic Representation. Symbolic representations transform continuous signals into sequences of discrete symbols drawn from a finite alphabet. This discretization process reduces sensitivity to noise and allows temporal patterns to be represented as symbolic words that can be analysed using techniques similar

to those used in text processing. In this work, we evaluate three symbolic approaches that employ different symbolic transformations and modelling strategies: WEASEL, MrSQM, and FallLM.

WEASEL 2.0 [23], based on the Symbolic Fourier Approximation (SFA), applies a random selection of dilated windows, extracting SFA words at multiple scales and dilations to form a high-dimensional dictionary. The resulting feature vector is classified with a ridge regression classifier.

MrSQM [18] discretises segments of the time series into symbolic tokens (SAX or SFA) and constructs features from symbolic subsequences extracted across multiple resolutions, which are then classified using logistic regression.

FallLM. We implement *FallLM*, a lightweight symbolic representation for motion signals (see Fig. 2). The acceleration magnitude signal is first segmented into short intervals using Piecewise Aggregate Approximation (PAA), after which the aggregated values are discretised into symbolic tokens using SAX. Each window is therefore represented as a short sequence of symbols capturing coarse motion dynamics. SAX discretization uses Gaussian breakpoints derived from the standard normal distribution, yielding a standardised symbolic alphabet.

In addition to the SAX token sequence, we augment each motion sentence with two impact-related tokens derived from the peak acceleration magnitude within the window. The first token encodes the absolute impact level by discretising the peak acceleration into three levels informed by the biomechanical ranges reported in [10]: low ($< 1.8g$, typical for normal walking), medium ($1.8g$–$2.5g$, associated with deliberate vertical displacement such as climbing stairs), and high ($> 2.5g$, associated with hard impacts or falls). Because these thresholds are defined in units of gravitational acceleration (g), they remain consistent across datasets and sensor configurations. We show that this coarse, dataset-invariant encoding transfers better than an absent or finer-grained token (see Appendix A.1).

To capture contextual information about the impact relative to surrounding motion, we also include a relative impact token that reflects the peak acceleration relative to the magnitude distribution within the window. This token provides a coarse indication of how abrupt the impact is compared to the surrounding movement dynamics. Together, the absolute and relative impact tokens complement the SAX representation by incorporating information about motion intensity while preserving the symbolic structure of the motion sequence. The resulting symbolic sequence is converted into an n-gram bag-of-words representation using a TF-IDF (Term Frequency-Inverse Document Frequency) vectoriser, and a logistic regression classifier is trained on these features for fall detection.

Kernel-Based Representation. MiniRocket [6] transforms the input signal using a large number of fixed convolutional kernels and extracts simple statistics from the resulting feature maps, such as the proportion of positive values. These features form a high-dimensional representation that can be efficiently classified using a linear model (Ridge Regression Classifier).

Interval-Based Representation. QUANT [7] extracts quantile statistics from hierarchical intervals of the time series, capturing distributional properties of the

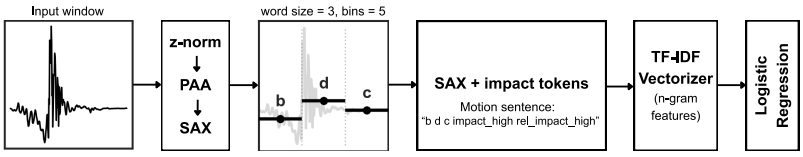

Fig. 2. Overview of the FallLM pipeline.

signal at different temporal locations. These interval features are then used by an ensemble classifier (ExtraTrees Classifier).

Foundation Model Representation. Mantis [8] is a transformer-based model pre-trained using contrastive learning to capture general temporal patterns across datasets. In our experiments, the pre-trained model is finetuned to generate embeddings for fall detection.

4 Experiments

4.1 Experimental Setup

All experiments used Python 3.10 on an Apple M4 Pro (24 GB). Implementations of WEASEL 2.0, MrSQM, MiniRocket, and QUANT were obtained from `aeon` v0.8.1 [15], each evaluated as an integrated pipeline with its canonical classifier and default parameters. To address the extreme class imbalance between falls and ADLs, we apply random oversampling of the minority (fall) class within each fold's training split, balancing the classes before model fitting; no classifier-level class weighting is used. For MrSQM, we enable 5 SAX representations and disable SFA, so it serves as a purely SAX-based comparator, directly comparable to FallLM's SAX basis. For Mantis, we implement a scikit-learn-style wrapper around the pretrained model and fine-tune it using PyTorch 2.7.1 with Apple Metal Performance Shaders (MPS) acceleration. Training employs early stopping based on validation loss with a patience of 10 epochs.

