HALO: LEARNING MOTION REPRESENTATIONS AS CONTROLLED DEPARTURES FROM SIGNAL STATISTICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Handcrafted statistical features remain surprisingly competitive for wearable recognition, especially when models must transfer across sensor placements, orientations, and acquisition conditions. Their strength comes from simple motion statistics that are stable across domains, but these fixed summaries can miss sample-specific cues needed for recognition. We therefore hypothesize that robust transfer is better achieved by *refining*, rather than replacing, this statistical structure. To test this hypothesis, we introduce HALO (**H**andcrafted **A**nchors, **L**imited **O**ffsets), a compact 0.16M-parameter encoder that represents each input with a handcrafted statistical anchor and a bounded neural correction. The anchor provides a stable reference across domains, while the correction adds discriminative information that fixed statistics do not capture. We pretrain HALO in HARCauKer, a synthetic motion environment with physically coupled inertial rendering, and evaluate the frozen encoder on 14 public datasets. HALO consistently improves recognition under sensor-position and cross-dataset shifts. Its frozen representations also transfer effectively through classifier-free retrieval. Disrupting the learned correction directions sharply degrades retrieval, showing that the offsets capture transferable, sample-specific structure rather than arbitrary residual variation. These results suggest a simple principle for wearable representation learning: preserve what stable statistics already capture, and learn what they miss.

1 INTRODUCTION

Wearable activity recognition is difficult to reuse across deployment conditions. A change in wearer, body location, device, or recording pipeline can substantially alter the measured waveform even when the underlying activity remains the same, while collecting labels for every new condition is costly. A useful wearable representation must therefore preserve activity-relevant information while remaining robust to these shifts. Despite substantial progress in deep wearable sensing and motion pretraining, generalization across people, placements, and datasets remains a persistent challenge (Bao & Intille, 2004; Bulling et al., 2014; Stisen et al., 2015; Sztyler & Stuckenschmidt, 2016; Ordóñez & Roggen, 2016; Yuan et al., 2024; Narayanswamy et al., 2025; Cai et al., 2026).

This challenge exposes a tension in representation design. Handcrafted statistics encode known signal structure and often provide strong descriptors without encoder training (Preece et al., 2009; Christ et al., 2018; Lubba et al., 2019), but their predefined vocabulary cannot adapt to other information. Convolutional feature transforms such as ROCKET and MINIROCKET offer another alternative to trained deep encoders (Dempster et al., 2020; 2021). Deep encoders adapt, but learn their representation largely from data (Hammerla et al., 2016; Mahmud et al., 2020; Yue et al., 2022; Feofanov et al., 2025; 2026). Recent methods combine both: rTsfNet predicts multiple input-dependent 3D IMU rotations, extracts handcrafted features from each view, and combines them for classification (Enokibori, 2024); SensorLM builds sensor-language representations from precomputed wearable features (Zhang et al., 2025). This raises a further question: *instead of learning a transferable representation from scratch, can we start from a useful statistical representation and learn only how to refine it?*

We hypothesize the latter. This motivates HALO (**H**andcrafted **A**nchors, **L**imited **O**ffsets), which starts from handcrafted statistics and learns only the information needed to adapt them (Fig. 1). For each sensor triad, the handcrafted features form an anchor, and a neural encoder predicts an input-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

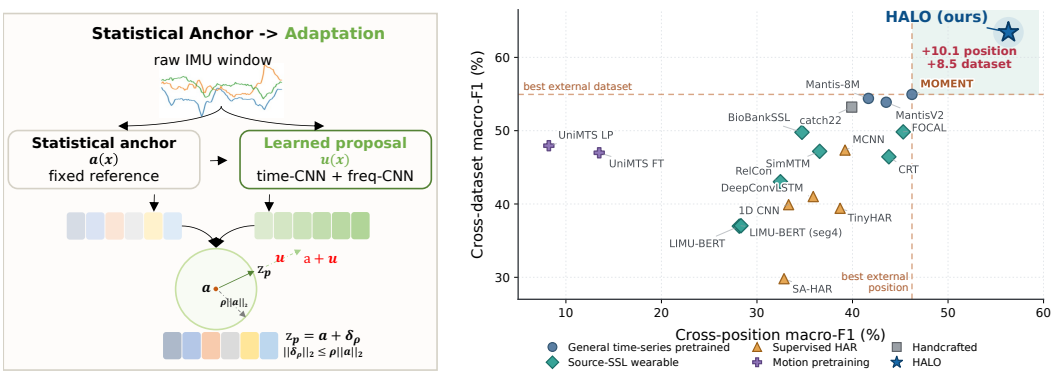

Figure 1: **HALO refines a statistical representation through controlled learned adaptation. Left:** An input-dependent offset adds information absent from fixed features. A radius bounds departure from the anchor, yielding several representations from one encoder prediction. **Right:** Mean macro-F1 for cross-position and cross-dataset transfer. HALO improves both axes; dashed lines mark the strongest external result on each (higher is better).

dependent offset from that anchor. The offset is constrained so that learning can enrich the statistical representation without freely redefining it. In this way, statistics serve as the basis of the learned representation rather than as a separate feature stream.

A radius controls adaptation around the anchor: zero recovers the original statistics, while larger values give the learned offset more influence. One encoder prediction thus yields a family of representations at different departures from the prior, letting us test how much adaptation benefits transfer without choosing a fixed or fully learned representation.

Learning such a correction introduces a second challenge: the model should adapt to motion, not to incidental properties of the acquisition process. In real wearable datasets, these factors are often entangled, making it difficult to determine whether a representation captures the underlying activity or the conditions under which it was recorded. Prior work demonstrates synthetic pretraining (Hollmann et al., 2022; Dooley et al., 2023; Xie et al., 2026; Feofanov et al., 2026) and motion-to-sensor simulation (Young et al., 2011; Leng et al., 2024; Zhang et al., 2024b). We build on these ideas with HAR-CauKer, a motion-first synthetic pretraining environment in which latent motion programs drive a rigid-body renderer to generate physically coupled inertial measurements. Because the simulator exposes the factors that generated each observation, we can supervise the encoder with program identity, view agreement, and latent motion factors that are unavailable in real recordings (Section 3.3). After pretraining, we discard these supervision heads and evaluate the frozen representation.

The results reveal a consistent division of labor between prior structure and learned adaptation. The statistical anchor already exceeds every evaluated external method on cross-position and cross-dataset recognition. HALO improves this strong reference by a further 7.2 and 3.9 macro-F1 points and leads all six position-transfer directions. Classifier-free retrieval shows that the improvement is already present in the frozen geometry: the learned direction transfers on its own, and its bounded composition with the anchor performs best. Input-matched direction interventions and structural controls further distinguish this refinement from wider features, direct concatenation, or an unconstrained residual. Together, these results support a simple view of wearable representation learning: retain stable signal structure and learn what it misses.

Our contributions are threefold:

- We introduce HALO, a compact representation framework that uses generic handcrafted statistics as a reference and learns controlled, input-dependent corrections around that reference, yielding a radius-indexed family of transferable representations.
- We introduce HAR-CauKer, a motion-first synthetic pretraining environment with physically coupled inertial rendering, acquisition variation, and supervision from simulator-only motion factors; ablations identify acquisition diversity as its main source of transfer.

- Across 14 public datasets, we show that learned directions add complementary, sample-specific structure to an already strong statistical anchor, while bounded composition improves cross-position and cross-dataset transfer without replacing that reference.

2 Related Work

2.1 Handcrafted and Learned Representations for Wearable Signals

Handcrafted representations describe sensor windows through explicit signal calculations. Classical inertial HAR uses time- and frequency-domain summaries of magnitude, variability, and periodicity (Bao & Intille, 2004; Preece et al., 2009; Bulling et al., 2014); TSFRESH organizes a large feature library, while CATCH22 selects a compact set of broadly useful characteristics (Christ et al., 2018; Lubba et al., 2019). These definitions apply across recordings but do not guarantee invariance to placement or calibration; they provide reusable descriptors yet omit properties outside the chosen vocabulary. ROCKET and MINIROCKET instead use convolutional feature transforms (Dempster et al., 2020; 2021), while deep HAR models learn filters and temporal interactions from data (Ordóñez & Roggen, 2016; Zhou et al., 2022; Bian et al., 2025).

Recent work combines designed and learned features. rTsfNet predicts multiple input-dependent 3D IMU rotations, recomputes handcrafted features in each coordinate system, and combines the views for classification (Enokibori, 2024): learning changes *where* measurements are made. At a different scale and task, SensorLM aligns text with minute-level features derived from wearable sensors (Zhang et al., 2025): engineered measurements define *what* the model reads. Both show that established signal computations can support learning; HALO instead refines the statistical description.

2.2 Self-Supervised Pretraining for Wearable Signals

Self-supervised objectives specify what an encoder should retain. LIMU-BERT models masked IMU signals (Xu et al., 2021), MOMENT reconstructs masked patches across heterogeneous time series (Goswami et al., 2024), and HiMAE probes different temporal resolutions in a hierarchical masked autoencoder (Lee et al., 2026). Contrastive methods specify related observations: predictive coding and TS2Vec use temporal context (Haresamudram et al., 2021; Yue et al., 2022); FOCAL separates modality-shared from modality-specific information (Liu et al., 2023); and RelCon grades accelerometry similarity with a learned motif distance (Xu et al., 2025). Each design encodes a different prior about what unlabeled signals should preserve.

Scale and domain variation pose further questions. Large wearable studies use extensive unlabeled recordings to learn reusable embeddings (Yuan et al., 2024; Narayanswamy et al., 2025); MantisV2 explores synthetic pretraining and test-time aggregation for frozen encoders (Feofanov et al., 2026). For distribution shift, CrossHAR diversifies source signals and pretrains hierarchically for unseen datasets (Hong et al., 2024), while ContrastSense uses domain information in contrastive learning (Dai et al., 2024). BenchHAR finds that strong within-dataset learning does not settle generalization to unseen conditions (Cai et al., 2026). Objectives, variation, and protocols shape transfer. HALO asks how pretraining refines useful statistics across wearers, positions, and datasets.

2.3 Synthetic Pretraining for Inertial Sensing

Synthetic pretraining makes variation available without collecting labels for every target condition. Prior-generated tasks have supported tabular inference and time-series forecasting (Hollmann et al., 2022; Dooley et al., 2023); for classification, CauKer composes Gaussian-process kernels within causal programs to generate structured time series (Xie et al., 2026). These generators can vary latent temporal patterns at scale. A generic generated sequence does not specify how a moving body and its mounted sensors jointly produce the observed channels.

Motion-to-sensor work addresses that physical link. IMUSim renders inertial readings from trajectories (Young et al., 2011); IMUGPT 2.0 converts generated human motion into virtual IMU data (Leng et al., 2024); and UniMTS synthesizes wearable sequences from motion skeletons to cover locations and orientations (Zhang et al., 2024b). This line models what a sensor could observe from a specified motion, while synthetic time-series pretraining provides control over the motion-

generating process. HAR-CauKer brings the two together by driving physically coupled inertial rendering with programmable latent motion and using the known motion factors during pretraining.

3 Reference-Relative Motion Pretraining

HALO treats an existing statistical description as a useful reference rather than asking an encoder to rediscover it. For each inertial signal, fixed statistics define an anchor, a neural network proposes an input-dependent correction in the same space, and a radius limits departure from the anchor. Learning the correction then requires supervision that distinguishes motion from changes in how sensors observe it. We train on simulated motion, then freeze the encoder for transfer.

3.1 Statistical anchor and learned correction

For a triad $x \in \mathbb{R}^{3\times T}$ resampled to $L = 128$, define scale and shape:

$$s(x) = \max\Big\{\epsilon, \left(L^{-1}\sum_{t=1}^{L}\|x_t\|_2^2\right)^{1/2}\Big\}, \qquad \widetilde{x} = x/s(x), \quad \epsilon = 10^{-6}. \tag{1}$$

The descriptor and neural branches receive $\widetilde{x}$, while $\log s(x)$ retains magnitude. The norm-based scale split commutes with axis rotations, though individual statistics may not.

We build a fixed vocabulary crossing derived signals, temporal resolutions, statistics, segment aggregations, and waveform or spectral domains. Of its 10,098 candidate coordinates, we sample 128 once with seed 0 and append $\log s(x)$. Affine-free BatchNorm fitted on synthetic data gives the 129-dimensional anchor $a(x)$ and a common coordinate scale for the departure bound. This avoids target-specific selection; see Sections B and C for definitions and draw tests.

The anchor cannot adapt to cues outside its fixed vocabulary. A dilated temporal CNN captures local and longer-range patterns in $\widetilde{x}$, while a frequency CNN reads its normalized log-magnitude spectrum. Robust pooling gives $h_{\text{time}}(x), h_{\text{freq}}(x) \in \mathbb{R}^{128}$.

$$u(x) = \text{Linear}_{385\to 129}(\text{LN}([a(x) \mid h_{\text{time}}(x) \mid h_{\text{freq}}(x)])). \tag{2}$$

The final layer is zero-initialized, so initially $u(x) = 0$ and the encoder is exactly the statistical anchor. Training learns sample-specific directions without reconstructing the named statistics. Together, the branches, normalization, and proposal use 158,812 parameters.

3.2 Bounded departures from the anchor

We constrain the learned proposal to an anchor-relative ball, retaining the statistical reference:

$$\delta_\rho(x) = u(x)\min\left\{1, \frac{\rho\|a(x)\|_2}{\max(\|u(x)\|_2, \epsilon)}\right\}, \qquad z_\rho(x) = a(x) + \delta_\rho(x). \tag{3}$$

This scalar projection preserves the proposal direction and guarantees $\|z_\rho(x) - a(x)\|_2 \le \rho\|a(x)\|_2$ for every input. Thus $\rho = 0$ recovers the anchor, while larger radii permit more adaptation in standardized feature space. We train at $\rho = 1$; one encoder pass supplies $a(x)$ and $u(x)$ for any radius, with only scalar projections added. The radius is thus an explicit intervention on representation distance, evaluated through fixed-radius transfer comparisons.

3.3 Motion-first synthetic supervision

The bound controls distance, not whether a correction reflects motion or sensor placement. CauKer treats causal-program outputs as observed series (Xie et al., 2026). HAR-CauKer instead assigns three graph nodes each to angular velocity and world-frame acceleration. A rigid-body renderer integrates angular velocity into orientation, rotates acceleration and a fixed magnetic field into the sensor frame, and adds gravity to the accelerometer. Gyroscope, linear-acceleration, accelerometer, and magnetometer signals thus share one latent motion rather than being generated independently. Two acquisition-corrupted views retain that motion while varying sampling, mounting, calibration, and noise (Fig. 3 and Section B).

