Physical Self-Supervised Learning: IMU Sensing without Manual Labels

Yuyang Leng
George Mason University
Fairfax, VA, USA
yleng2@gmu.edu

Renyuan Liu
George Mason University
Fairfax, VA, USA
rliu23@gmu.edu

Shaohan Hu
Global Technology Applied Research,
JPMorgan Chase
New York, NY, USA
shaohan.hu@jpmchase.com

Peijun Zhao
Global Technology Applied Research,
JPMorgan Chase
New York, NY, USA
peijun.zhao@jpmchase.com

Chun-Fu (Richard) Chen
Global Technology Applied Research,
JPMorgan Chase
New York, NY, USA
richard.cf.chen@jpmchase.com

Songqing Chen
George Mason University
Fairfax, VA, USA
sqchen@gmu.edu

Shuochao Yao
George Mason University
Fairfax, VA, USA
shuochao@gmu.edu

Abstract

Deep neural networks have become a promising approach for IMU-based sensing, but their scalability is fundamentally limited by costly labeled data and poor robustness to heterogeneous devices, placements, and users. Existing unsupervised and self-supervised methods reduce but do not remove this dependence, still requiring labeled data for domain adaptation and largely ignoring known physical structure. We propose physical self-supervised learning, an autoencoder-style paradigm for label-free IMU sensing. We replace the conventional neural decoder with an auto-adaptive physics decoder—a learnable family of kinematic equations that enforces explicit physical structure while adapting across environments—and adopt a hybrid two-stage IMU encoder with reconstruction in a structured latent space to mitigate sensor noise. Our framework further introduces probabilistic frequency–spatial constraints to disentangle sensor and object motion, a multi-view kinematic tree to exploit sparse physical self-supervised signals, and an uncertainty-aware formulation to handle the inherent ambiguity of IMU inference. Evaluated on inertial tracking and full-body motion capture over public datasets and realistic deployments, physical self-supervised learning reduces errors by up to 5× for tracking and 4× for motion capture in challenging generalization scenarios, consistently outperforming state-of-the-art supervised and self-supervised baselines without any labels. Code is available at GitHub

CCS Concepts

• **Computing methodologies** → **Machine learning**; • **Computer systems organization** → **Sensor networks**.

MobiSys '26, Cambridge, United Kingdom

ACM ISBN 979-8-4007-2027-7/26/06
https://doi.org/10.1145/3745756.3809252

Keywords

Mobile sensing, IMU sensing, Label-free learning, Motion capture

ACM Reference Format:
Yuyang Leng, Renyuan Liu, Shaohan Hu, Peijun Zhao, Chun-Fu (Richard) Chen, Songqing Chen, and Shuochao Yao. 2026. Physical Self-Supervised Learning: IMU Sensing without Manual Labels. In *The 24th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '26), June 21–25, 2026, Cambridge, United Kingdom.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3745756.3809252

1 Introduction

For over a decade, the mobile sensing community has leveraged deep neural networks (DNNs) to tackle a wide range of challenges [36, 37, 74]. Rapid advances in deep learning have led to powerful data-driven paradigms capable of addressing increasingly complex mobile sensing problems efficiently on mobile devices [27, 39, 42–44, 54, 75–77]. Among the many sensing modalities, the Inertial Measurement Unit (IMU) is among the most ubiquitous: IMUs are embedded in virtually all modern smartphones, wearables, and IoT devices, and they underpin large-scale applications such as smart health [15, 19], autonomous vehicles [23, 82], drones [18, 85], and augmented reality [55, 71].

Despite this progress, two major challenges continue to limit data-driven (DNN-based) mobile sensing: the scarcity of labeled data and the heterogeneity of sensing environments. Unlike well-established DNN application domains such as vision and language, where data collection and labeling can be decoupled thanks to the human-interpretable nature of the data, embedded and mobile sensing tasks often require labels to be obtained simultaneously with data collection. This typically depends on deploying additional sensor modalities or involving human annotators in controlled lab environments, making labeling extremely costly and fundamentally unscalable. At the same time, the performance of data-driven models is highly sensitive to the sensing environment. Factors such as

device manufacturing variations, user behavior, deployment locations, sensor orientations, and movement during the sensing period (e.g., a loosely worn smartwatch or a smartphone in a loose pocket or bag), along with other contextual variables, can all critically impact model performance [63]. These two issues reinforce one another: to combat environmental heterogeneity, we need more labeled data collected outside controlled lab settings, yet obtaining such labels requires heavy manual effort or extra instrumentation and is difficult to scale in practice.