For FallLM, the acceleration magnitude signal is discretised using SAX with an alphabet size of 5 and a word size of 3, fixed a priori from the temporal structure of fall events rather than tuned. A word size of 3 maps the 3-second window to the three biomechanical phases of a fall—pre-impact, impact, and post-impact (1 second per symbol)—and keeps motion sentences short enough to remain human-readable for the interpretability analysis (Section 5.4). Each SAX symbol denotes an amplitude level from `a` (lowest) to `e` (highest). The default configuration for FallLM, together with sensitivity analyses on the impact tokens and alphabet size, is provided in Appendix A. After appending the two impact tokens, each motion sentence comprises five tokens, vectorised as n-grams (range $(4, 5)$) and classified by logistic regression; this n-gram range forces the classifier to weigh the structural shape of the motion together with its impact magnitude.

We conduct three experiments: (1) subject-wise cross-validation on FARSEEING (Section 5.1), (2) a data-scarcity analysis that progressively reduces the number of training falls (Section 5.2), and (3) a cross-dataset transfer experiment from simulated (FallAllD) to real-world (FARSEEING) falls (Section 5.3).

4.2 Training and Evaluation

Training. Each three-second window (300 samples) is labelled with a binary target indicating the presence or absence of a fall event.

Testing. The test set consists of unsegmented signals, each containing a single fall annotated with a ground-truth impact index f. We define a tolerance interval of $[f-3, f+20)$ seconds around the impact. The 3-second pre-impact allowance captures the falling phase, and the post-impact interval covers the period during which the person may remain on the ground. A detection window overlapping this interval is counted as a true positive (see Section 3.3 for further details).

Evaluation Metrics. Given the extreme class imbalance of streaming detection, we report Precision, Recall, and F_1 Score rather than accuracy. We additionally report Detection Delay (in seconds) and inference runtime per window as a measure of computational efficiency. Detection Delay is the signed time difference between the detection and the impact index; negative values indicate detection prior to the annotated impact, and smaller absolute values indicate more temporally precise detection.

5 Results

5.1 Cross-validation on Real-World Falls

Table 2 presents the five-fold subject-wise cross-validation results on the FARSEEING dataset (an average of 31 falls per fold). The foundation model, Mantis, obtains the highest F_1 score (0.83), with Quant, WEASEL, and MiniRocket close behind. Mantis achieves the highest recall (0.89) and Quant the highest precision (0.84), while WEASEL and MiniRocket maintain a balance between the two. These methods cluster tightly, indicating that with enough labelled real-world falls, interval-based, kernel-based, foundation model, and SFA-based symbolic representations all reach comparable in-domain performance.

The two SAX-based symbolic methods, FallLM and MrSQM, achieve lower F_1 scores (0.64 and 0.66). FallLM attains high recall (0.87), but the lowest precision (0.52). In a streaming setting with many ADL windows, this manifests as frequent false alarms, which we examine in Section 5.4. FallLM is the most computationally efficient method (0.01 ms per sample), though all evaluated methods are already fast enough for real-time and always-on deployment.

Overall, these results show that several representation families achieve strong real-world performance when sufficient labelled data is available, and that the highest in-domain F_1 does not single out one representational paradigm.

Table 2. Five-fold subject-wise cross-validation results on the FARSEEING dataset. Values are reported as mean(std) across folds.

Model	Runtime (ms)	Precision	Recall	F1	Delay (s)
Mantis	0.84(0.33)	0.78(0.05)	**0.89**(0.07)	**0.83**(0.02)	-1.28(0.22)
Quant	0.27(0.06)	**0.84**(0.08)	0.80(0.01)	0.82(0.04)	-0.88(0.15)
WEASEL	4.77(2.94)	0.82(0.08)	0.80(0.05)	0.81(0.03)	-1.23(0.25)
MiniRocket	0.27(0.18)	0.80(0.06)	0.82(0.08)	0.80(0.03)	-1.12(0.22)
MrSQM	3.90(2.49)	0.64(0.09)	0.68(0.04)	0.66(0.05)	**-0.51**(0.25)
FallLM	**0.01**(0.00)	0.52(0.11)	0.87(0.10)	0.64(0.09)	-2.31(0.20)

5.2 Data Scarcity Analysis

To examine behaviour under limited real-world data, we train each method on fractions (1%, 5%, 10%, 25%, 50%, 100%) of the 112 fall events in the training set while retaining all negative samples, averaging five runs over randomly sampled fall subsets per fraction. We run this on FARSEEING only, as simulated falls are not scarce. Figure 3 presents the resulting trends.

With only 1% of the available fall events (two falls), FallLM is the only method to achieve substantial detection (mean $F_1 \approx 0.35$), recovering falls with minimal supervision but at low and highly variable precision, depending on whether the two sampled falls are representative. Conversely, Mantis fires rarely but almost always correctly (near-perfect precision, low recall, mean $F_1 \approx 0.11$). This conservative behaviour may reflect the priors it inherits from pretraining. QUANT and MiniRocket detect only isolated falls, while MrSQM and WEASEL fail entirely ($F_1 = 0$). Notably, the methods that perform best with abundant data collapse under extreme scarcity, underscoring that strong in-domain performance does not imply robustness to scarcity.