Three heads supervise z_1 from both views. Program classification predicts which of 3,000 fixed causal graphs generated an instance, averaging cross-entropy over views; the classes denote generating dynamics, not human activities. Contrastive NT-Xent treats two acquisition-corrupted views of the same motion instance as positives. Factor regression predicts ten standardized simulator variables from both views, describing rhythm, rotation, tilt, intensity, and impact. Their equally weighted sum is $\mathcal{L} = \mathcal{L}_{\text{class}} + \mathcal{L}_{\text{contrast}} + \mathcal{L}_{\text{factor}}$. These objectives encourage mechanism discrimination, acquisition consistency, and sensitivity to motion-related factors without using real recordings or activity labels. After discarding the heads, we evaluate all three transfer protocols and ablate supervision, rendering, and corruptions on cross-dataset transfer (Section B.1).

3.4 Frozen transfer

At evaluation, each inertial triad is resampled to 128 samples and passed independently through the same frozen encoder; outputs from multiple triads are concatenated. Retrieval uses the fixed 129-dimensional z_1, while recognition uses the prespecified $[z_{0.2} \mid z_1 \mid z_{10}]$ stack from one encoder pass. The encoder, anchor normalization, and radii remain fixed, and only a source-domain readout is fitted for each channel layout. See Sections B and 4 for preprocessing and readout details.

4 Evaluation Setup

Our evaluation follows HALO from task performance to representation structure. We first test whether a frozen encoder supports recognition when subjects, sensor positions, or entire datasets change. We then remove the task classifier and inspect whether the same activities remain neighbors across conditions. Finally, component and design controls trace the transfer gains to the statistical anchor, the learned correction, and the radius that joins them. Within each evaluation unit, every representation uses the same source and target windows.

Transfer protocols. Building on the dataset collection and transfer axes of Cai et al. (2026), we define three protocols. Cross-subject holds out people within each dataset. Cross-position holds out both people and sensor body group, testing six ordered transfers among trunk, upper-limb, and lower-limb sites. Cross-dataset holds out each dataset in turn; each supervised readout is fit once on the pooled union of the other twelve datasets and tested on the holdout. These protocols yield 65 subject folds, 28 available position fold–direction units, and 13 held-out datasets. Our dataset-level leave-one-out protocol differs from BenchHAR's dataset-level five-fold split; Section A.4 details our splits and the two unavailable position units.

Task harmonization. Cross-subject evaluation preserves each dataset's native activity vocabulary and available inertial channels. Position and dataset transfer require comparable inputs and labels, so we use six-channel, six-second windows and a shared eight-activity vocabulary. UT-Complex joins only the position protocol, giving it 14 datasets versus 13 for the other two. HALO resamples each triad to 128 samples; exact pool sizes, layouts, and exceptions are in Table 7 and Section A.

Representations and baselines. For classifier-free retrieval, we compare fixed z_1 with its anchor, CATCH22, Mantis-8M, and MantisV2 (Lubba et al., 2019; Feofanov et al., 2025; 2026). Diagnostic linear probes additionally compare a, $\widehat{u}$, and z_1 from one HALO checkpoint. Headline recognition adds released MOMENT-1-base (Goswami et al., 2024), source-pretrained RelCon (Xu et al., 2025), and LIMU-BERT variants (Xu et al., 2021). HALO uses a source-trained random forest; other methods retain their specified readouts (Sections A.6 and F). Released foundation models keep their pretrained checkpoints, while benchmark-specific pretraining and all fitted readouts use source data only. Pretraining data and compute differ across released models.

Readouts, metrics, and aggregation. Retrieval reports activity-macro-averaged Recall@1, with Recall@5 and AP@100 as secondary metrics; linear probes and headline recognition use multiclass macro-F1. Across positions, retrieval and linear probes average all 28 units. Headline recognition averages folds within each ordered direction and then the six direction means, giving directions equal weight despite two missing fold–direction units. Cross-subject recognition averages folds within each dataset and then the 13 dataset means; cross-dataset scores average the 13 held-out

datasets. Main-text scores use a 0–100 scale and differences use unrounded values. Comparisons are paired by held-out unit, not window; shared source pools and target subjects make unit-bootstrap intervals descriptive rather than independent-sample confidence intervals (Section A.4).

5 Recognition Under Subject, Position, and Dataset Shifts

HALO leads a broad benchmark spanning handcrafted pipelines, general time-series foundation models, wearable self-supervised encoders, motion-pretrained models, and end-to-end supervised networks. With a source-trained random forest, its fixed $[z_{0.2} \mid z_1 \mid z_{10}]$ representation reaches $84.9/56.3/63.4$ cross-subject/position/dataset macro-F1, exceeding the strongest complete-coverage external method by $0.6/10.1/8.5$ points (Table 1). The result also reveals how HALO builds this advantage: the statistical anchor already reaches $83.9/49.1/59.5$ and leads every external method on both transfer protocols, while the learned correction adds a further $1.0/7.2/3.9$ points. Its largest contribution appears precisely when the sensor position or dataset changes.

Table 1: **Headline recognition under three shifts** (macro-F1, 0–100). Panels list all evaluated methods; margins compare HALO with the best complete-coverage external method. Per-unit scores: Tables 20 to 22.

(a) Cross-subject
+0.6 *vs. best external*

Method	Mean	Rank
HALO (ours)	**84.9**	**2.5**
HALO anchors	83.9	3.7
MOMENT	77.1	9.6
Mantis-8M	<u>84.3</u>	4.0
MantisV2	83.8	4.3
CATCH22	76.1	10.8
LIMU-BERT*	68.8	14.6
LIMU-BERT (seg4)*	69.0	14.5
RelCon*	59.2	16.4
MiniROCKET	84.3	3.0
TinyHAR	78.9	7.9
MCNN	78.8	8.5
TinierHAR	75.9	9.5
1D CNN	74.6	11.3
DeepConvLSTM	70.8	13.6
SA-HAR	66.6	15.1
UniMTS LP	65.8	15.3
UniMTS FT	80.6	6.1

(b) Cross-position
+10.1 *vs. best external*

Method	Mean	Rank
HALO (ours)	**56.3**	**1.0**
HALO anchors	49.1	3.0
MOMENT	<u>46.2</u>	4.5
MantisV2	43.5	6.3
Mantis-8M	41.7	8.0
CATCH22	39.9	8.5
LIMU-BERT*	28.2	16.7
LIMU-BERT (seg4)*	28.3	16.3
RelCon*	32.5	14.5
FOCAL	45.3	5.3
CRT	43.8	6.0
SimMTM	36.6	11.2
BioBankSSL	34.7	12.7
MCNN	39.2	8.5
TinyHAR	38.7	9.3
DeepConvLSTM	35.9	10.8
1D CNN	33.3	14.0
SA-HAR	32.8	14.3
UniMTS LP	8.2	19.8
UniMTS FT	13.5	19.2

(c) Cross-dataset
+8.5 *vs. best external*

Method	Mean	Rank
HALO (ours)	**63.4**	**3.1**
HALO anchors	59.5	4.9
MOMENT	<u>54.9</u>	7.2
Mantis-8M	54.4	7.8
MantisV2	53.9	7.8
CATCH22	53.2	8.5
LIMU-BERT*	37.0	13.3
LIMU-BERT (seg4)*	37.0	13.4
RelCon*	43.0	12.8
FOCAL	49.8	9.5
BioBankSSL	49.8	9.4
SimMTM	47.2	11.4
CRT	46.4	11.9
MCNN	47.4	9.7
DeepConvLSTM	41.0	13.4
1D CNN	39.9	14.4
TinyHAR	39.4	12.8
SA-HAR	29.8	17.9
UniMTS LP	48.0	10.5
UniMTS FT	47.0	10.3

Lower rank is better; underlining marks the best complete-coverage external mean. Ranks average dataset/direction scores and ties.
*Source-pretrained frozen encoder with random-forest readout; seg4 pools four segments. MOMENT uses released weights. LP/FT: linear probe/full fine-tuning. Details: Section A.6.

The improvement is broad rather than concentrated in one favorable target. HALO leads all six ordered position transfers, ranks first on $9/13$ held-out datasets, and improves over its anchor on $12/13$ datasets (Tables 21 and 22). Recognition establishes the downstream benefit; we next remove the random forest and ask whether the benefit is already present in the geometry of the frozen representation.

6 How HALO Changes the Frozen Representation

6.1 Transfer without a task classifier

HALO preserves activity neighborhoods even without a task classifier. We retrieve source windows for each target query using source-fitted scaling and cosine similarity; every representation sees the same queries and source gallery. Success therefore reflects the geometry of the frozen embedding rather than the capacity of a downstream readout (Section D).

The fixed z_1 representation reaches $56.2/60.7$ Recall@1 across positions/datasets, compared with $46.4/52.1$ for MantisV2 (Table 2 and Fig. 2). It scores higher in $24/28$ position units and $12/13$ held-out datasets. The advantage also extends beyond the nearest match: AP@100 rises from MantisV2's $27.5/28.7$ to $40.1/40.2$ across the two protocols (Section D).

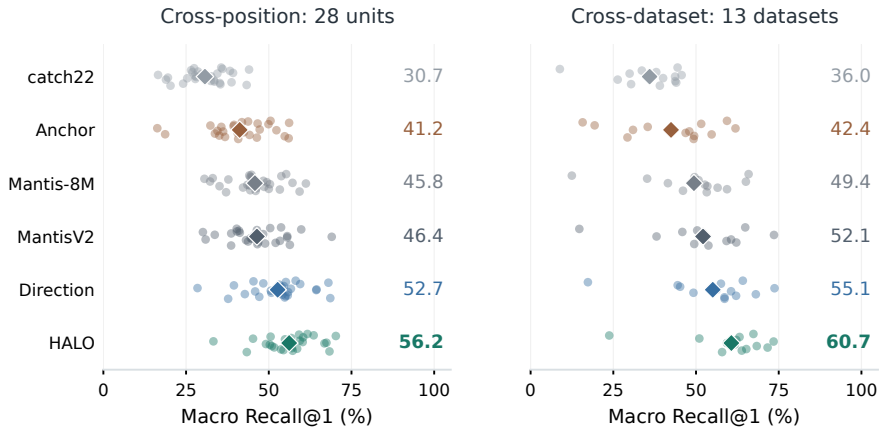

Figure 2: **HALO preserves activity neighborhoods across unseen conditions.** Dots are held-out evaluation units; diamonds and labels show unit means. Each representation uses the same target queries and source galleries within a protocol.

6.2 From statistical reference to learned refinement

The representation improves in three steps: a stable reference, a learned direction, and their bounded composition. The anchor a reaches $41.2/42.4$ Recall@1 across positions/datasets, already ahead of CATCH22's $30.7/36.0$. The unit proposal direction $\widehat{u}$ rises to $52.7/55.1$, showing that synthetic pretraining adds structure that transfers to real activity windows. Combining the two yields the strongest result, $56.2/60.7$ for z_1 (Table 2). All three representations have 129 coordinates per triad and use the same source-only retrieval processing; the gain therefore reflects how their information is organized, not a wider embedding.

Table 2: **Retrieval isolates the anchor, learned direction, and their composition.** Macro Recall@1 over 28 cross-position units or 13 held-out datasets, without a classifier. **(a)** HALO components and the strongest external comparator. **(b)** Donor interventions preserve each window's anchor and departure budget; parentheses show the drop. Within-class donors use activity labels for diagnosis only.

(a) Retrieval and decomposition

Representation	Position	Dataset
MantisV2	46.4	52.1
Anchor a	41.2	42.4
Direction $\widehat{u}$	52.7	55.1
Composition z_1	**56.2**	**60.7**

(b) Direction intervention at $\rho = 1$

Intervention	Position	Dataset
Original z_1	56.2	60.7
Direction shuffled	40.3 (-15.9)	40.7 (-20.0)
Norm shuffled	56.2 (-0.0)	60.4 (-0.3)
Within-class shuffled	52.4 (-3.7)	56.8 (-3.9)

The two components remain complementary at the level of individual transfers. Against whichever component performs better in each unit, z_1 gains $3.4/5.6$ Recall@1 points and wins in $24/28$ position units and $11/13$ datasets. Source-trained linear probes show the same pattern: z_1 reaches $56.3/65.0$ macro-F1, compared with $55.0/59.8$ for $\widehat{u}$ and $47.1/57.2$ for a. The composition therefore improves both unsupervised neighborhoods and source-trained class boundaries.

The learned direction is useful because it is matched to the input. Replacing it with a direction from another window lowers Recall@1 by $15.9/20.0$ points. Even donors from the same activity lower Recall@1 by $3.7/3.9$ points, indicating that the correction retains information beyond the class label. Shuffling only the raw proposal norm changes Recall@1 by $0.0/0.3$ points because the bound is active for 98.7% of triads at $\rho = 1$; in this regime the norm cancels and the normalized direction determines the correction (Section D.1).

6.3 Reducing position shift while preserving activity structure

The correction changes the geometry in the direction required for position transfer. On windows recorded simultaneously at two body sites, we compare the paired-site distance with the distance between activities; lower nuisance-to-signal ratio (NSR) means that position changes move a sample less than activity changes. The anchor has mean NSR 0.796, while z_1 lowers it to 0.655 and improves all 16 dataset/site-pair cells (Table 18). Thus HALO does more than separate activities in aggregate: it makes the same motion more stable across sensor locations.

7 What Makes the Refinement Work?

7.1 Bounded composition, not feature accumulation

The transfer gain comes from composing the anchor and neural proposal in a shared, bounded space. Using the learned branches alone reaches $50.8/58.4$ cross-position/dataset macro-F1, and directly concatenating them with the anchor reaches $50.2/59.0$. An unbounded correction improves these transfer scores to $52.1/60.5$, but the bounded z_1 reaches $58.8/65.7$ (Table 3). The comparison separates HALO's design from simply appending features or adding an unconstrained residual. Across these separately trained variants, bounded composition delivers the strongest transfer scores.

Table 3: **Structured composition and bounding.** Macro-F1 under three shifts. Rows report their corresponding training runs; z_1 uses the canonical $\rho = 1$ checkpoint, and "No bound" trains $a + u$.