This paper therefore asks the following question: *Can we develop a learning paradigm for IMU sensing that requires no manually labeled data, yet achieves performance comparable to or even surpassing that of supervised counterparts across diverse sensing environments?*

Unsupervised and self-supervised learning paradigms are natural candidates when seeking to reduce labeling effort. Many of the most successful approaches in this space are autoencoder-style methods: from classical autoencoders [17], to variational autoencoders (VAEs) [33], to the recent masked autoencoders (MAEs) [22]. Each of these has inspired a family of models designed to reduce labeling requirements while automatically learning useful representations from raw data. In the mobile sensing domain, there have also been significant efforts to adapt these techniques to sensing tasks [20, 29, 53, 57, 70]. Unfortunately, none of these efforts fully eliminates the need for labels: a non-trivial fraction of labeled data (typically on the order of 10% $\sim$ 20%) is still required for domain adaptation. As a result, the data-scaling problem in mobile sensing remains unresolved, since manual data collection and labeling are still needed whenever the sensing environment changes.

Our key observation is that existing approaches grant autoencoder-style models too much freedom. On the one hand, because both the encoder and decoder are parameterized by DNNs, these models excel at learning representations that preserve information and suppress noise, but they also tend to produce uninterpretable latent spaces. Consequently, additional labeled data are needed to map these latent representations to the physical quantities of interest and to align them across sensing environments. On the other hand, most IMU sensing tasks are governed by kinematic equations that are known a priori. In the absence of explicit guidance, current autoencoder-style methods expend much of their capacity fitting these known physical dynamics, rather than focusing on adapting to the underlying variability of real-world sensing environments. Our intuition is that physical laws already provide rich prior knowledge that can be embedded into a new learning paradigm—one that simultaneously guides the model to output physically meaningful IMU quantities and directs its capacity toward automatically adapting to complex, hard-to-model sensing environments.

To this end, we propose *physical self-supervised learning*, a novel autoencoder-style paradigm that enables label-free learning for IMU sensing tasks (e.g., inertial tracking and full-body motion capture). Our paradigm replaces the fully DNN-parameterized decoder with an auto-adaptive physics decoder: a parameterized set of kinematic equations that automatically adapts to data to model a family of kinematic systems. Unlike traditional physics-based modeling, which requires handcrafting precise physical models for each sensing setup—often laborious and brittle—and unlike purely data-driven decoders that yield opaque representations, our auto-adaptive physics decoder embeds explicit kinematic structure while learning to adapt to diverse sensing scenarios, including variations in sensor placement and movement. In addition, we replace a monolithic encoder with a hybrid, two-stage IMU encoder and perform reconstruction in a structured latent space, allowing the reconstruction loss to act in the latent domain and thereby mitigating the impact of sensor noise.

Beyond this core architectural design, our physical self-supervised framework introduces three additional contributions. First, we propose *probabilistic frequency–spatial constraints* that naturally disentangle sensor motion from object motion. Second, we extend the classical kinematic tree for articulated objects to a *multi-view kinematic tree*, enabling the use of sparse physical self-supervised signals from multiple sensor locations. Third, we develop an uncertainty-aware formulation that explicitly models ambiguity and helps resolve the inherently ill-posed nature of IMU sensing tasks.

We evaluate our physical self-supervised learning framework on two representative IMU sensing tasks: (1) Inertial tracking: estimating the global translation and orientation of an object using a single IMU, and (2) Motion capture: recovering full-body pose using multiple IMUs. Across multiple public datasets and realistic deployment scenarios, our framework consistently outperforms state-of-the-art supervised and self-supervised fine-tuning baselines. For inertial tracking, it reduces error by 1.5×–2× compared to prior deep learning approaches, and under challenging generalization settings the improvement reaches up to 5×. For motion capture, our method achieves 1.3×–1.8× lower error, with 3×–4× gains in generalization scenarios. Taken together, these results demonstrate that our label-free physical self-supervised learning framework not only outperforms all supervised and self-supervised baselines, but also generalizes robustly across sensor deployments, users, motion patterns, and datasets.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 presents the technical details of our physical self-supervised learning framework. The experimental setup and evaluation results are given in Sections 4 and 5. Finally, Section 6 concludes the paper.

2 Related Work

Self-Supervised Learning. Due to the scarcity of labeled data and the diverse nature of sensing data, self-supervised learning has become a key area of research in the mobile sensing community. Techniques such as contrastive learning [21, 53, 56] and masked reconstruction [20, 57, 70] have been adapted and tailored specifically for sensing data. Additionally, multi-modality sensing has been explored to facilitate cross-domain information fusion [29, 53]. While these approaches have made progress in mitigating the challenges posed by limited labeled data and sensing heterogeneity, they still rely on a moderate fraction of labeled data (approximately 10% $\sim$ 20%) to achieve performance on par with supervised methods. Additionally, incorporating labeled data from diverse sensing domains remains essential for effectively addressing the issue of sensing heterogeneity.