As more fall events become available, all methods improve rapidly, reaching near-peak performance by roughly 25–50% of available falls; several show slight precision declines thereafter, likely reflecting the heterogeneity of real falls and their similarity to fall-like ADLs. FallLM, however, plateaus near $F_1 = 0.64$: its recall stays high, but its precision does not improve with additional data, so other methods overtake it on F_1 once they accumulate enough falls. Among the symbolic methods, MrSQM does not detect falls until 25% of falls are available, suggesting that FallLM's scarcity-detection capability stems from its impact tokens rather than from symbolic representation in general.

These results show that the choice of motion representation is decisive under data scarcity. Representations encoding strong, dataset-invariant priors can be particularly valuable when labelled falls are scarce. More broadly, effective fall detection may not require large collections of real-world falls, provided the representation is matched to the scarcity regime.

5.3 Cross-Dataset Transfer

To assess robustness under domain shift, we train each method on the simulated FallAllD dataset and evaluate it on real-world FARSEEING falls without re-

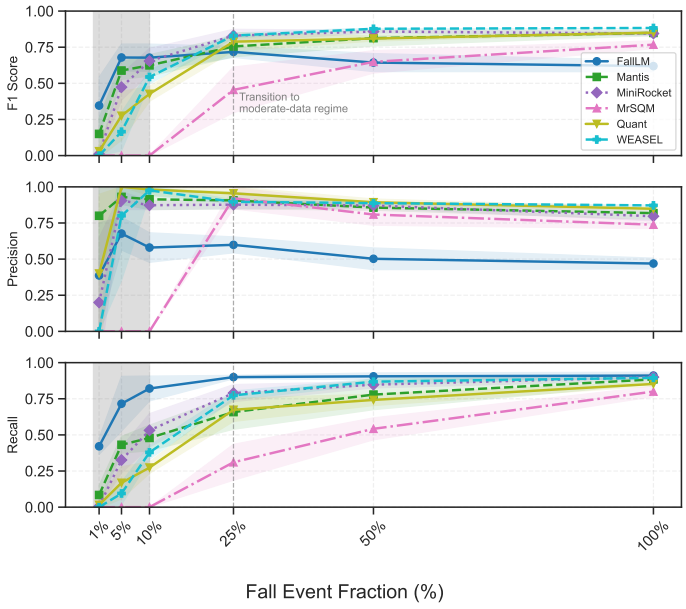

Fig. 3. Performance under varying levels of real-world fall data availability (FARSEEING). The x-axis represents the fraction of fall events used for training, while all ADL samples are retained. The shaded region highlights the extreme data scarcity regime (1–10% falls). Results are averaged across five randomly sampled subsets of fall events.

training or adaptation. To enable a direct comparison between in-domain and cross-domain performance, the same FallAllD-trained model is evaluated both on held-out simulated data and on the FARSEEING test folds. Table 3 reports the results as the mean over five subject-wise folds, sorted by relative degradation.

The simulation–reality gap is substantial and affects all methods. Mantis, MiniRocket, and WEASEL exceed $F_1 = 0.95$ on simulated falls but fall to 0.37–0.50 on real falls, confirming that high simulated performance can badly overestimate real-world effectiveness. QUANT transfers best among the established methods (real F_1 0.62), while FallLM achieves the highest real-world F_1 (0.67) and the smallest relative degradation overall (9.5% drop in F_1).

The precision and recall values further reveal that the methods fail in distinct ways. MiniRocket, WEASEL, Mantis, and QUANT retain high precision under transfer but suffer large recall drops, the largest for WEASEL (-0.71). Having learned the morphology of simulated falls, they fire only on the real falls that most resemble them, missing the majority. The two SAX-based methods, FallLM and MrSQM, retain more recall under transfer (drops of 0.30 and 0.31, versus 0.46–0.71 for the others), indicating that the symbolic discretisation helps preserve sensitivity across domains. However, unlike MrSQM, which suffers a precision drop (-0.49), FallLM is the only method whose precision *improves* on real data (from 0.59 to 0.68). We attribute this to the absolute impact magnitude, expressed

Table 3. Cross-dataset transfer: models trained on simulated falls (FallAllD) and evaluated on real-world falls (FARSEEING), sorted by relative F_1 degradation. Real-world values are mean(std) over five subject-wise FARSEEING folds. P: precision, R: recall.