Configuration	Cross-subject	Cross-position	Cross-dataset
Anchor a alone	83.9	49.1	59.5
Learned branches $[h_{\text{time}} \mid h_{\text{freq}}]$	83.3	50.8	58.4
Direct concatenation $[a \mid h_{\text{time}} \mid h_{\text{freq}}]$	84.2	50.2	59.0
No bound (trained with $z = a + u$)	81.9	52.1	60.5
HALO z_1	83.4	**58.8**	**65.7**
HALO $[z_{0.2} \mid z_1 \mid z_{10}]$	**84.9**	56.3	63.4

7.2 How departure and pretraining shape transfer

The departure radius reveals how much each shift benefits from the statistical prior. From one frozen checkpoint, increasing ρ from 0.2 to 10 raises cross-position macro-F1 from 50.9 to 59.1 and cross-dataset macro-F1 from 61.0 to 66.5, while cross-subject macro-F1 falls from 84.4 to 82.2 (Table 4). Staying near the anchor favors transfer within a dataset; larger departures help when the sensor position or dataset changes. Tiling z_1 to the width of the three-radius stack does not recover its cross-subject gain, so the stack contributes distinct views rather than repeated coordinates. The headline stack is fixed across protocols, whereas the radius sweep is a diagnostic of this prior–adaptation trade-off.

HAR-CauKer transfers primarily by varying acquisition while preserving the underlying motion. Removing acquisition corruptions causes the largest cross-dataset loss, 4.82 points (Table 5). Keeping only contrastive supervision on the same synthetic corpus loses 2.50 points, while removing the synthetic-class target or cross-sensor coupling costs 1.35 or 1.30 points. Factor targets and rigid-body rendering provide smaller gains of 0.86 and 0.83 points, comparable to the observed pretraining-seed variation. The ablation thus identifies nuisance diversity as the main driver, with structured motion supervision providing additional, smaller benefits (Table 13).

Table 4: **Departure budget.** Macro-F1 from one frozen checkpoint; position uses the 28-unit mean. A single radius has 129 coordinates per triad; tiling and the fixed stack have 387. Full sweep: Table 19.

Readout	Subject	Position	Dataset
$\rho = 0.2$	84.4	50.9	61.0
$\rho = 1$	83.4	58.8	65.7
$\rho = 3$	82.3	56.3	63.4
$\rho = 10$	82.2	**59.1**	**66.5**
z_1 tiled 3×	83.3	58.3	65.4
Fixed stack	**84.9**	56.5	63.4

Table 5: **HAR-CauKer ablations.** Change in cross-dataset macro-F1 (points) relative to the matched full objective or generator. Each arm uses three pretraining seeds; full means and variation are in Table 13.

Removed or changed	Δ F1
Simulator-factor targets	−0.86
Synthetic-class target	−1.35
Contrastive objective only	−2.50
Rigid-body rendering	−0.83
Cross-sensor coupling	−1.30
Acquisition corruptions	**−4.82**

7.3 Efficiency

HALO's transfer gains come from a 0.16M-parameter encoder, roughly 26× smaller than MantisV2. Its neural operations use 2.87 MMACs, and the full encoder takes 4.65 ms per six-channel window on the evaluation CPU, versus 871.37 MMACs and 19.22 ms for MantisV2 (Table 6). All three radii share one encoder pass. Anchor extraction accounts for 2.96 ms, about 64% of HALO's measured latency, and is the clear target for further speedups. Readout costs appear in Section F.

Table 6: **Frozen encoder cost per six-channel window.** CPU uses one thread at batch 1; GPU uses batch 256. MMACs count neural operations; latency includes statistical extraction. Readout costs are excluded, and all three HALO radii share one encoder pass.

Representation	Params (M)	Size (MB)	MMACs	CPU (ms) ↓	GPU (μs) ↓
Mantis-8M	8.10	30.95	1612.19	27.38	326.7
MantisV2	4.19	17.98	871.37	19.22	237.5
HALO, three radii	**0.16**	**0.61**	**2.87**	**4.65**	**31.1**
Anchor extraction only	0	0	–	2.96	29.6

8 Discussion and Conclusion

HALO treats signal statistics as a reference and learns an input-dependent correction whose departure is explicitly bounded. In aggregate, the anchor transfers on its own, the learned direction carries additional activity structure, and their composition improves both classifier-free retrieval and source-trained recognition across acquisition shifts. These results support refining a useful statistical representation rather than replacing it.

The radius sweep shows why controllable departure matters. The training-radius z_1 provides a fixed representation probe; a prespecified multi-radius stack leads cross-subject recognition, while looser single-radius readouts score higher on position and dataset transfer. Choosing a radius for a new target remains a source-validation problem, and saturation at the training radius leaves raw proposal norms at larger radii less calibrated.

The same design also reveals its boundary: when the anchor itself fails, a reference-relative correction may inherit that weakness. HALO reaches 17.5 macro-F1 on KU-HAR versus 66.9 for SimMTM, and 14.8 on HARSense versus 33.0 for the source-pretrained RelCon variant. These cases motivate diagnostics of anchor quality, sensor units, and preprocessing. Statistical extraction also dominates HALO's CPU latency, and wearable energy remains unmeasured. Better source-only radius selection and cheaper anchor computation are practical next steps.

Reproducibility statement

Sections B and 3 describe the feature construction, projection, and pretraining recipe; Sections A and 4 describe the evaluation protocols and datasets. Sections D and E provide full classifier scores and retrieval metrics. The intended release includes the synthetic renderer, anchor enumeration, evaluation harness and folds, frozen checkpoint, retrieval galleries, and scripts for regenerating tables and figures. Classifier benchmarks, classifier-free retrieval, and diagnostic direction interventions are identified separately to avoid conflating their evidence.

Ethics statement

This study uses public activity-recognition datasets and involves no new data collection. Wearable recognition can support monitoring and assistance but can also enable behavioral surveillance. Transfer performance does not establish fairness across demographic groups, and we do not evaluate such fairness or privacy protection here. Deployment would require attention to consent, data access, and the consequences of incorrect activity predictions.

AI use statement

Generative AI tools assisted with manuscript restructuring and language editing, interpretation and consistency checks of reported results, refinement of mathematical statements, feedback on experimental methodology, and figure-generation code. AI coding assistance was also used in experimental implementation and analysis; reported numerical outcomes come from executed code and saved experiment records. The authors are responsible for verifying the experimental records, mathematical claims, citations, and final AI-assisted content before submission.

References

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *ESANN*, pp. 437–442, 2013.

Oresti Banos, Rafael Garcia, Juan A Holgado-Terriza, Miguel Damas, Hector Pomares, Ignacio Rojas, Alejandro Saez, and Claudia Villalonga. mHealthDroid: a novel framework for agile development of mobile health applications. In *International Workshop on Ambient Assisted Living*, pp. 91–98. Springer, 2014.

Ling Bao and Stephen S. Intille. Activity recognition from user-annotated acceleration data. In *Pervasive Computing*, pp. 1–17. Springer, 2004. doi: 10.1007/978-3-540-24646-6_1.

Billur Barshan and Murat Cihan Yüksek. Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units. *The Computer Journal*, 57(11): 1649–1667, 2014.

Sizhen Bian, Mengxi Liu, Vitor Fortes Rey, Daniel Geissler, and Paul Lukowicz. TinierHAR: Towards ultra-lightweight deep learning models for efficient human activity recognition on edge devices. In *Proceedings of the 2025 ACM International Symposium on Wearable Computers (ISWC)*, 2025. doi: 10.1145/3715071.3750410.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Andreas Bulling, Ulf Blanke, and Bernt Schiele. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys*, 46(3):1–33, 2014.

Yize Cai, Rui Feng, Anlan Yu, Baoshen Guo, and Zhiqing Hong. BenchHAR: Benchmarking self-supervised learning for generalizable sensor-based activity recognition, 2026. URL https://arxiv.org/abs/2605.08296. arXiv:2605.08296.

Nurul Amin Choudhury, Soumen Moulik, and Diptendu Sinha Roy. HARSense: Statistical human activity recognition dataset. IEEE Dataport, 2021. DOI: 10.21227/9pt3-2m34.

Maximilian Christ, Nils Braun, Julius Neuffer, and Andreas W. Kempa-Liehr. Time series FeatuRe extraction on basis of scalable hypothesis tests (tsfresh – a Python package). *Neurocomputing*, 307:72–77, 2018.

Gaole Dai, Huatao Xu, Hyungjun Yoon, Mo Li, Rui Tan, and Sung-Ju Lee. Contrastsense: Domain-invariant contrastive learning for in-the-wild wearable sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(4):1–32, 2024.

Angus Dempster, François Petitjean, and Geoffrey I Webb. Rocket: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, 34(5):1454–1495, 2020.

Angus Dempster, Daniel F. Schmidt, and Geoffrey I. Webb. MiniRocket: A very fast (almost) deterministic transform for time series classification. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 248–257, 2021.

Jiaxiang Dong, Haixu Wu, Haoran Zhang, Li Zhang, Jianmin Wang, and Mingsheng Long. SimMTM: A simple pre-training framework for masked time-series modeling. In *Advances in Neural Information Processing Systems*, volume 36, pp. 29996–30025, 2023.

Samuel Dooley, Gurnoor Singh Khurana, Chirag Mohapatra, Siddartha Naidu, and Colin White. Forecastpfn: Synthetically-trained zero-shot forecasting, 2023. URL https://arxiv.org/abs/2311.01933.

Yu Enokibori. rTsfNet: A DNN model with multi-head 3D rotation and time series feature extraction for IMU-based human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(4):202:1–202:26, 2024. doi: 10.1145/3699733.

Vasilii Feofanov, Songkang Wen, Marius Alonso, Romain Ilbert, Hongbo Guo, Malik Tiomoko, Lujia Pan, Jianfeng Zhang, and Ievgen Redko. Mantis: Lightweight calibrated foundation model for user-friendly time series classification. *arXiv preprint arXiv:2502.15637v1*, 2025. URL https://arxiv.org/abs/2502.15637v1.

Vasilii Feofanov, Songkang Wen, Jianfeng Zhang, Lujia Pan, and Ievgen Redko. MantisV2: Closing the zero-shot gap in time series classification with synthetic data and test-time strategies, 2026. URL https://arxiv.org/abs/2602.17868. arXiv:2602.17868.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 16115–16152. PMLR, 2024. URL https://proceedings.mlr.press/v235/goswami24a.html.

Nils Y. Hammerla, Shane Halloran, and Thomas Plötz. Deep, convolutional, and recurrent models for human activity recognition using wearables. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 1533–1540, 2016.

Harish Haresamudram, Irfan Essa, and Thomas Plötz. Contrastive predictive coding for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(2):65:1–65:26, 2021. doi: 10.1145/3463506.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

Zhiqing Hong, Zelong Li, Shuxin Zhong, Wenjun Lyu, Haotian Wang, Yi Ding, Tian He, and Desheng Zhang. Crosshar: Generalizing cross-dataset human activity recognition via hierarchical self-supervised pretraining. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(2):1–26, 2024.

Simon A. Lee, Cyrus Tanade, Hao Zhou, Juhyeon Lee, Megha Thukral, Md. Sazzad Hissain Khan, Keum San Chun, Baiying Lu, Migyeong Gwak, Mehrab Bin Morshed, Viswam Nathan, Mahbubur Rahman, Li Zhu, Subramaniam Venkatraman, and Sharanya Desai. HiMAE: Hierarchical masked autoencoders discover resolution-specific structure in wearable time series. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=iPAy5VpGQa.

Zikang Leng, Amitrajit Bhattacharjee, Hrudhai Rajasekhar, Lizhe Zhang, Elizabeth Bruda, Hyeokhyen Kwon, and Thomas Plötz. Imugpt 2.0: Language-based cross modality transfer for sensor-based human activity recognition. *arXiv preprint arXiv:2402.01049*, 2024.

Shengzhong Liu, Tomoyoshi Kimura, Dongxin Liu, Ruijie Wang, Jinyang Li, Suhas Diggavi, Mani Srivastava, and Tarek Abdelzaher. FOCAL: Contrastive learning for multimodal time-series sensing signals in factorized orthogonal latent space. In *Advances in Neural Information Processing Systems*, volume 36, pp. 47309–47338, 2023.

Carl H Lubba, Sarab S Sethi, Philip Knaute, Simon R Schultz, Ben D Fulcher, and Nick S Jones. catch22: CAnonical time-series CHaracteristics. *Data Mining and Knowledge Discovery*, 33(6): 1821–1852, 2019.

Saif Mahmud, M Tanjid Hasan Tonmoy, Kishor Kumar Bhaumik, A K M Mahbubur Rahman, M Ashraful Amin, Mohammad Shoyaib, Muhammad Asif Hossain Khan, and Amin Ahsan Ali. Human activity recognition from wearable sensor data using self-attention. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI)*, 2020.

Mohammad Malekzadeh, Richard G Clegg, Andrea Cavallaro, and Hamed Haddadi. Mobile sensor data anonymization. In *Proceedings of the International Conference on Internet of Things Design and Implementation*, pp. 49–58, 2019.

Girish Narayanswamy, Xin Liu, Kumar Ayush, Yuzhe Yang, Xuhai Xu, Shun Liao, Jake Garrison, Shyam A. Tailor, Jacob Sunshine, Yun Liu, Tim Althoff, Shrikanth Narayanan, Pushmeet Kohli, Jiening Zhan, Mark Malhotra, Shwetak Patel, Samy Abdel-Ghaffar, and Daniel McDuff. Scaling wearable foundation models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Francisco Javier Ordóñez and Daniel Roggen. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1):115, 2016.

Stephen J. Preece, John Yannis Goulermas, Laurence P. J. Kenney, and David Howard. A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data. *IEEE Transactions on Biomedical Engineering*, 56(3):871–879, 2009.

Attila Reiss and Didier Stricker. Introducing a new benchmarked dataset for activity monitoring. In *16th International Symposium on Wearable Computers*, pp. 108–109. IEEE, 2012.

Jorge Reyes-Ortiz, Davide Anguita, Luca Oneto, and Xavier Parra. Smartphone-based recognition of human activities and postural transitions. UCI Machine Learning Repository, 2015.

Muhammad Shoaib, Stephan Bosch, Ozlem Durmaz Incel, Hans Scholten, and Paul JM Havinga. Fusion of smartphone motion sensors for physical activity recognition. *Sensors*, 14(6):10146–10176, 2014.