Physics Augmented Learning. The integration of physical information to enhance the learning process has been widely explored

across various communities. In the machine learning field, physics-informed neural networks have been developed to embed physical knowledge, such as differential equation [8, 9, 34] and physics symmetry [13, 14, 60], directly into the learning process. However, most existing approaches aim to augment and regularize the learning process using predefined physical rules, rather than achieving fully label-free training. Furthermore, these methods are often evaluated on simplified synthetic datasets, limiting their applicability to more complex, real-world sensing scenarios.

In mobile sensing research, recent efforts have also been made on utilizing physical information to enhance learning-based sensing tasks. Researchers have explored augmenting IMU data by synthesizing complementary data from various sources, such as physical knowledge [48, 69], video clips [35], images [80], multi-sensor correlations [83], and text-to-motion synthesis models [40]. These data augmentation techniques, when combined with existing labeled datasets, have demonstrated improved performance in sensing tasks and enhanced cross-dataset generalization. However, most of the existing work is limited to human activity recognition classification tasks and still heavily depends on labeled datasets.

IMU Sensing. IMU sensing tasks, such as inertial tracking and motion capture, have been a focus of study in the community for decades [4, 5, 38, 46, 49, 59, 61, 62, 64, 66, 88]. These techniques have been widely applied across diverse domains, including indoor localization [86], gesture recognition [30], autonomous systems [2], fitness [31], context-aware interfaces [3], and rehabilitation [50]. For these impactful applications, our goal is to enable label-free training and facilitate large-scale deployment on IoT devices.

3 Physical Self-Supervised Learning

This section presents our physical self-Supervised learning framework for IMU sensing tasks. We first introduce the overall paradigm in Section 3.1. Section 3.2 explains how we disentangle sensor motion from object motion. Section 3.3 describes how the multi-view kinematic tree leverages sparse physical self-supervision, and Section 3.4 details our uncertainty-aware physical self-Supervised learning.

3.1 Basic Physical Self-Supervised Paradigm

3.1.1 Framework Overview. Although our proposed physical self-Supervised learning framework still follows an auto-encoder-like structure, this section describes how its three key components—the encoder, decoder, and loss function—are redesigned in our new learning paradigm to enable sensing without manual labels. As illustrated in Figure 1, the input is a temporal sequence of 3D IMU measurements from one or more sensors, which is first processed by the hybrid IMU encoder to infer both the target physical states (e.g., joint motions) and an environment-aware representation of the sensing setup. These outputs are then passed to the Auto-Adaptive Physics Decoder, where the environment-aware representation is further transformed by CondProj layers into learnable environment variables, such as bone lengths and sensor-related parameters. Together with the predicted physical states, these variables are used in parameterized physical equations to reconstruct the corresponding IMU measurements. To reduce the effect of sensor noise, the reconstructed sequence is fed into the encoder again, and the reconstruction objective is enforced in the intermediate latent space rather than directly in the raw data space. Through this training process, the model learns to produce physical states and environment variables that are both physically consistent and robust to real-world sensing noise. During inference, the auto-adaptive physical decoder is no longer needed, while the encoder directly predicts the target physical states and environment variables from the input IMU sequences.

3.1.2 Auto-Adaptive Physics Decoder. Unlike common autoencoder families (e.g., variational autoencoders and masked autoencoders), which typically rely on fully learnable DNN decoders to map the latent space back to the data space and often produce latent representations that are difficult to interpret, our auto-adaptive physics decoder instead uses explicit, parameterized physical equations that automatically adapt to the data to model a family of kinematic systems. The parameters of these equations are conditioned on the input IMU readings. This design removes the need to perfectly specify a single underlying physical model and allows physical self-supervised learning to focus on adapting to different sensing environments rather than merely fitting a fixed set of known dynamics.

For clarity, we first consider the IMU kinematics of a point mass, which we use to describe reference-frame transformations between the sensed object and the IMU, given by the following system of ODEs:

$$\begin{cases} \frac{d}{dt}p^{(\text{imu})}(t) = v^{(\text{imu})}(t) \\ \frac{d}{dt}v^{(\text{imu})}(t) = R^{(\text{imu})}(t)a(t) + g \\ \frac{d}{dt}R^{(\text{imu})}(t) = R^{(\text{imu})}(t)\hat{\omega}(t) \end{cases} \tag{1}$$

where $p^{(\text{imu})}(t)$ and $v^{(\text{imu})}(t)$ denote the position and velocity expressed in the global frame of reference (GFR); $R^{(\text{imu})}(t)$ is the orientation, represented as a rotation matrix that maps vectors from the local frame of reference (LFR) to the GFR; $a(t)$ is the accelerometer measurement; g is the gravity vector; and $\hat{\omega}(t)$ is the skew-symmetric matrix corresponding to the gyroscope measurement.