		Simulated			Real			
Model	**Representation**	F_1	P	R	F_1	P	R	**F_1 Drop%**
FallLM	SAX (symbolic)	0.74	0.59	0.97	**0.67**(0.09)	0.68(0.14)	0.67(0.11)	**9.5**
Quant	Interval-based	0.95	0.97	0.94	0.62(0.04)	0.90(0.06)	0.48(0.06)	34.7
MrSQM	SAX (symbolic)	0.94	0.91	0.97	0.51(0.08)	0.42(0.05)	0.67(0.15)	45.7
MiniRocket	Kernel-based	0.96	0.93	0.99	0.50(0.08)	0.89(0.10)	0.36(0.08)	47.9
Mantis	Foundation model	0.98	0.97	0.99	0.50(0.05)	0.79(0.13)	0.37(0.06)	49.0
WEASEL	SFA (symbolic)	0.95	0.95	0.95	0.37(0.09)	0.85(0.11)	0.24(0.08)	62.1

in physical units of g, which remains valid across datasets. This suggests that while symbolic discretisation aids recall retention, it is the physically-grounded impact tokens that additionally preserve precision.

These results show that in-domain accuracy is a poor predictor of cross-dataset robustness, and that the *type* of representational prior matters more than raw performance: representations anchored to dataset-invariant physical quantities transfer better than those that model dataset-specific signal statistics.

5.4 Symbolic Motion Pattern Analysis

A key advantage of FallLM's symbolic representation is that the learned model is inspectable. Each feature is a human-readable motion motif, and its logistic regression weight indicates how strongly it pushes a window toward the fall or ADL class (Table 4). Fall and ADL motifs frequently share amplitude-shape prefixes and differ mainly in their impact token. For example, `b d c impact_med` is a strong fall indicator (weight +1.83), while `b d c impact_low`, identical in shape, pushes towards the ADL class (weight −0.98, not among the top motifs shown). Fall motifs are characterised by `impact_high` and `impact_med` (often with `rel_impact_high`), whereas ADL motifs are dominated by `impact_low`. This confirms that FallLM's decisions rest on physically-grounded impact magnitude, and explains its limited in-domain precision: the boundary between falls and ADLs lies in the medium-impact regime, where vigorous activities and genuine falls are least separable. The acceleration waveforms underlying these motifs (see Appendix B) confirm that the impact tokens correspond to physically meaningful peaks.

Figure 4 further shows the distribution of impact levels among the top fall and ADL motifs learned on each dataset. On FallAllD, every top fall motif carries `impact_high`, so the model learns an exclusively high-impact notion of a fall. On FARSEEING, the fall vocabulary extends into the medium-impact range, reflecting that real-world falls span a wider range of intensities. This explains the transfer behaviour of Section 5.3: a model trained on FallAllD inherits a high-impact-only fall vocabulary that transfers as a strict, high-precision detector on

Table 4. Top fall and ADL motifs learned by FallLM on FARSEEING, by logistic regression weight.

Fall motifs		**ADL motifs**	
Motif	Weight	Motif	Weight
`d c impact_high rel_impact_high`	+2.60	`c c c impact_low`	−2.35
`b d c impact_med`	+1.83	`c c impact_low rel_impact_low`	−1.76
`b d c impact_high rel_impact_high`	+1.80	`c c impact_low rel_impact_med`	−1.74
`b d c impact_high`	+1.80	`c c c impact_low rel_impact_med`	−1.71
`d c impact_med rel_impact_high`	+1.60	`c c c impact_low rel_impact_low`	−1.65

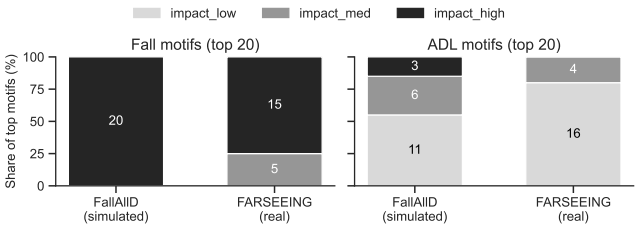

Fig. 4. Impact-level distribution of the top-20 fall and ADL motifs on FallAllD and FARSEEING. Numbers are motif counts (out of 20).

FARSEEING, where high impact is unambiguous, but which misses lower-impact real falls, capping its recall.

6 Discussion

Our results point to a single overarching lesson: in-domain accuracy is a poor predictor of real-world robustness. The representations that dominate the simulated benchmarks degrade severely under domain shift. A practitioner selecting a method on simulated performance alone would choose among the least deployable options. This reframes how wearable fall detection should be evaluated: cross-dataset and scarcity behaviour are not secondary robustness checks but primary selection criteria.

These results suggest that what governs transfer is not model simplicity but the *type* of prior a representation encodes. Methods that learn dataset-specific signal statistics capture the morphology of simulated falls and lose recall when real falls differ in shape; FallLM instead relies on absolute impact magnitude (in units of g), a prior whose meaning is invariant across datasets. This is supported by the recall decomposition (Section 5.3) and by MrSQM, a second SAX method that, lacking impact anchoring, collapses in precision whereas FallLM's improves.