Muhammad Shoaib, Stephan Bosch, Ozlem Durmaz Incel, Hans Scholten, and Paul J. M. Havinga. Complex human activity recognition using smartphone and wrist-worn motion sensors. *Sensors*, 16(4):426, 2016. doi: 10.3390/s16040426. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC4850940/.

Niloy Sikder and Abdullah-Al Nahid. KU-HAR: An open dataset for heterogeneous human activity recognition. *Pattern Recognition Letters*, 146:46–54, 2021.

Allan Stisen, Henrik Blunck, Sourav Bhattacharya, Thor Siiger Prentow, Mikkel Baun Kjærgaard, Anind Dey, Tobias Sonne, and Mads Møller Jensen. Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM conference on embedded networked sensor systems*, pp. 127–140, 2015.

Timo Sztyler and Heiner Stuckenschmidt. On-body localization of wearable devices: An investigation of position-aware activity recognition. In *2016 IEEE international conference on pervasive computing and communications (PerCom)*, pp. 1–9. IEEE, 2016.

Gary Weiss. WISDM smartphone and smartwatch activity and biometrics dataset. UCI Machine Learning Repository, 2019. DOI: 10.24432/C5HK59.

Shifeng Xie, Vasilii Feofanov, Jianfeng Zhang, Themis Palpanas, and Ievgen Redko. CauKer: Classification time series foundation models can be pretrained on synthetic data. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/702b67152ec4435795f681865b67999c-Abstract-Conference.html.

Huatao Xu, Pengfei Zhou, Rui Tan, Mo Li, and Guobin Shen. LIMU-BERT: Unleashing the potential of unlabeled data for IMU sensing applications. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, pp. 220–233, 2021.

Maxwell A Xu, Jaya Narain, Gregory Darnell, Haraldur T Hallgrimsson, Hyewon Jeong, Darren Forde, Richard Andres Fineman, Karthik Jayaraman Raghuram, James Matthew Rehg, and Shirley You Ren. RelCon: Relative contrastive learning for a motion foundation model for wearable data. In *The Thirteenth International Conference on Learning Representations*, 2025.

Yan Yan, Dali Chen, Yushi Liu, Jinjin Zhao, Bo Wang, Xuankun Wu, Xiaohao Jiao, Yuqian Chen, Huihui Li, and Xuchao Ren. TNDA-HAR. IEEE Dataport, 2021. DOI: 10.21227/4epb-pg26.

Jianbo Yang, Minh Nhut Nguyen, Phyo Phyo San, Xiaoli Li, and Shonali Krishnaswamy. Deep convolutional neural networks on multichannel time series for human activity recognition. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3995–4001, 2015.

Alexander D Young, Martin J Ling, and Damal K Arvind. Imusim: A simulation environment for inertial sensing algorithm design and evaluation. In *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pp. 199–210. IEEE, 2011.

Hang Yuan, Shing Chan, Andrew P Creagh, Catherine Tong, Aidan Acquah, David A Clifton, and Aiden Doherty. Self-supervised learning for human activity recognition using 700,000 person-days of wearable data. *npj Digital Medicine*, 7(1):91, 2024.

Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 8980–8987, 2022.

Mi Zhang and Alexander A Sawchuk. USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pp. 1036–1043, 2012.

Wenrui Zhang, Ling Yang, Shijia Geng, and Shenda Hong. Self-supervised time series representation learning via cross reconstruction transformer. *IEEE Transactions on Neural Networks and Learning Systems*, 35(11):16129–16138, 2024a. doi: 10.1109/TNNLS.2023.3292066.

Xiyuan Zhang, Diyan Teng, Ranak Roy Chowdhury, Shuheng Li, Dezhi Hong, Rajesh K. Gupta, and Jingbo Shang. UniMTS: Unified pre-training for motion time series. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b.

Yuwei Zhang, Kumar Ayush, Siyuan Qiao, A. Ali Heydari, Girish Narayanswamy, Max Xu, Ahmed Metwally, Jinhua Xu, Jake Garrison, Xuhai Xu, Tim Althoff, Yun Liu, Pushmeet Kohli, Jiening Zhan, Mark Malhotra, Shwetak Patel, Cecilia Mascolo, Xin Liu, Daniel McDuff, and Yuzhe Yang. SensorLM: Learning the language of wearable sensors. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1559. URL https://proceedings.nips.cc/paper_files/paper/2025/hash/42cd98f0e7520d4a63c34891ac1c972f-Abstract-Conference.html.

Yexu Zhou, Haibin Zhao, Yiran Huang, Till Riedel, Michael Hefenbrock, and Michael Beigl. TinyHAR: A lightweight deep learning model designed for human activity recognition. In *Proceedings of the 2022 ACM International Symposium on Wearable Computers*, pp. 89–93, 2022. doi: 10.1145/3544794.3558467.

A BENCHMARK COVERAGE, DATASETS, AND EVALUATION PROTOCOLS

The evaluation suite covers 14 public datasets through two complementary data builds: a dataset-specific build for cross-subject recognition and a harmonized build for cross-position and cross-dataset transfer. The inventories below are exported from the benchmark manifest dated September 13, 2026. They describe the stored data used by the evaluation pipelines, rather than the sample counts of the original publications. Window counts include overlapping windows and are not counts of independent recordings.

A.1 COVERAGE ACROSS DATA BUILDS AND GENERALIZATION AXES

The thirteen cross-subject datasets are DSADS (Barshan & Yüksek, 2014), HARSense (Choudhury et al., 2021), HHAR (Stisen et al., 2015), KU-HAR (Sikder & Nahid, 2021), MHEALTH (Banos et al., 2014), MotionSense (Malekzadeh et al., 2019), PAMAP2 (Reiss & Stricker, 2012), RealWorld (Sztyler & Stuckenschmidt, 2016), Shoaib (Shoaib et al., 2014), TNDA-HAR (Yan et al., 2021), UCI-HAR/HAPT (Anguita et al., 2013; Reyes-Ortiz et al., 2015), USC-HAD (Zhang & Sawchuk, 2012), and WISDM (Weiss, 2019). UT-Complex (Shoaib et al., 2016) is additionally included in the pooled cross-position protocol. The inventory keys `Motion` and `UCI` correspond to MotionSense and the UCI HAR/HAPT data family, respectively.

The dataset-specific inventory contains 448,881 windows in total; excluding the 15,574 UT-Complex windows leaves 433,307 windows for cross-subject evaluation. This configuration covers 6–19 classes and 8–89 recorded subject IDs per evaluated dataset, with 20–100 Hz sampling configurations and 6–45 stored channels. It retains richer activity taxonomies and sensor channels where available. The harmonized inventory contains 173,483 windows across fourteen datasets, all stored at 20 Hz with 120 samples and six channels. Excluding UT-Complex yields the 169,883-window, thirteen-dataset LODO pool. The two builds reuse underlying recordings and must not be added to obtain a unique-data total.

Table 7: **Complementary benchmark protocols.** The window column describes the available pool, not the training size of each fold. Evaluation units share data and are not independent repetitions.

Protocol	Datasets	Pool windows	Held-out units
Cross-subject	13	433,307	Five subject folds per dataset: 65 folds
Cross-position	14	173,483	Six ordered body-group directions with subject holdout: 28 units
Cross-dataset	13	169,883	One held-out dataset, twelve source datasets: 13 units

A.2 DATASET-SPECIFIC WINDOWS, CHANNELS, AND ACQUISITION DETAILS

Table 8 lists the dataset-specific build. Stored channels are distinguished from encoder features: HALO processes supported three-axis groups, while non-triad auxiliary channels do not automatically enter its representation. A single position ID also does not imply a single sensor. DSADS, MHEALTH, and PAMAP2 store several body sites as channel blocks inside a window; other multi-position datasets emit positions as separate windows.

DSADS and HARSense. DSADS uses non-overlapping five-second windows at 25 Hz. Its 45 channels are acceleration, gyroscope, and magnetometer triads at the torso, both arms, and both legs, stored jointly. The 19 classes include exercise equipment, cycling, rowing, and basketball as well as basic postures and locomotion. HARSense retains 16 phone channels at 50 Hz, including acceleration with gravity, linear acceleration, gravity, gyroscope, and rotation-vector fields. It retains sleeping as a seventh activity; six-second windows use a three-second hop within contiguous same-activity runs.

HHAR, KU-HAR, and MotionSense. HHAR has heterogeneous devices and irregular timestamps. Its dataset-specific build aligns accelerometer and gyroscope streams to a uniform 100 Hz grid within each user/device/activity run using the device timestamps; 100 Hz is therefore a build

Table 8: **Dataset-specific build inventory.** Counts refer to stored windows, not distinct recordings. S: recorded subject IDs; K: activity classes; f: Hz; L: samples/window; h: hop in seconds; C: stored channels. Blocks denote body sites stored together in one window; IDs denote separate rows. †: UT-Complex is included only in pooled cross-position evaluation. The thirteen evaluated cross-subject datasets contain 433,307 windows.

Dataset	Windows	S	K	f	L	h	C	Site organization
DSADS	9,120	8	19	25	125	5	45	5 blocks
HARSense	518	12	7	50	300	3	16	1 ID
HHAR	30,498	9	6	100	600	3	6	2 IDs
KU-HAR	18,092	89	18	100	600	3	6	1 ID
MHEALTH	2,102	10	12	50	300	3	23	3 blocks
MotionSense	8,884	24	6	50	300	3	6	1 ID
PAMAP2	8,893	9	18	100	600	3	36	3 blocks
RealWorld	147,664	15	8	50	300	3	6	7 IDs
Shoaib	20,355	10	7	50	300	3	12	5 IDs
TNDA-HAR	59,200	50	8	50	300	3	9	5 IDs
UCI (HAR/HAPT)	23,693	30	12	50	64	0.64	6	1 ID
USC-HAD	8,216	14	12	100	600	3	6	1 ID
UT-Complex†	15,574	1	13	50	300	3	12	2 IDs
WISDM	96,072	51	18	20	120	3	6	2 IDs
Stored total	448,881							

configuration, not a claim that every device samples natively at that rate. KU-HAR retains 18 classes and 89 recorded subjects, with six-second windows at 100 Hz and no window crossing a trial boundary. MotionSense uses the six available accelerometer/gyroscope channels at 50 Hz, with six-second windows and a three-second hop within subject/activity/trial files. Additional device-motion fields requested in the initial build specification were unavailable in the provided export and were not fabricated.

MHEALTH and PAMAP2. MHEALTH stores 23 channels at 50 Hz: chest acceleration, two ECG leads, and acceleration, gyroscope, and magnetometer channels at the left ankle and right arm. The inertial triad extractor excludes the two ECG leads. Twelve activity classes are retained after dropping the null class, with six-second windows and 50% overlap. PAMAP2 retains 36 channels from the hand, chest, and ankle: two accelerometer ranges, gyroscope, and magnetometer triads per site. The build preserves 18 classes and uses six-second windows at 100 Hz with 50% overlap. Temperature, invalid orientation fields, and the sparsely sampled heart-rate channel are excluded. Missing values are interpolated within each segment and channel, and windows do not cross activity, subject, or session boundaries.

RealWorld, Shoaib, and TNDA-HAR. RealWorld retains eight classes and seven separately windowed positions at 50 Hz. Its accelerometer and gyroscope streams are stacked into six-channel windows. Shoaib retains seven classes, including biking, and 12 channels per position across acceleration, linear acceleration, gyroscope, and magnetometer. TNDA-HAR retains eight classes, including cycling, and nine channels across five positions. All three emit positions as separate six-second windows with a three-second hop within continuous runs.

UCI HAR/HAPT and USC-HAD. The dataset-specific `UCI` build includes six basic activities and six short postural transitions from the HAPT extension (Reyes-Ortiz et al., 2015). To retain these transition segments, it uses 1.28-second windows (64 samples at 50 Hz) and a 0.64-second hop. This is our windowing configuration, distinct from the original published fixed-window configuration. USC-HAD retains twelve activities and six front-right-hip channels at 100 Hz, with six-second windows and 50% overlap inside each trial. Its stored acceleration and gyroscope units remain those of the source data; missing values are interpolated within the build segments.

WISDM and UT-Complex. WISDM retains eighteen activity classes, 51 subjects, and two position IDs with six channels at 20 Hz. The activity set includes hand interactions such as typing,

eating, writing, clapping, and folding clothes. UT-Complex stores thirteen classes and two position IDs with twelve channels. Under the benchmark protocol, UT-Complex contributes only to the pooled cross-position evaluation and is excluded from cross-subject and LODO evaluation.

A.3 HARMONIZED TRANSFER DATA AND CLASS COVERAGE

The transfer build uses six-second windows at 20 Hz and a common eight-label vocabulary: sitting, standing, lying, upstairs, downstairs, walking, running, and jumping. Each dataset contains only the applicable subset, ranging from five to eight classes (Tables 9 and 10). Dataset-specific class counts and window counts differ from the richer build because of label selection, sensor selection, and windowing. The same nominal activity vocabulary therefore coexists with substantial class imbalance and heterogeneous sensor placements.

Table 9: **Transfer-build dataset and placement inventory.** Every window has 120 samples at 20 Hz and six stored channels. S and K are subject-ID and class counts. Positions are the exported names/IDs, not a new anatomical annotation. UT-Complex (†) participates only in the pooled position protocol.

Dataset	Windows	S	K	Exported positions (ID)
DSADS	16,000	8	8	chest (0); wrist (1); knee (3)
HARSense	270	12	6	position (100)
HHAR	9,175	9	5	waist (8)
KU-HAR	4,339	80	8	waist (8)
MHEALTH	1,452	10	7	wrist (1); ankle (5)
MotionSense	4,534	24	6	pocket (7)
PAMAP2	5,796	9	8	chest (0); ankle (5); hand (6)
RealWorld	74,040	15	8	seven exported position IDs (0, 1, 2, 4, 8, 9, 11)
Shoaib	8,850	10	6	wrist (1); pocket (7); waist (8); upperarm (11)
TNDA-HAR	25,915	50	7	wrist (1); thigh (2); ankle (5); upperarm (11); back (12)
UCI (HAR/HAPT)	2,088	30	6	waist (8)
USC-HAD	3,039	14	8	pocket (7)
UT-Complex†	3,600	1	6	wrist (1); pocket (7)
WISDM	14,385	51	5	wrist (1); pocket (7)
Position pool	173,483			Fourteen datasets
LODO pool	169,883			Thirteen datasets; excludes UT-Complex

Encoder adaptation and numerical scales. HALO linearly resamples each stored transfer triad from 120 to 128 samples before feature computation; the six-second physical window remains unchanged. The dataset-specific build is adapted to the same encoder input length from its own stored resolution. Input units are not converted to a common physical scale. HALO separates per-window shape and scale internally; retrieval then fits coordinate-wise location and scale on the source gallery only. The different builds are complementary evaluation settings, not a calibrated progression in the amount of distribution shift. Shape normalization does not make the complete representation invariant to sensor units: for positive gain c above the scale floor, $\log s(cx) = \log s(x) + \log c$. This coordinate affects both the proposal input and the anchor-relative budget after frozen BatchNorm. The evaluated pipeline neither clips it nor excludes it from the budget. Unit augmentation provides finite training variation, not protection against arbitrarily uncalibrated sensor gains.