This robustness has a cost: FallLM is a probe of the principle rather than a deployable detector. Its anchoring makes it imprecise in-domain, where genuine falls and vigorous activities overlap in the medium-impact regime (Section 5.4), and biased toward high-impact events. Trained on simulated falls, whose discriminative signal lies at the high-impact end, it inherits a high-impact-only notion of a

fall and systematically misses lower-impact real falls. The same prior that confers robustness thus bounds the method's ceiling: a deployable detector would need to combine impact-anchored robustness with sensitivity to low-impact events.

For practice, these findings carry several implications. Cross-dataset evaluation should be treated as a standard reporting requirement rather than an optional analysis, since simulated performance can greatly overstate real-world effectiveness. Additionally, under data scarcity, representation choice matters more than model capacity.

On the choice of detection paradigm, our binary formulation does not exploit the structure of normal activity that one-class methods model. However, such approaches face a specific difficulty in this domain: high-impact ADLs such as heavy sitting or jumping are statistically anomalous yet are not falls, so detectors keyed on unusual motion would tend to flag them (Section 5.4). A systematic comparison of detection paradigms is left to future work.

Several limitations bound these conclusions. Our analysis rests on a single clinically-verified real-world dataset, and the scarcity of real fall data that motivates this work also limits the breadth of our cross-dataset claims. We evaluate one representative method per family, each with its canonical classifier, so observed differences reflect representation–classifier pipelines rather than representations in isolation. We deliberately study zero-shot transfer to measure the unadapted simulation–reality gap; domain adaptation would likely narrow it but would obscure the quantity we set out to characterise. Finally, the interpretability analysis reflects the highest-weighted motifs of a single model and is illustrative rather than exhaustive.

7 Conclusion

We presented a systematic evaluation of motion representations for wearable fall detection under real-world data scarcity and domain shift, comparing interval-based, kernel-based, symbolic, and foundation model approaches on simulated and clinically-verified real-world falls. Across cross-validation, controlled data scarcity, and cross-dataset transfer, we found that the methods strongest on simulated data degrade most under domain shift: the benchmark-topping model is not the most deployable one. Robustness instead tracks the type of prior a representation encodes. Representations anchored to dataset-invariant physical quantities transfer more gracefully than those modelling dataset-specific signal statistics.

As an interpretable case study, FallLM illustrates both sides of this principle: its physically-grounded impact tokens yield the smallest transfer degradation, the highest real-world F_1, and detection from as few as two fall examples, but at the cost of in-domain precision and a structural blindness to low-impact falls. As real-world fall data remains extremely scarce, we argue that cross-dataset evaluation should become standard practice, and that reconciling transfer robustness with precision is a key direction for future work. To support reliable benchmarking, we

release our code and streaming evaluation pipeline, and continue to work toward the curation of open real-world fall datasets.

Acknowledgments. This study has emanated from research funded by Research Ireland under the Government of Ireland Postdoctoral Fellowship Programme (Grant Number: GOIPD/2025/1434).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Aderinola, T.B., Palmerini, L., D'Ascanio, I., Chiari, L., Klenk, J., Becker, C., Caulfield, B., Ifrim, G.: Watch your step: A cost-sensitive framework for accelerometer-based fall detection in real-world streaming scenarios. arXiv preprint arXiv:2509.11789 (2025)
2. Aderinola, T.B., Palmerini, L., D'Ascanio, I., Chiari, L., Klenk, J., Becker, C., Caulfield, B., Ifrim, G.: Accurate and efficient real-world fall detection using time series techniques. In: International Workshop on Advanced Analytics and Learning on Temporal Data. pp. 52–79. Springer (2024)
3. Bach, N.C., Van Chi, N., Cuong, D.D., Tuan, N.A., Kieu, T.Q.T., Phuong, N.T., Thien, N.D., Thao, L.Q.: A lightweight and efficient deep learning model for real-time fall detection on edge device. Biomedical Signal Processing and Control **113**, 108942 (2026)
4. Camp, K., Murphy, S., Pate, B.: Integrating fall prevention strategies into ems services to reduce falls and associated healthcare costs for older adults. Clinical interventions in aging pp. 561–569 (2024)
5. Dempster, A., Petitjean, F., Webb, G.I.: Rocket: exceptionally fast and accurate time series classification using random convolutional kernels. Data Mining and Knowledge Discovery **34**(5), 1454–1495 (2020)
6. Dempster, A., Schmidt, D.F., Webb, G.I.: Minirocket: A very fast (almost) deterministic transform for time series classification. In: Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining. pp. 248–257 (2021)
7. Dempster, A., Schmidt, D.F., Webb, G.I.: Quant: A minimalist interval method for time series classification. Data Mining and Knowledge Discovery pp. 1–26 (2024)
8. Feofanov, V., Wen, S., Alonso, M., Ilbert, R., Guo, H., Tiomoko, M., Pan, L., Zhang, J., Redko, I.: Mantis: Lightweight calibrated foundation model for user-friendly time series classification. arXiv preprint arXiv:2502.15637 (2025)
9. Gil-Martín, M., López-Iniesta, J., Fernández-Martínez, F., San-Segundo, R.: Reducing the impact of sensor orientation variability in human activity recognition using a consistent reference system. Sensors **23**(13), 5845 (2023)
10. Huynh, Q.T., Tran, B.Q.: Time-frequency analysis of daily activities for fall detection. Signals **2**(1), 1–12 (2021)
11. Klenk, J., Schwickert, L., Palmerini, L., Mellone, S., Bourke, A., Ihlen, E.A., Kerse, N., Hauer, K., Pijnappels, M., Synofzik, M., et al.: The farseeing real-world fall repository: a large-scale collaborative database to collect and share sensor signals from real-world falls. European review of aging and physical activity **13**, 1–7 (2016)
12. Lee, J.S., Tseng, H.H.: Development of an enhanced threshold-based fall detection system using smartphones with built-in accelerometers. IEEE Sensors Journal **19**(18), 8293–8302 (2019)