Table 10: **Activity coverage and imbalance in the transfer build.** Entries are stored window counts. Up/Down denote upstairs/downstairs. A dash indicates an activity absent from a dataset; the shared taxonomy does not imply that all eight classes occur in every dataset.

Dataset	Sit	Stand	Lie	Up	Down	Walk	Run	Jump
DSADS	2,000	2,000	2,000	2,000	2,000	2,000	2,000	2,000
HARSense	60	36	–	31	21	78	44	–
HHAR	2,160	1,751	–	1,665	1,259	2,340	–	–
KU-HAR	889	898	872	396	380	348	257	299
MHEALTH	200	200	200	192	–	200	400	60
MotionSense	1,105	999	–	490	402	1,112	426	–
PAMAP2	915	936	951	573	513	1,185	486	237
RealWorld	11,176	11,048	11,196	8,468	6,814	11,270	12,438	1,630
Shoaib	1,500	1,500	–	1,350	1,500	1,500	1,500	–
TNDA-HAR	3,880	3,760	3,870	3,245	3,250	4,010	3,900	–
UCI (HAR/HAPT)	362	398	401	318	254	355	–	–
USC-HAD	412	372	600	325	301	608	271	150
UT-Complex†	600	600	–	600	600	600	600	–
WISDM	3,062	2,959	–	2,825	–	2,780	2,759	–
Total	28,321	27,457	20,090	22,478	17,294	28,386	25,081	4,376

A.4 Source–target partitions, readouts, and aggregation

Cross-subject evaluation. Each of the thirteen eligible datasets uses five subject-grouped folds, yielding 65 folds. All windows assigned to a subject stay on the same side of a fold, including windows from different positions and overlapping windows. Models are trained or readouts fitted only on source subjects. Scores are averaged within datasets and then across datasets. UT-Complex is excluded because the build cannot support this grouping requirement.

Cross-position evaluation. Global subject IDs in the pooled transfer build are sorted and divided into five folds. For each ordered pair of the three body groups, source examples come from training subjects at source positions, and target examples from held-out subjects at target positions. Both subject identity and sensor-group membership therefore change. The six ordered directions would give 30 units; fold 4 has no trunk target windows, so its two directions targeting trunk are absent for every method, leaving 28 shared units. The evaluator omits source/target pools with fewer than 50 windows and source pools with fewer than two activities. This protocol uses the three body-location groups studied by Cai et al. (2026); the joint subject holdout, unit filtering, and resulting 28 evaluation units are defined for our build above. The exported position-ID groups are fixed by the evaluator: trunk $= \{0, 8, 9, 12\}$, upper limb $= \{1, 6, 11\}$, and lower limb $= \{2, 3, 4, 5, 7\}$. IDs outside these sets, including HARSense's 100, do not enter position-transfer units; HARSense remains eligible for the other protocols. These are benchmark ID assignments, not a new mapping from native anatomical names. In particular, pocket ID 7 is assigned to lower limb, and RealWorld IDs 4 and 9 enter lower limb and trunk, respectively. The unstratified subject partition preserves the shared evaluator's splits but leaves uneven placement coverage; it is not a balanced anatomical sampling design.

Cross-dataset evaluation. Each of the thirteen datasets is held out in turn. Its source is the union of the other twelve, and no target-dataset windows are used to fit the classifier, distance preprocessing, or probe hyperparameters. The full eligible pool contains 169,883 windows; a particular fold's training pool is smaller by the size of its held-out dataset. This evaluates data-source transfer under the shared vocabulary, rather than recognition of unseen activity labels.

Common galleries and metrics. Retrieval uses the same queries and source gallery across representations within a unit. Galleries larger than 20,000 windows are subsampled by activity using a deterministic seed that does not depend on features. Smaller galleries are retained in full; rare source classes can leave the capped gallery below 20,000. Median/IQR scaling is fitted on the gallery, then applied to source and target before L2 normalization and cosine retrieval. Linear-probe regularization is selected on a subject-grouped source split. Recall and macro-F1 are computed over classes

present in each target unit; an absent relevant gallery class yields zero retrieval success. Recognition averages six direction means for cross-position, whereas retrieval and linear probing average the 28 units. Cross-subject and LODO recognition average dataset means. Overlapping sources and repeated target subjects across position directions limit independence; uncertainty is described in Sections 4 and D.

A.5 Full dataset-specific activity lists

Table 11 records the dataset-specific labels retained by the richer build. Keeping the original distinctions exposes finer recognition demands than the shared eight-class transfer vocabulary: postural transitions, equipment-specific exercise, household activities, and hand interactions. Similar label names in different datasets are not counted as distinct global classes or automatically merged outside the specified transfer mapping.

Table 11: **Dataset-specific activity taxonomies.** Labels follow the inventory; underscores and capitalization are normalized for readability. These are not pooled as distinct labels across datasets. †: inventoried, not evaluated in the within-dataset protocol.

Dataset	Activity labels
DSADS	sitting; standing; lying on back; lying on right side; ascending stairs; descending stairs; standing in elevator still; moving around in elevator; walking in parking lot; walking treadmill flat; walking treadmill inclined; running treadmill; exercising stepper; exercising cross trainer; cycling bike horizontal; cycling bike vertical; rowing; jumping; playing basketball.
HARSense	sitting; standing; walking; running; upstairs; downstairs; sleeping.
HHAR	biking; sitting; standing; walking; upstairs; downstairs.
KU-HAR	stand; sit; talk-sit; talk-stand; stand-sit; lay; lay-stand; pick; jump; push-up; sit-up; walk; walk-backward; walk-circle; run; stair-up; stair-down; table-tennis.
MHEALTH	standing; sitting; lying; walking; climbing stairs; waist bends forward; frontal elevation of arms; knees bending; cycling; jogging; running; jump front & back.
MotionSense	downstairs; upstairs; walking; jogging; sitting; standing.
PAMAP2	lying; sitting; standing; walking; running; cycling; nordic walking; watching tv; computer work; car driving; ascending stairs; descending stairs; vacuum cleaning; ironing; folding laundry; house cleaning; playing soccer; rope jumping.
RealWorld	climbingdown; climbingup; jumping; lying; running; sitting; standing; walking.
Shoaib	walking; sitting; standing; jogging; biking; upstairs; downstairs.
TNDA-HAR	sitting; standing; lying; upstairs; downstairs; cycling; walking; running.
UCI (HAR/HAPT)	walking; walking upstairs; walking downstairs; sitting; standing; laying; stand to sit; sit to stand; sit to lie; lie to sit; stand to lie; lie to stand.
USC-HAD	walking forward; walking left; walking right; walking upstairs; walking downstairs; running forward; jumping up; sitting; standing; sleeping; elevator up; elevator down.
UT-Complex†	walking; standing; jogging; sitting; biking; upstairs; downstairs; typing; writing; drinking coffee; talking; smoking; eating.
WISDM	walking; jogging; stairs; sitting; standing; typing; brushing teeth; eating soup; eating chips; eating pasta; drinking from cup; eating sandwich; kicking soccer ball; playing catch tennis ball; dribbling basketball; writing; clapping; folding clothes.

A.6 Method-specific downstream evaluation

For the established reference baselines, downstream classifier choices follow their original evaluations or the corresponding benchmark implementations, with the adaptations specified below. We evaluate these method-specific readouts on the shared source–target splits. Random forests read out HALO, the statistical anchors, catch22, and the two Mantis encoders; random-forest evaluation of frozen representations is also used by MantisV2 (Feofanov et al., 2026). For random-forest comparisons, we use our common setting of 300 class-balanced trees fitted on source data. MiniRocket uses ridge classification (Dempster et al., 2021), and frozen UniMTS uses a linear

probe (Zhang et al., 2024b). BioBankSSL, CRT, FOCAL, and SimMTM (Yuan et al., 2024; Zhang et al., 2024a; Liu et al., 2023; Dong et al., 2023) use the source-pretraining and classifier configurations of the corresponding transfer benchmark (Cai et al., 2026). Supervised TinyHAR, TinierHAR, MCNN, 1D CNN, DeepConvLSTM, and SA-HAR are trained from scratch on each source split (Zhou et al., 2022; Bian et al., 2025; Yang et al., 2015; Hammerla et al., 2016; Ordóñez & Roggen, 2016; Mahmud et al., 2020). UniMTS is additionally evaluated with source-data fine-tuning; its inputs are assigned to the nearest joints in its 22-joint skeleton representation. The full result tables identify availability by protocol. Released encoders retain their original pretraining and model sizes. Each transfer table reports twenty method configurations, including HALO, spanning fixed features, source-trained neural models, and frozen or fine-tuned pretrained encoders. The recognition benchmark compares transfer performance under these method-specific downstream protocols. Classifier-free retrieval and the common linear probe for HALO components (Section 6) provide complementary evidence about representation structure.

Additional frozen-feature comparisons. We add MOMENT, LIMU-BERT, LIMU-BERT (seg4), and RelCon (Goswami et al., 2024; Xu et al., 2021; 2025) across all $65/28/13$ subject/position/dataset units. All four use the common 300-tree, class-balanced random forest. Their readout implementation reproduces the reference MantisV2 scores on four subject folds, all 28 position units, and all 13 dataset units to numerical precision. Only MOMENT uses a released pretrained checkpoint. LIMU-BERT and RelCon are pretrained separately on each unit's unlabeled source windows, then frozen before fitting the forest; held-out windows do not enter their pretraining. These are method reproductions under the budgets below, not evaluations of the published large-scale RelCon model. Available LIMU-BERT checkpoints overlap the evaluation corpora, so we use source-only pretraining instead.

MOMENT input and readout. We use `AutonLab/MOMENT-1-base` with channel-wise embeddings concatenated in input order. Each channel contributes 768 coordinates, giving $4{,}608$ for the six-channel transfer build and up to $34{,}560$ for DSADS. Inputs longer than 512 samples are linearly resampled; shorter inputs are trimmed to a multiple of eight and zero-padded to 512, with an input mask marking valid samples. The reported row uses channel concatenation throughout.

LIMU-BERT and the segment-pooling control. The `base_v1` encoder processes accelerometer–gyroscope blocks at 20 Hz. Pretraining uses $10{,}000$ steps per unit, batch size 128, and Adam at learning rate 10^{-3}. Time-mean pooling yields 72 features per block; LIMU-BERT (seg4) concatenates the means of four temporal segments for 288 features per block, using the same pretrained encoder. Native-build body-site blocks are concatenated, so these widths are multiplied by the number of blocks. Missing gyroscopes are zero-filled; magnetometer, ECG, gravity, and linear-acceleration channels are excluded. The original sequence-based GRU readout is replaced here by the common forest to compare frozen features; seg4 tests sensitivity to temporal pooling.

RelCon and sensor units. RelCon uses the method's ResNet1D encoder, CrossAttn motif-distance network, and relative contrastive objective. Per unit, we run $3{,}000$ motif-training steps and $2{,}000$ representation steps with batch size 32 and 20 within-user candidates. It receives accelerometer triads only and produces 256 features per triad, concatenated across sites. The resulting model is not the original billion-segment pretrained checkpoint. For these source-pretrained IMU baselines, dataset-specific acceleration and angular-velocity units are converted before their model-specific preprocessing. Unit conventions are inferred from per-dataset signal summaries; they are distinct from source-fitted feature scaling. In the transfer build, KU-HAR, PAMAP2, and USC-HAD use acceleration conversion from g to $\mathrm{m/s^2}$ and angular velocity from degrees/s to radians/s; Shoaib requires the latter conversion. Native-build conversions are determined separately for each dataset and sensor block.

The four new configurations use one readout seed; the source-pretrained IMU methods also use one pretraining seed. Fold variation is not pretraining-seed uncertainty. Their results characterize the recorded budgets and adapters, and do not establish performance at larger pretraining scale.

A.7 Illustrative activity–placement reversal

This PAMAP2 diagnostic uses synchronized hand and ankle recordings, subject-grouped evaluation, and three random-forest seeds. Both TSFRESH (Christ et al., 2018) and frozen Mantis-8M are evaluated as full eighteen-class problems at each site; the table reports the two highlighted activities rather than a binary task.

Table 12: **An illustrative activity–placement interaction.** Per-class F1, averaged over three forest seeds, in the eighteen-class PAMAP2 case study. This separate diagnostic is not pooled with the main transfer benchmark.

Activity	Sensor site	TSFRESH	Mantis-8M
Nordic walking	Hand	97.8	91.7
Nordic walking	Ankle	54.1	66.9
Running	Hand	69.9	83.7
Running	Ankle	91.2	83.0

Subject-clustered paired bootstrap intervals for the hand-minus-ankle change in the Mantis-minus-TSFRESH F1 difference are $[-35.5, -8.1]$ points for Nordic walking and $[0.9, 52.6]$ for running. The interactions exclude zero, while individual cells do not. The example illustrates condition-dependent relative performance; it does not establish a population-wide ranking, the failure of fixed fusion, or the necessity of a particular refinement mechanism.

B Encoder and Pretraining Details

Scale split and anchor normalization. Scale and shape are defined in Eq. (1); the epsilon floor stabilizes nearly zero triads. The descriptor and neural branches use $\widetilde{x}$; $\log s(x)$ remains an anchor coordinate. The scale split commutes with rotations of the sensor axes because $s(x)$ uses the vector norm, although axis-dependent descriptor coordinates need not be rotation invariant. If $A(x)$ denotes the selected fixed statistical coordinates followed by $\log s(x)$, synthetic pretraining fits an affine-free BatchNorm and yields

$$a(x) = \mathrm{BN}_{\text{no-affine}}(A(x)) \in \mathbb{R}^{129}. \tag{4}$$

Its running statistics are frozen at evaluation; it learns no coordinate-wise scale or bias. The correction shares this standardized vector space but is not trained to reconstruct each named statistic. Its norm bound measures displacement in representation space, not physical units.

Anchor grid. The grid crosses derived signal, segment count, within-segment statistic, cross-segment aggregation, and signal domain. Derived signals are the three recorded axes, vector magnitude, median-centered magnitude, jerk magnitude, components parallel and orthogonal to the window mean direction, and the three cross-axis products. Segment counts are $P \in \{1, 2, 4, 8, 16, 32\}$. Within each segment we compute mean, standard deviation, minimum, maximum, median, MAD, RMS, zero-crossing rate, or slope. For $P > 1$, order-free aggregation uses the mean, standard deviation, minimum, maximum, $0.25/0.75$ quantiles, or a nine-scale logarithmic amplitude encoding. The same construction is applied to the waveform and to a log power spectrum with DC removed and power normalized by its mean. The order-free grid contains $10{,}098$ coordinates; the released specification is a seed-0 uniform draw of 128 coordinates. This generic draw does not use downstream labels. Replacing the descriptor or its width is a design-time change that requires retraining anchor-dependent layers; the anchor-size and draw controls are in Section C.

The time-domain branch uses a kernel-7 depthwise-separable stem and four residual blocks with dilations $1, 2, 4, 8$ and width 64. The frequency branch removes the DC component of the real-FFT magnitude, peak-normalizes it, applies $\log(1 + 10\,\cdot)$, and uses two kernel-5 convolutions, the second grouped four ways. Both branches pool by mean, standard deviation, maximum, and the $0.25/0.75$ quantiles before projecting to 128 coordinates. Together they contain $108{,}248$ parameters; the complete frozen encoder has $158{,}812$ trainable parameters (0.16M rounded).

Synthetic data and supervision. HAR-CauKer retains CauKer's kernel-composition and causal-program machinery but adds short-scale and rough motion components and replaces an unbounded

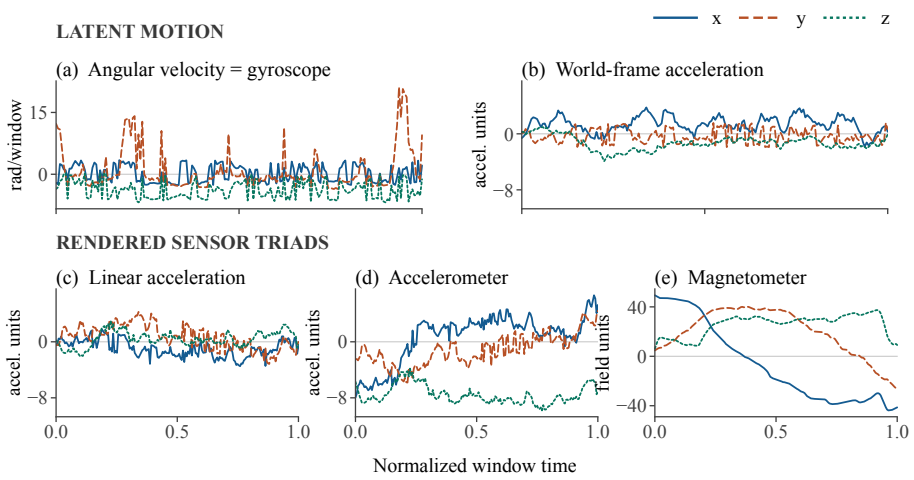

Figure 3: **One HAR-CauKer motion instance rendered as coupled sensors.** The gyroscope equals latent angular velocity (a); a shared orientation maps world-frame acceleration (b) to linear acceleration (c) and acceleration with gravity (d), and rotates a fixed magnetic vector into (e). All panels show the same 192 clean samples.

mean function with bounded ramps and impacts. One frozen graph defines a generating class; each instance resamples its processes and initial orientation. The classes identify generating dynamics, not human activities. The corpus contains 3,000 generating classes, 48 instances per class, and four inertial triads per instance, stored at length $L_{\text{sim}} = 192$. Triads are adapted to 128 samples for the encoder. For each program, a uniform permutation of the fourteen DAG nodes assigns the first three to angular velocity and the next three to world-frame acceleration; this assignment is fixed across instances of that program. Simulation uses normalized window time with $\Delta t = 1/L_{\text{sim}} = 1/192$, so angular velocity is expressed per unit window time. This integration step is not a physical 192 Hz acquisition rate. For body-frame angular velocity ω_t, world-frame linear acceleration a_t^w, gravity g, and a fixed world-frame magnetic vector m, the renderer integrates orientation and emits

$$\begin{aligned} R_{t+1} &= R_t \operatorname{Exp}([\bar{\omega}_t]_\times \Delta t), \quad & x_t^{\text{gyro}} &= \omega_t, \\ x_t^{\text{lin}} &= R_t^\top a_t^w, & x_t^{\text{acc}} &= R_t^\top (a_t^w + g), \quad x_t^{\text{mag}} = R_t^\top m. \end{aligned} \tag{5}$$

The implementation uses $\bar{\omega}_t = (\omega_t + \omega_{t+1})/2$ for the orientation update, with $\bar{\omega}_t = \omega_t$ at the last sample. Sensor observations use the current R_t. Each stored triad is independently cropped and resampled twice. The nuisance recipe includes uniform three-dimensional mounting rotations; axis reflection with probability 0.15; crop fractions in $[0.6, 1.0]$; per-axis gain noise with standard deviation 0.06; bias scaled by 0.05 of the window standard deviation; 3° non-orthogonality noise; global unit scale in $[0.4, 2.5]$ on a log scale; white or pink noise at 12–45 dB SNR, with pink noise chosen with probability 0.5; baseline drift with probability 0.4; and clipping or sample-and-hold dropout, each with probability 0.1. NT-Xent acts on normalized projection-head outputs from the two views. The factor head regresses standardized simulator targets: log dominant frequency of angular velocity and acceleration, periodicity, angular-velocity spectral entropy, log total rotation, tilt range, log dynamic-to-static acceleration ratio, rotation–translation correlation, log jerk, and graph depth. These targets describe the synthetic signals or generator, rather than human activity labels.

All heads take $z_1 \in \mathbb{R}^{129}$ as input. The program classifier is a linear $129 \to 3000$ layer; the factor regressor is a linear $129 \to 10$ layer; and the contrastive head is Linear$(129, 128)$–GELU–Linear$(128, 128)$, followed by L2 normalization. Factor regression applies all ten targets to every modality without a modality-specific mask. Some factors, such as world-relative tilt from gyroscope measurements, are not uniquely observable from a single triad. The auxiliary task therefore encourages prediction under the synthetic joint distribution, not guaranteed recovery of physical factors from every sensor type.

Pretraining objectives. All three heads use the same z_1 from each of two augmented views. We train with

$$\mathcal{L} = \mathcal{L}_{\text{class}} + \mathcal{L}_{\text{contrast}} + \mathcal{L}_{\text{factor}}, \tag{6}$$

where $\mathcal{L}_{\text{class}}$ averages generating-program cross-entropy over the views, $\mathcal{L}_{\text{contrast}}$ is NT-Xent at temperature 0.1 on normalized projection-head outputs with the two views as a positive pair, and $\mathcal{L}_{\text{factor}}$ averages mean-squared error to the ten standardized simulator targets. Program identity distinguishes generating mechanisms, view agreement encourages consistency under acquisition changes, and factor regression encourages prediction of motion-related factors under the synthetic joint distribution. The supervision, rendering, coupling, and corruption controls are described in Section B.1.

Optimization. Training uses $40{,}000$ steps, batch size 512, AdamW with learning rate 3×10^{-3}, a one-cycle schedule, weight decay 10^{-4}, mixed precision, and gradient clipping at 5. The training radius is $\rho = 1$ and the three loss weights are equal. The descriptor uses 128 fixed sampled coordinates plus log-scale, with affine-free BatchNorm. BatchNorm running statistics and target normalization values are learned from the synthetic training process; encoder evaluation freezes the former and discards all pretraining heads.

Frozen triad-wise readouts. At evaluation, real windows are sanitized, divided into named three-axis sensor groups (or consecutive triples when names are unavailable), and linearly resampled to 128 samples. The same frozen encoder processes each triad independently; its outputs are concatenated for multisensor windows. A change in triad count changes feature width, so a source-only classifier is fitted for each dataset configuration. Retrieval uses the 129-coordinate z_1 with source-gallery median/IQR standardization, L2 normalization, and cosine similarity. The component decomposition applies this processing separately to a, the unit proposal direction, and z_1. Recognition concatenates $[z_{0.2} \mid z_1 \mid z_{10}] \in \mathbb{R}^{387}$ per triad from one shared anchor and proposal; it fits a class-balanced, 300-tree random forest on source windows. Neither encoder, normalization, nor radius choice uses target data. Multiple radii increase readout width and cost even though they require only one encoder forward pass.

B.1 Causal ablations of the synthetic training world

Table 13: **Complete HAR-CauKer ablations.** Cross-dataset macro-F1 mean $\pm$ standard deviation over three pretraining seeds in separate ablation runs. Deltas use the matched full objective or full generator within each ablation group.

Pretraining configuration	Macro-F1 ↑	Δ (points)
Privileged supervision		
Full objective	$\mathbf{62.36 \pm 0.64}$	–
w/o simulator-factor targets	61.50 ± 0.52	-0.86
w/o synthetic-class target	61.01 ± 0.52	-1.35
contrastive objective only	59.86 ± 0.25	-2.50
Physical generation		
Full physical generator	$\mathbf{61.99 \pm 0.43}$	–
w/o rigid-body rendering	61.16 ± 0.76	-0.83
w/o cross-sensor coupling	60.69 ± 0.25	-1.30
Acquisition variation		
w/o acquisition corruptions	57.54 ± 0.34	$\mathbf{-4.82}$

The supervision controls in Table 5 separate the benefit of synthetic samples from the information available inside the simulator. Removing the continuous factor targets lowers cross-dataset macro-F1 by 0.86 points; removing the generating-class target lowers it by 1.35 points; and retaining only the contrastive objective lowers it by 2.50 points. The matched generator controls independently show losses of 0.83 points without rigid-body rendering and 1.30 points without cross-sensor coupling. The largest change is the 4.82-point loss without acquisition corruptions. The 0.83- and 0.86-point changes are comparable in scale to the 0.64-point pretraining-seed standard deviation of

Table 14: **Stability across anchor dimensions.** Mean ± standard deviation macro-F1 over coordinate draws 0–2 at pretraining seed 1. Here k counts sampled statistical coordinates before the appended log-scale coordinate, and z denotes HALO at $\rho = 1$.

View	Protocol	Metric	$k = 64$	$k = 128$	$k = 256$
a	Cross-subject	mF1	0.8261 ± 0.0067	0.8383 ± 0.0010	0.8417 ± 0.0016
a	Cross-position	mF1	0.4839 ± 0.0263	0.4829 ± 0.0137	0.4828 ± 0.0002
a	Cross-dataset	mF1	0.5518 ± 0.0172	0.5689 ± 0.0085	0.5818 ± 0.0135
z	Cross-subject	mF1	0.8272 ± 0.0056	0.8436 ± 0.0027	0.8538 ± 0.0024
z	Cross-position	mF1	0.5331 ± 0.0093	0.5228 ± 0.0020	0.5079 ± 0.0072
z	Cross-dataset	mF1	0.6033 ± 0.0089	0.6199 ± 0.0045	0.6027 ± 0.0153

Table 15: **Stability across anchor-coordinate draws at $k = 128$.** Macro-F1 at pretraining seed 1 in the separate anchor-construction study; draw 0 uses the shipped coordinate selection. These scores are not the fixed-readout sweep. "Within 2 SD" compares draw 0 with the other four coordinate draws. The final column gives the reference standard deviation over pretraining seeds 1–3 for z; it is comparable to the coordinate-draw variation.

View	Protocol	Draw 0	Draw 1	Draw 2	Draw 3	Draw 4	Mean ± SD	Draw-0 rank	Within 2 SD	Pretrain-seed SD
a	Cross-subject	0.8384	0.8393	0.8373	0.8392	0.8391	0.8387 ± 0.0009	4/5	Yes	–
a	Cross-position	0.4681	0.4952	0.4855	0.4863	0.4784	0.4827 ± 0.0101	5/5	No	–
a	Cross-dataset	0.5729	0.5746	0.5592	0.5552	0.5660	0.5656 ± 0.0084	2/5	Yes	–
z	Cross-subject	0.8443	0.8405	0.8459	0.8410	0.8481	0.8440 ± 0.0032	3/5	Yes	0.0017
z	Cross-position	0.5251	0.5219	0.5215	0.5316	0.5337	0.5268 ± 0.0056	3/5	Yes	0.0046
z	Cross-dataset	0.6234	0.6214	0.6149	0.6022	0.6178	0.6159 ± 0.0084	1/5	Yes	0.0064

the full objective. These differences suggest contributions from physical rendering and factor targets but do not establish their statistical significance or effects on the other two protocols.

C Stability Across Anchor Dimension and Coordinate Draws

We isolate anchor construction from pretraining randomness by fixing pretraining seed 1 and varying either the number k of sampled statistical coordinates or their random draw. These controls test whether HALO's anchor-relative gains can be explained by representation width or by a favorable sampled subset of the statistical grid.

The correction remains effective across anchor dimensions. Increasing k does not produce a monotonic transfer improvement: cross-position HALO decreases from 0.5331 at $k = 64$ to 0.5079 at $k = 256$, while cross-dataset peaks at $k = 128$. More importantly, z_1 exceeds its dimension-matched anchor at every k, by $4.92/3.99/2.51$ cross-position points and $5.15/5.10/2.09$ cross-dataset points for $k = 64/128/256$, respectively (Table 14). The transfer improvement therefore persists when anchor capacity changes.

The correction is consistent across the sampled coordinate draws. At $k = 128$, z_1 outperforms the matched anchor in all 15 protocol–draw comparisons. Its standard deviation across the five coordinate draws is only $0.32/0.56/0.84$ macro-F1 points across subject/position/dataset protocols, comparable to the $0.17/0.46/0.64$-point reference variation over pretraining seeds. Draw 0 is within two standard deviations of the other draws for all three HALO protocols and ranks third, third, and first. For anchor-only cross-position it is the weakest draw, while its corrected representation ranks first on cross-dataset transfer (Table 15). These controls support a correction benefit across the sampled anchor choices; they do not prove insensitivity to every descriptor or pretraining seed.