13. Liu, C.P., Li, J.H., Chu, E.P., Hsieh, C.Y., Liu, K.C., Chan, C.T., Tsao, Y.: Deep learning-based fall detection algorithm using ensemble model of coarse-fine cnn and gru networks. In: 2023 IEEE International Symposium on Medical Measurements and Applications (MeMeA). pp. 1–5. IEEE (2023)
14. Liu, J., Li, X., Huang, S., Chao, R., Cao, Z., Wang, S., Wang, A., Liu, L.: A review of wearable sensors based fall-related recognition systems. Engineering Applications of Artificial Intelligence **121**, 105993 (2023)
15. Middlehurst, M., Ismail-Fawaz, A., Guillaume, A., Holder, C., Rubio, D.G., Bulatova, G., Tsaprounis, L., Mentel, L., Walter, M., Schäfer, P., et al.: aeon: a python toolkit for learning from time series. arXiv preprint arXiv:2406.14231 (2024)
16. Mohan, D., Al-Hamid, D.Z., Chong, P.H.J., Sudheera, K.L.K., Gutierrez, J., Chan, H.C., Li, H.: Artificial intelligence and iot in elderly fall prevention: A review. IEEE Sensors Journal **24**(4), 4181–4198 (2024)
17. Nguyen, D.A., Pham, C., Argent, R., Caulfield, B., Le-Khac, N.A.: Model and empirical study on multi-tasking learning for human fall detection. Vietnam Journal of Computer Science pp. 1–14 (2024)
18. Nguyen, T.L., Ifrim, G.: Fast time series classification with random symbolic subsequences. In: International Workshop on Advanced Analytics and Learning on Temporal Data. pp. 50–65. Springer (2022)
19. Owusu, E., Acquah, I., Asare, M.A., Yeboah, B.A.: Litefallnet: A lightweight deep learning model for efficient real-time fall detection. Digital Health **11**, 20552076251386698 (2025)
20. Palmerini, L., Klenk, J., Becker, C., Chiari, L.: Accelerometer-based fall detection using machine learning: Training and testing on real-world falls. Sensors **20**(22), 6479 (2020)
21. Rastogi, S., Singh, J.: A systematic review on machine learning for fall detection system. Computational intelligence **37**(2), 951–974 (2021)
22. Saleh, M., Abbas, M., Le Jeannes, R.B.: Fallalld: An open dataset of human falls and activities of daily living for classical and deep learning applications. IEEE Sensors Journal **21**(2), 1849–1858 (2020)
23. Schäfer, P., Leser, U.: Weasel 2.0: a random dilated dictionary transform for fast, accurate and memory constrained time series classification. Machine Learning **112**(12), 4763–4788 (2023)
24. Sucerquia, A., López, J.D., Vargas-Bonilla, J.F.: Sisfall: A fall and movement dataset. Sensors **17**(1), 198 (2017)
25. Tan, C.W., Dempster, A., Bergmeir, C., Webb, G.I.: Multirocket: multiple pooling operators and transformations for fast and effective time series classification: Cw tan. Data Mining and Knowledge Discovery **36**(5), 1623–1646 (2022)
26. Vignesh, P.G., Kumar, N., et al.: Lidar-based elderly fall detection system for indoor environments using neural network algorithms. In: 2025 4th International Conference on Sentiment Analysis and Deep Learning (ICSADL). pp. 1564–1568. IEEE (2025)
27. World Health Organization: Step safely: strategies for preventing and managing falls across the life-course (2021)
28. Zafar, R.O., Zafar, F.: Real-time activity and fall detection using transformer-based deep learning models for elderly care applications. BMJ Health & Care Informatics **32**(1), e101439 (2025)
29. Zhang, J., Li, Z., Liu, Y., Li, J., Qiu, H., Li, M., Hou, G., Zhou, Z.: An effective deep learning framework for fall detection: model development and study design. Journal of medical internet research **26**, e56750 (2024)

Appendix A FallLM Configuration

Table A1 summarises the fixed FallLM hyperparameters used throughout the paper. These values define the production configuration evaluated in the main paper; the ablations in the following sections vary one parameter at a time while holding all others at these values. In the ablation tables, the rows labelled *default* (`n_bins`=5) and *3-level* (impact token) correspond to this configuration.