Table 16: **Complete cross-condition retrieval metrics.** Dimensions are per six-channel window. Scores are activity-macro-averaged within each unit and then unit-averaged. AP@100 divides by the number of relevant gallery items, capped at 100.

		Cross-position			Cross-dataset		
Representation	Dim.	R@1	R@5	AP@100	R@1	R@5	AP@100
CATCH22	132	30.7	58.3	14.0	36.0	68.8	14.7
Anchor a	258	41.2	64.2	24.0	42.4	69.1	20.7
Mantis-8M	1536	45.8	69.7	27.2	49.4	74.1	26.4
MantisV2	1536	46.4	69.6	27.5	52.1	77.4	28.7
Direction $\widehat{u}$	258	52.7	74.6	35.0	55.1	78.6	33.5
HALO z_1	258	**56.2**	**76.6**	**40.1**	**60.7**	**82.3**	**40.2**

D Complete Representation Results

For a query of activity c, AP@K is

$$\mathrm{AP}\,@K = \frac{\sum_{r=1}^{K} P(r)\,\mathbf{1}[y_r = c]}{\min\{N_c, K\}}, \tag{7}$$

where N_c is the number of activity-c examples in the gallery. A class absent from the gallery receives zero. Recall@1 and Recall@5 indicate whether a same-activity example occurs within the first one or five ranks. Metrics are macro-averaged over query activities and then averaged over evaluation units. All source preprocessing is fitted separately for each representation; comparing their performance does not presume identical raw coordinate metrics.

Descriptive paired intervals. The unit-bootstrap 95% intervals for HALO-minus-MantisV2 Recall@1 are $[7.0, 12.3]$ points across positions and $[5.6, 11.8]$ across datasets. They summarize variation over the selected units. They do not account for all dependence introduced by overlapping source sets and repeated target subjects across position directions. We therefore report the complete unit distributions and win counts in the main text without treating these intervals as tests based on independent experimental replicates.

Linear probing. The source-trained multinomial probe chooses regularization on a subject-grouped inner source split. Feature scaling and model selection use source windows only. HALO's mean improvement over the better of the anchor and direction within each unit is 1.0 macro-F1 across positions ($18/28$ units) and 4.2 across datasets ($11/13$). In retrieval, the corresponding improvements are 3.4 Recall@1 ($24/28$) and 5.6 ($11/13$). This per-unit comparator is diagnostic and uses evaluation scores to identify the better component; it is not an operational selector.

D.1 Direction intervention

Direction and raw-norm interventions share deterministic donor permutations and selection masks. Donors are chosen separately within gallery and query; no vector is transplanted across that boundary. Every window retains its own anchor and projection budget. The within-class arm uses ground-truth activity to choose donors and is label-assisted analysis only.

At $\rho = 1$, projection is active on approximately 98.7% of triads. In the active regime, $\delta_\rho = \rho\|a\|_2\widehat{u}$, so the proposal norm cancels algebraically. Norm shuffling preserves the output whenever the donor norm also leaves the constraint active; the original active fraction alone does not give the exact fraction unchanged after intervention. Direction shuffling establishes that the preserved direction carries useful input-dependent structure; it does not establish that distance traveled is irrelevant or identify a particular physical feature encoded by the residual.

Holding the anchor fixed, the active projection has Jacobian $\partial\delta_\rho/\partial u = (\rho\|a\|_2/\|u\|_2)(I - \widehat{u}\widehat{u}^\top)$, which removes the radial component of the task gradient. Proposal norms can still change through shared parameter updates, inactive examples, and weight decay. Training at $\rho = 1$ therefore does not calibrate the raw norm for loose-radius inference.

Table 17: **Direction intervention at $\rho = 1$.** Recall@1 without fitting a classifier. Direction and norm arms share donor permutations and preserve each window's anchor and budget. The within-class arm uses ground-truth activity to choose donors.

	Cross-position		Cross-dataset	
Representation	R@1	Drop	R@1	Drop
Original z_1	56.2	–	60.7	–
Direction shuffled	40.3	**15.9**	40.7	**20.0**
Norm shuffled	56.2	0.0	60.4	0.3
Direction shuffled within class	52.4	3.7	56.8	3.9

Table 18: **Placement displacement relative to activity separation.** One checkpoint, 16 simultaneous dataset/site-pair cells from DSADS, PAMAP2, and MHEALTH. NSR is median paired-site distance divided by median between-activity distance within each cell; mean and median summarize cells. "Lower" counts cells with NSR below the anchors. The last two columns report projection activity and displacement relative to HALO's own anchor in this diagnostic.

	NSR		Lower	Projection	Displacement
Representation	Mean	Median	/16	active	$\|\Delta z\|/\|\Delta a\|$
Mantis-8M	0.942	0.987	0	–	–
MantisV2	0.942	0.975	0	–	–
CATCH22	0.711	0.729	12	–	–
HALO anchors	0.796	0.820	–	–	1.00
HALO, $\rho = 0.2$	0.786	0.820	14	100%	0.88
HALO, $\rho = 1$	0.655	0.678	16	100%	0.70
HALO, $\rho = 10$	0.639	0.674	16	0%	1.24

D.2 Placement displacement

For simultaneous recordings at two body sites, we measure nuisance-to-signal ratio (NSR) as the median paired-site distance divided by the median between-activity distance. This diagnostic uses 16 dataset/site-pair cells from DSADS, PAMAP2, and MHEALTH. It complements recognition by asking whether a representation moves the same activity less across positions, relative to its separation between activities.

D.3 Structural recognition controls

Table 3 reports recognition scores for the direct-concatenation and unbounded-correction controls. The variants are separately trained runs, so the cross-row gaps summarize the observed design pattern rather than paired effects.

D.4 Complete radius sweep

Table 19: **Complete fixed-readout sweep for one checkpoint.** Macro-F1 without encoder retraining. Dimensions are per triad. Transfer columns average 28 position units and 13 held-out datasets; the main benchmark instead averages six position-direction means. The default stack uses the headline cross-subject benchmark reference.

Readout	Dimension	Subject	Position	Dataset
$\rho = 0.2$	129	84.4	50.9	61.0
$\rho = 1$	129	83.4	58.8	65.7
$\rho = 3$	129	82.3	56.31	63.43
$\rho = 10$	129	82.2	**59.1**	**66.5**
$\rho = 1$, tiled three times	387	83.3	58.3	65.4
Default: $\{0.2, 1, 10\}$	387	**84.9**	56.5	63.4

The fixed-radius rows reuse one frozen anchor and proposal. The tiled control repeats z_1 without adding information, while the default readout concatenates three distinct radii. These results compare readouts, not separately pretrained encoders, and do not provide a label-free rule for choosing a radius on a new target. The default stack reuses the headline cross-subject benchmark score of 84.9 throughout. The anchor-construction study likewise reports its own run-specific results; its draw-0 row is not the single-radius reference used here.

E Full Activity-Recognition Results

The following tables provide all evaluated methods for each protocol. Cross-subject standard deviations describe five folds. Cross-position standard deviations describe available subject folds within each of the six ordered position directions. The macro-F1 values in the detailed cross-subject table use a 0–1 scale; the cross-position and cross-dataset tables and main summary use 0–100. Ranks are recomputed across all listed configurations from the displayed dataset or direction means, assigning average ranks to ties. MiniROCKET's own average rank excludes its incomplete RealWorld result. New baseline rows report sample standard deviations; their aggregate deviations describe all folds or held-out datasets, not repeated pretraining runs.

Table 20: **Cross-subject evaluation** (macro-F1, 0–1; mean ± SD over five subject folds). Overall means average thirteen dataset means. ‡ MiniROCKET completed only two RealWorld folds due to memory limits; its 0.843 mean includes this partial result, while its rank uses twelve complete datasets. HARS: HARSense; MHLTH: MHEALTH; Motion: MotionSense; RealW: RealWorld; TNDA: TNDA-HAR; UCI: UCI HAR/HAPT; USC: USC-HAD. *Source-pretrained reproduction with random-forest readout (Section A.6); ranks use displayed dataset means.

model	DSADS	HARS	HHAR	KU-HAR	MHLTH	Motion	PAMAP2	RealW	Shoaib	TNDA	UCI	USC	WISDM	mean	avg. rank
HALO (ours)	.931 ±.051	.824 ±.109	.839 ±.059	.884 ±.105	.958 ±.036	**.960** ±.020	**.773** ±.065	**.816** ±.020	.960 ±.017	**.917** ±.026	.751 ±.027	.799 ±.039	**.628** ±.025	**.849** ±.108	**2.5**
Fitted on the source-subject training split															
MiniROCKET (Dempster et al., 2021)	.927 ±.040	.799 ±.073	**.850** ±.034	**.897** ±.110	.959 ±.045	.918 ±.067	.757 ±.113	.767‡ ±.034‡	**.974** ±.016	.914 ±.028	**.807** ±.008	.797 ±.079	.586 ±.023	.843 ±.118	3.0
TinyHAR (Zhou et al., 2022)	.879 ±.044	.666 ±.120	.769 ±.045	.876 ±.141	.910 ±.050	.903 ±.064	.663 ±.068	.746 ±.035	.953 ±.018	.858 ±.030	.757 ±.040	.735 ±.063	.541 ±.026	.789 ±.131	7.9
MCNN (Yang et al., 2015)	.859 ±.083	.766 ±.076	.780 ±.053	.860 ±.135	.854 ±.079	.892 ±.062	.654 ±.083	.731 ±.012	.944 ±.020	.851 ±.027	.769 ±.036	.725 ±.059	.552 ±.023	.788 ±.119	8.5
TinierHAR (Bian et al., 2025)	.854 ±.078	.515 ±.079	.772 ±.048	.788 ±.136	.877 ±.098	.895 ±.060	.633 ±.113	.750 ±.019	.959 ±.014	.855 ±.037	.714 ±.026	.746 ±.059	.516 ±.038	.759 ±.148	9.5
1D CNN (Hammerla et al., 2016)	.774 ±.054	.671 ±.148	.752 ±.070	.789 ±.133	.844 ±.092	.871 ±.050	.540 ±.150	.691 ±.037	.943 ±.028	.850 ±.028	.755 ±.055	.694 ±.092	.518 ±.045	.746 ±.143	11.3
DeepConvLSTM (Ordóñez & Roggen, 2016)	.700 ±.062	.455 ±.120	.728 ±.103	.779 ±.134	.813 ±.066	.889 ±.074	.497 ±.130	.708 ±.023	.917 ±.034	.837 ±.043	.698 ±.038	.674 ±.084	.512 ±.034	.708 ±.160	13.6
SA-HAR (Mahmud et al., 2020)	.745 ±.095	.352 ±.262	.723 ±.075	.665 ±.142	.870 ±.044	.805 ±.046	.524 ±.098	.622 ±.016	.894 ±.038	.784 ±.032	.635 ±.023	.551 ±.095	.481 ±.031	.666 ±.179	15.1
Handcrafted features, no network															
RF (Breiman, 2001) + catch22 (Lubba et al., 2019)	.808 ±.080	.656 ±.112	.779 ±.064	.820 ±.117	.944 ±.026	.859 ±.095	.737 ±.071	.748 ±.031	.907 ±.030	.862 ±.022	.628 ±.029	.644 ±.052	.498 ±.024	.761 ±.137	10.8
RF + HALO anchors $A(x)$	.928 ±.020	**.833** ±.090	.783 ±.057	.869 ±.118	.951 ±.022	.938 ±.027	.771 ±.082	.797 ±.014	.957 ±.018	**.917** ±.025	.751 ±.030	.805 ±.047	.604 ±.026	.839 ±.110	3.7
Pretrained on a general time-series or motion corpus, frozen															
MOMENT-1-base (Goswami et al., 2024)	.821 ±.064	.673 ±.058	.803 ±.065	.831 ±.105	.931 ±.041	.871 ±.100	.729 ±.058	.764 ±.031	.915 ±.023	.869 ±.025	.631 ±.022	.684 ±.042	.497 ±.017	.771 ±.131	9.6
MantisV2 (4.2 M)	.926 ±.045	.797 ±.125	.827 ±.061	.886 ±.111	.945 ±.036	.925 ±.057	.760 ±.057	.791 ±.033	.952 ±.012	.911 ±.027	.770 ±.022	.803 ±.043	.598 ±.023	.838 ±.111	4.3
Mantis-8M (8 M)	**.938** ±.045	.832 ±.132	**.850** ±.055	.882 ±.106	.955 ±.035	.908 ±.076	.755 ±.074	.801 ±.021	.953 ±.004	.912 ±.029	.746 ±.008	**.839** ±.054	.585 ±.027	.843 ±.115	4.0
UniMTS (linear probe) (Zhang et al., 2024b)	.795 ±.044	.464 ±.023	.726 ±.025	.521 ±.111	.772 ±.062	.873 ±.017	.482 ±.142	.745 ±.035	.845 ±.054	.800 ±.048	.571 ±.037	.482 ±.056	.480 ±.013	.658 ±.164	15.3
Pretrained on a motion corpus, then fine-tuned end-to-end															
UniMTS (full fine-tune) (Zhang et al., 2024b)	.882 ±.042	.739 ±.115	.784 ±.075	.786 ±.195	**.975** ±.022	.957 ±.036	.723 ±.123	.804 ±.043	.973 ±.005	.892 ±.039	.737 ±.037	.624 ±.081	.598 ±.032	.806 ±.142	6.1
Source-pretrained IMU methods; frozen encoder and random-forest readout															
LIMU-BERT* (Xu et al., 2021)	.806 ±.099	.659 ±.165	.679 ±.070	.492 ±.191	.833 ±.071	.832 ±.085	.643 ±.143	.590 ±.038	.859 ±.044	.744 ±.032	.681 ±.035	.690 ±.086	.432 ±.031	.688 ±.154	14.6
LIMU-BERT (seg4)* (Xu et al., 2021)	.804 ±.096	.647 ±.166	.687 ±.073	.532 ±.175	.838 ±.070	.825 ±.082	.632 ±.137	.592 ±.043	.863 ±.046	.746 ±.031	.681 ±.040	.689 ±.072	.439 ±.031	.690 ±.148	14.5
RelCon* (Xu et al., 2025)	.702 ±.097	.534 ±.126	.485 ±.078	.571 ±.096	.909 ±.047	.704 ±.055	.572 ±.082	.556 ±.015	.745 ±.026	.632 ±.021	.455 ±.023	.482 ±.048	.344 ±.024	.592 ±.155	16.4