Table A1. FallLM default configuration. Parameters held fixed across all experiments unless explicitly ablated.

Component	Setting
Window length	3 s (300 samples at 100 Hz)
PAA word size	3 (one symbol per 1 s phase)
SAX alphabet size (`n_bins`)	5 (a–e)
Absolute impact token	3 levels: low ($< 1.8g$), medium (1.8–2.5g), high ($> 2.5g$)
Relative impact token	3 levels (peak vs. within-window distribution)
Tokens per motion sentence	5 (3 SAX + 2 impact)
n-gram range	$(4, 5)$
Vectoriser	TF-IDF
Classifier	Logistic regression

Appendix A.1 Impact-token configuration

We report a transfer ablation (FallAllD → FARSEEING) over three configurations of the impact token, with the SAX alphabet fixed at the 5-bin production default. The *SAX + impact (3-level)* configuration corresponds to FallLM exactly as evaluated in the main paper.

Table A2. Impact-token configuration ablation. FallAllD columns report the in-domain (simulated) score on a single held-out split; FARSEEING columns report mean (std) across the five FARSEEING transfer folds (real-world). The 3-level configuration is FallLM as used in the main paper.

	FallAllD (sim)			FARSEEING (real, mean (std))		
Configuration	F_1	P	R	F1	P	R
SAX only	0.670	0.808	0.573	0.418 (0.057)	0.302 (0.048)	0.692 (0.108)
SAX + impact (3-level)	0.738	0.594	0.973	0.667 (0.093)	0.682 (0.136)	0.668 (0.111)
SAX + impact (12-bin)	0.794	0.684	0.945	0.464 (0.136)	0.711 (0.184)	0.350 (0.119)

Table A2 compares three impact-token configurations under transfer: no impact token (*SAX only*), the 3-level categorical token used in our final design

(*SAX + impact, 3-level*), and a fine-grained 12-bin physical-magnitude token (*SAX + impact, 12-bin*) with bins learned from the training data. With five SAX symbols, shape information alone is already informative in-domain: *SAX only* reaches $F_1 = 0.670$. Adding either impact token improves in-domain performance further, to $F_1 = 0.738$ (3-level) and $F_1 = 0.794$ (12-bin), the 12-bin variant being best in-domain.

The picture reverses under transfer, and the two alternatives fail in opposite ways. The 12-bin variant collapses to $F_1 = 0.464 \pm 0.136$ through a sharp loss of recall ($R = 0.350$): its fine-grained bins, fitted to FallAllD's impact distribution, match too few of FARSEEING's real falls, so it fires rarely (precision $P = 0.711$) but misses most falls. *SAX only* fails in the opposite direction ($F_1 = 0.418 \pm 0.057$): lacking any impact cue, its symbolic vocabulary is not discriminative enough to suppress false positives, giving low precision ($P = 0.302$) despite reasonable recall. The 3-level categorical token avoids both failure modes, giving by far the best real-world performance ($F_1 = 0.667 \pm 0.093$) and the smallest drop from its in-domain score ($0.738 \rightarrow 0.667$). This confirms that coarse, dataset-invariant magnitude categories generalise better than either no magnitude information or high-resolution magnitude fitted to a specific population.

Appendix A.2 SAX Alphabet Size

We fix the SAX alphabet size to `n_bins`=5, motivated primarily by interpretability. Five amplitude levels yield a compact, human-readable symbol set (a–e) while retaining sufficient resolution to capture meaningful motion patterns. Smaller alphabets tend to produce highly compressed symbolic vocabularies in which distinct motion patterns collapse to similar motifs, whereas larger alphabets increase symbolic complexity without providing a clear performance advantage. We therefore selected `n_bins`=5 as a practical balance between symbolic expressiveness and interpretability.

To verify that the conclusions of the paper are not sensitive to this choice, we sweep `n_bins` from 3 to 10 on the in-domain FallAllD dataset (Table A3) and, for completeness, report the corresponding sweep on FARSEEING (Table A4).

Table A3. Alphabet size ablation on FallAllD. Mean (std) over subject-wise 5-fold CV.

`n_bins`	F_1	Precision	Recall
3	0.745 (0.047)	0.607 (0.061)	0.971 (0.028)
4	0.753 (0.046)	0.616 (0.061)	0.973 (0.030)
5 (default)	0.746 (0.047)	0.606 (0.062)	0.976 (0.026)
6	0.751 (0.048)	0.615 (0.062)	0.969 (0.030)
7	0.727 (0.071)	0.631 (0.061)	0.887 (0.173)
8	0.747 (0.046)	0.615 (0.059)	0.956 (0.020)
9	0.738 (0.064)	0.666 (0.074)	0.828 (0.049)
10	0.741 (0.050)	0.613 (0.064)	0.943 (0.036)

On FallAllD, performance is essentially flat across the entire range, with F_1 varying only between 0.727 and 0.753. The default choice of `n_bins`=5 is not the highest-scoring configuration, lying marginally below `n_bins`=4 (0.753), `n_bins`=6 (0.751), and `n_bins`=8 (0.747), and well within one standard deviation of all other settings. These results indicate that alphabet size is not performance-critical in-domain.