Table 21: **Cross-position evaluation** (macro-F1, 0–100; mean ± standard deviation across subject folds for each ordered source-to-target position pair). All methods use identical folds. For UniMTS, sensors are assigned to its 22-joint input according to each dataset's published placement. Means average six directions; ranks use displayed direction means. *Source-pretrained reproduction with random-forest readout (Section A.6).

model	lower→trunk	lower→upper	trunk→lower	trunk→upper	upper→lower	upper→trunk	mean	avg. rank
HALO (ours)	**54.584** ±4.424	**57.579** ±9.486	**52.221** ±6.657	**64.180** ±9.151	**55.273** ±12.501	**54.082** ±15.385	**56.320** ±10.007	**1.0**
Handcrafted, no network								
RF + catch22 (Lubba et al., 2019)	45.235 ±4.493	39.041 ±11.299	38.542 ±8.173	44.163 ±9.136	34.420 ±12.807	38.105 ±17.531	39.918 ±10.768	8.5
RF + HALO anchors $A(x)$	44.332 ±11.492	52.033 ±7.617	47.421 ±9.342	56.820 ±12.626	47.194 ±12.624	46.921 ±19.360	49.120 ±11.954	3.0
Pretrained once on a general time-series corpus, frozen								
MOMENT-1-base (Goswami et al., 2024)	49.503 ±7.622	46.070 ±11.796	42.647 ±5.358	50.710 ±8.172	42.800 ±12.625	45.626 ±15.809	46.226 ±10.125	4.5
MantisV2 (4.2 M)	37.590 ±8.310	48.072 ±10.560	37.724 ±5.958	50.259 ±11.622	45.368 ±9.527	42.145 ±14.583	43.526 ±10.538	6.3
Mantis-8M (8 M)	33.379 ±9.434	44.185 ±8.848	43.210 ±8.306	48.047 ±10.626	43.234 ±11.777	37.946 ±20.463	41.667 ±11.711	8.0
UniMTS (linear probe) (Zhang et al., 2024b)	2.573 ±1.640	7.863 ±2.880	4.511 ±1.763	5.125 ±3.006	16.466 ±5.375	12.739 ±8.804	8.213 ±6.396	19.8
Self-supervised on this benchmark's own unlabelled source split, then a supervised classifier on the frozen encoder								
LIMU-BERT* (Xu et al., 2021)	23.281 ±6.765	34.125 ±16.895	24.413 ±6.932	27.790 ±8.829	30.103 ±6.382	29.193 ±4.453	28.151 ±9.373	16.7
LIMU-BERT (seg4)* (Xu et al., 2021)	23.279 ±6.661	34.631 ±16.956	24.683 ±6.665	27.807 ±9.580	29.899 ±5.902	29.647 ±4.532	28.324 ±9.460	16.3
RelCon* (Xu et al., 2025)	30.737 ±2.112	40.100 ±4.166	32.079 ±6.405	31.674 ±8.129	28.496 ±4.794	31.654 ±6.373	32.457 ±6.422	14.5
BioBankSSL (Yuan et al., 2024)	37.927 ±3.444	28.130 ±7.349	32.758 ±8.390	36.263 ±10.280	33.340 ±8.993	39.839 ±7.400	34.709 ±8.266	12.7
CRT (Zhang et al., 2024a)	45.999 ±3.682	39.226 ±7.138	45.993 ±7.228	43.508 ±9.529	46.883 ±9.513	41.221 ±10.140	43.805 ±7.958	6.0
FOCAL (Liu et al., 2023)	44.326 ±5.083	39.020 ±6.180	46.032 ±6.879	42.836 ±9.233	50.650 ±7.613	48.988 ±8.449	45.309 ±7.789	5.3
SimMTM (Dong et al., 2023)	38.795 ±4.249	31.917 ±6.744	31.960 ±4.614	39.325 ±10.981	36.293 ±10.124	41.070 ±10.589	36.560 ±8.393	11.2
Trained end-to-end on this unit's own source split, no pretraining								
TinyHAR (Zhou et al., 2022)	38.979 ±7.291	43.281 ±8.465	33.510 ±12.687	39.298 ±12.357	39.174 ±12.536	37.981 ±19.919	38.704 ±11.808	9.3
MCNN (Yang et al., 2015)	39.377 ±3.388	43.139 ±5.910	32.206 ±7.731	39.531 ±7.847	36.108 ±11.217	44.959 ±19.083	39.220 ±10.106	8.5
1D CNN (Hammerla et al., 2016)	33.240 ±5.309	29.788 ±5.849	30.396 ±8.403	33.521 ±15.746	31.484 ±7.615	41.441 ±10.798	33.312 ±9.547	14.0
DeepConvLSTM (Ordóñez & Roggen, 2016)	31.791 ±3.143	32.149 ±6.100	34.299 ±5.237	34.670 ±11.993	35.022 ±7.538	47.404 ±13.725	35.889 ±9.320	10.8
SA-HAR (Mahmud et al., 2020)	34.368 ±5.956	30.722 ±4.117	28.802 ±5.903	33.549 ±11.590	33.380 ±13.121	36.179 ±5.155	32.833 ±8.120	14.3
Pretrained on a motion corpus, then fine-tuned end-to-end on this unit								
UniMTS (full fine-tune) (Zhang et al., 2024b)	16.222 ±6.408	17.294 ±6.629	8.947 ±3.147	15.326 ±7.004	6.656 ±4.085	16.471 ±10.554	13.486 ±7.267	19.2

Table 22: **Leave-one-dataset-out evaluation** (macro-F1, 0–100; shared eight-class vocabulary). Source pools match across methods. Abbreviations: HARSe: HARSense; KU-HA: KU-HAR; MHEAL: MHEALTH; Motio: MotionSense; PAMAP: PAMAP2; RealW: RealWorld; Shoai: Shoaib; TNDA-: TNDA-HAR; UCI: UCI HAR/HAPT; USC-H: USC-HAD. *Source-pretrained reproduction with random-forest readout (Section A.6); ranks use displayed dataset means.

model	DSADS	HARSe	HHAR	KU-HA	MHEAL	Motio	PAMAP	RealW	Shoai	TNDA-	UCI	USC-H	WISDM	mean	avg. rank
HALO (ours)	**70.524**	14.777	**72.893**	17.475	61.359	**90.149**	71.766	**63.330**	**74.044**	**75.347**	**64.430**	**74.852**	**73.562**	**63.424** ±22.165	**3.1**
Handcrafted, no network															
RF + catch22 (Lubba et al., 2019)	59.049	5.733	55.527	53.249	51.094	73.270	63.223	51.871	43.451	60.400	62.073	54.087	58.760	53.214 ±15.968	8.5
RF + HALO anchors $A(x)$	68.318	12.746	71.462	12.378	61.554	88.195	65.540	59.718	58.183	72.020	62.900	67.182	73.008	59.477 ±22.179	4.9
Pretrained once on a general time-series corpus, frozen															
MOMENT-1-base (Goswami et al., 2024)	59.516	8.452	60.771	55.861	50.384	75.455	61.686	55.297	40.499	61.777	62.916	61.344	60.355	54.947 ±16.056	7.2
MantisV2 (4.2 M)	58.542	28.314	58.833	32.547	60.176	86.763	36.970	46.767	50.868	66.379	64.000	40.254	70.013	53.879 ±16.649	7.8
Mantis-8M (8 M)	54.143	22.110	49.478	16.590	52.444	89.059	46.110	51.905	64.271	66.702	60.557	63.536	70.328	54.402 ±19.149	7.8
UniMTS (linear probe) (Zhang et al., 2024b)	56.468	6.794	43.840	21.818	57.054	68.268	61.040	48.900	58.492	48.794	49.064	64.559	38.284	47.952 ±17.360	10.5
Self-supervised on this benchmark's own unlabelled source split, then a supervised classifier on the frozen encoder															
LIMU-BERT* (Xu et al., 2021)	37.609	26.861	44.218	29.027	10.961	62.248	41.579	33.592	13.911	45.895	61.120	33.761	39.860	36.973 ±15.219	13.3
LIMU-BERT (seg4)* (Xu et al., 2021)	37.606	25.258	45.713	29.488	10.077	62.285	40.785	34.003	13.297	46.272	60.634	36.570	39.442	37.033 ±15.467	13.4
RelCon* (Xu et al., 2025)	35.057	**32.959**	40.627	34.961	41.710	59.748	36.743	37.249	49.109	41.293	45.624	46.881	57.506	43.036 ±8.471	12.8
BioBankSSL (Yuan et al., 2024)	66.170	9.047	41.337	62.164	43.110	58.489	58.674	56.920	35.402	59.495	47.684	70.149	38.477	49.778 ±16.460	9.4
CRT (Zhang et al., 2024a)	59.928	7.908	34.067	52.070	43.929	56.127	58.973	53.537	42.519	56.842	43.396	57.870	36.543	46.439 ±14.493	11.9
FOCAL (Liu et al., 2023)	60.894	10.568	38.320	61.630	43.722	58.208	57.793	58.117	48.547	58.624	45.678	67.284	38.482	49.836 ±15.013	9.5
SimMTM (Dong et al., 2023)	59.677	10.543	35.178	**66.886**	43.666	46.793	54.980	53.639	29.763	54.553	47.943	73.532	36.213	47.182 ±16.630	11.4
Trained end-to-end on this unit's own source split, no pretraining															
TinyHAR (Zhou et al., 2022)	63.178	13.237	35.223	43.095	48.287	63.510	26.720	60.318	34.353	37.829	39.782	10.707	36.107	39.411 ±16.837	12.8
MCNN (Yang et al., 2015)	61.515	16.290	35.269	46.999	**62.496**	68.240	30.789	61.020	30.014	45.709	60.111	60.991	36.152	47.353 ±16.370	9.7
1D CNN (Hammerla et al., 2016)	51.635	16.356	29.815	32.925	39.150	53.080	26.395	55.232	29.064	44.487	61.495	44.945	34.223	39.908 ±13.218	14.4
DeepConvLSTM (Ordóñez & Roggen, 2016)	59.303	14.820	31.964	38.286	55.065	46.443	28.687	55.872	22.203	42.540	59.071	43.851	35.037	41.011 ±14.247	13.4
SA-HAR (Mahmud et al., 2020)	51.353	9.836	20.213	35.377	24.945	37.544	14.931	45.291	11.587	40.608	42.430	19.982	33.424	29.809 ±13.688	17.9
Pretrained on a motion corpus, then fine-tuned end-to-end on this unit															
UniMTS (full fine-tune) (Zhang et al., 2024b)	62.736	4.460	35.359	9.959	47.827	60.611	**73.063**	51.458	50.106	54.612	63.122	54.353	43.151	46.986 ±20.096	10.3

F DEPLOYMENT COST

Table 23 reports costs in the measurement environment. The six-channel three-radius representation contains 774 coordinates and takes approximately 4.65 ms per window for encoding, or 5.75 ms including the optimized forest. The measured single-radius encoder takes 4.75 ms; the small difference from the three-radius timing is within their measurement spread, and is not evidence that extra radii reduce computation. The forest must be stored and refitted for each source task (median fitting time 62 s). Timing uses workstation CPU and datacenter GPU hardware and does not measure wearable-device energy.

Standalone anchor extraction takes 2.96 ms, approximately 63.6% of the full encoder's 4.65 ms CPU latency. This ratio compares separate timings rather than an instrumented component breakdown. Statistical extraction, including spectra and order statistics, is therefore a substantial deployment cost despite having no trainable parameters. HALO is slower in this environment than small supervised networks such as 1D CNN (0.12 ms), MCNN (0.60 ms), and TinyHAR (1.76 ms). Shared computations, incremental window statistics, and optimized FFT and order-statistic routines are possible optimizations; their benefit and wearable energy costs remain unmeasured.

The Mantis rows profile frozen encoders only. Like HALO, both Mantis models require a source-fitted random forest for the recognition benchmark, but their forest latency and storage were not profiled. Thus encoder rows support encoder-only comparisons, and HALO's combined row does not establish an end-to-end cost ranking against the Mantis pipelines.

Table 23: **Deployment cost.** Latency is measured per window in one process: single-thread CPU at batch 1 and GPU at batch 256. Encoder CPU brackets are interquartile ranges over 30 calls; readout brackets show the range across datasets. RandomForest size and latency use a compact tree layout and direct traversal. Dimensions are reported per triad, channel, or window as marked. MMACs count convolutional, linear, and recurrent operations; the three HALO radii share one encoder pass. "Refit?" refers to the listed component: all frozen encoders still require a source-fitted classifier for recognition. Mantis readout costs are not included.

representation	params (M)	MB	dim / unit	MMACs	CPU μs	GPU μs	refit?
Parameter-free							
HALO anchors $A(x)$	0	0	129 / triad	–	2960 [2933,2987]	29.6	no
Supervised, trained on the source split							
TinyHAR	0.06	0.23	64 / window	5.59	1757 [1696,1824]	13.2	yes
MCNN	0.51	1.93	128 / window	18.02	603 [597,616]	3.3	yes
TinierHAR	0.01	0.03	32 / window	0.3	1268 [1258,1273]	2.4	yes
1D CNN	0.04	0.14	128 / window	1.47	116 [103,119]	0.7	yes
DeepConvLSTM	0.3	1.13	128 / window	35.13	1094 [1084,1118]	4.8	yes
SA-HAR	0.03	0.11	32 / window	3.56	835 [826,864]	7.6	yes
Pretrained once, then frozen							
MantisV2	4.19	17.98	256 / channel	871.37	19215 [19150,19397]	237.5	no
Mantis-8M	8.1	30.95	256 / channel	1612.19	27378 [25314,31196]	326.7	no
HALO, one radius	0.16	0.61	129 / triad	2.87	4754 [4632,5843]	33.4	no
HALO (ours), three radii	0.16	0.61	387 / triad	2.87	4653 [4623,7182]	31.1	no
Readout, refitted for every deployment (median [min, max] over the 13 corpora)							
RandomForest on $A(x)$	–	1.67 [0.12,40.73]	258 / window	–	1088 [566,1161]	–	yes
RandomForest on HALO, three radii	–	1.75 [0.13,46.02]	774 / window	–	1097 [566,1162]	–	yes
HALO as deployed (encoder + forest)	0.16	2.35	–	2.9	5750	–	yes