Table A4. Alphabet size ablation on FARSEEING, reported for completeness. This sweep was *not* used to select `n_bins`. Mean (std) over subject-wise 5-fold CV.

`n_bins`	F_1	Precision	Recall
3	0.645 (0.106)	0.517 (0.125)	0.882 (0.083)
4	0.518 (0.175)	0.423 (0.225)	0.775 (0.069)
5 (default)	0.642 (0.092)	0.518 (0.106)	0.866 (0.100)
6	0.618 (0.142)	0.514 (0.182)	0.822 (0.055)
7	0.637 (0.125)	0.528 (0.146)	0.830 (0.075)
8	0.597 (0.136)	0.484 (0.171)	0.826 (0.067)
9	0.626 (0.145)	0.505 (0.160)	0.860 (0.042)
10	0.633 (0.104)	0.519 (0.131)	0.836 (0.070)

On FARSEEING, the same broad stability holds for `n_bins` ≥ 5, with all settings falling within one standard deviation of one another. The only notable deviation occurs at `n_bins`=4 ($F_1 = 0.518$), where performance drops sharply relative to neighbouring settings, suggesting that this particular alphabet size merges otherwise distinguishable motion patterns. More generally, FallLM exhibits limited sensitivity to alphabet size across both datasets, indicating that the representation is robust to moderate changes in symbolic resolution.

Crucially, the alphabet size was not fixed based on FARSEEING performance. Across the entire `n_bins` ≥ 5 range, FallLM's in-domain behaviour is stable. More broadly, the conclusions of the paper concern differences between representation families rather than a specific FallLM configuration.

Appendix B Motif Waveforms

To illustrate that FallLM's symbolic tokens correspond to physically meaningful motion, Figure A1 shows the raw acceleration waveforms underlying the three highest-weighted fall and ADL motifs learned on FARSEEING (cf. Table 4 of the main paper). Each fall motif exhibits a characteristic acceleration peak whose magnitude tracks the absolute impact token: motifs carrying `impact_high` peak around 2 g, while `impact_med` motifs peak lower. The ADL motifs, all carrying `impact_low`, remain close to 1.5 g throughout, with no comparable peak. This confirms that the discrete impact tokens are not arbitrary symbols but correspond directly to interpretable acceleration patterns, supporting the token-level analysis in the main paper.

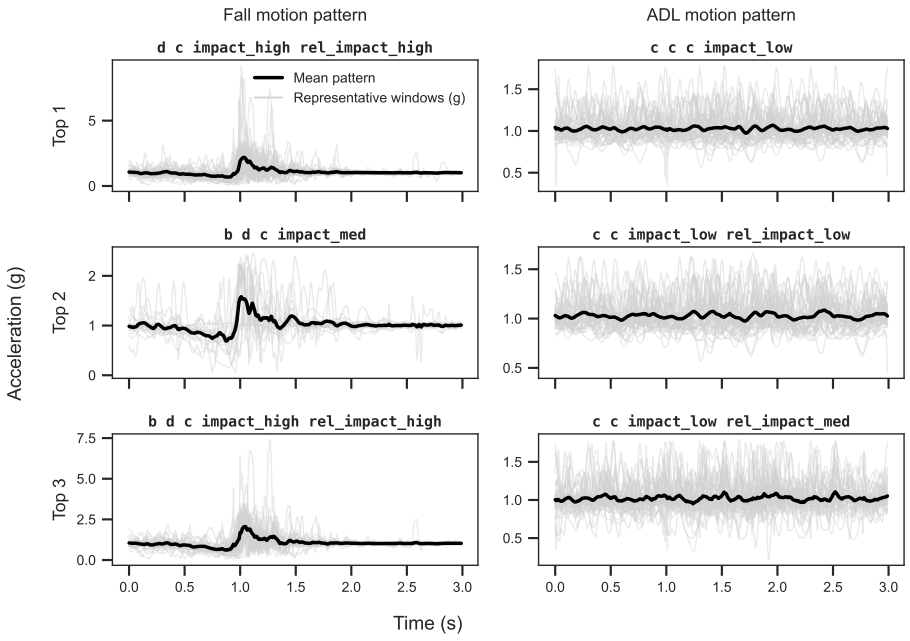

Fig. A1. Acceleration waveforms of the three highest-weighted fall (left) and ADL (right) motifs learned by FallLM on FARSEEING. Grey curves show representative windows for each motif; the black curve is their mean.