To construct the auto-adaptive physics decoder, we derive a discrete-time formulation suitable for sampled IMU data by discretizing the continuous-time dynamics in (1). Let $t_k = k\Delta t$ and denote $p_k^{(\text{imu})} = p^{(\text{imu})}(t_k)$, $v_k^{(\text{imu})} = v(^{(\text{imu})}t_k)$, $R_k^{(\text{imu})} = R^{(\text{imu})}(t_k)$, $a_k = a(t_k)$, and $\hat{\omega}_k = \hat{\omega}(t_k)$. Using a first-order (Euler) approximation, we obtain

$$\begin{cases} p_{k+1}^{(\text{imu})} = p_k^{(\text{imu})} + v_k^{(\text{imu})}\Delta t, \\ v_{k+1}^{(\text{imu})} = v_k^{(\text{imu})} + \left(R_k^{(\text{imu})}a_k + g\right)\Delta t, \\ R_{k+1}^{(\text{imu})} = R_k^{(\text{imu})}\exp\left(\hat{\omega}_k\Delta t\right), \end{cases} \tag{2}$$

where $\exp(\cdot)$ denotes the matrix exponential. For small Δt, we further approximate $\exp(\hat{\omega}_k\Delta t) \approx I + \hat{\omega}_k\Delta t$.

Therefore, the point-mass forward kinematics can be written as

$$[a, \omega] = f_{\text{imu}}(p^{(\text{imu})}, R^{(\text{imu})}) \tag{3}$$

where $f_{\text{imu}}(\cdot)$ denotes the differentiable forward IMU model that maps point-mass kinematics to IMU readings. This model forms one component of our physics decoder, which is implemented based on the discrete-time dynamics in (2).

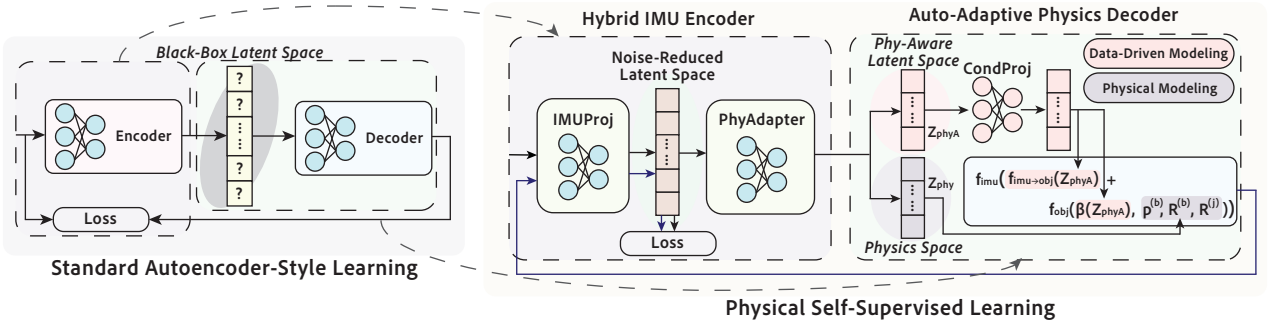


Figure 1: Physical Self-Supervised Learning.

In practice, however, many IMU-sensing tasks involve objects with substantially more complex kinematic structure, such as human arms or full-body motion, where each IMU measurement reflects the coupled dynamics of an articulated rigid-body system rather than a single point mass. We model such an articulated object as a kinematic tree [68]: a tree-structured assembly of rigid links connected by joints and rooted at a global base joint. As an illustrative example, we use the SMPL kinematic tree [47] for human pose and motion capture with IMUs. SMPL is a parametric 3D human body model in which body shape is represented by shape parameters $\vec{\beta} \in \mathbb{R}^{10}$, given as coefficients of a PCA body-shape basis. Body pose is defined by the relative 3D rotations of the 23 non-root joints in the kinematic tree. Let $\{R^{(j)}\}_{j=1}^{23}$ denote these joint rotations; in practice, we parameterize them using a 6D continuous representation rather than a 3D minimal one to improve continuity in the rotation space [89]. The global translation and orientation of the root joint (i.e., the pelvis) are represented by $\{p^{(b)}, R^{(b)}\}$.

There exists a differentiable forward-kinematics mapping for the SMPL human pose model (and, more generally, other articulated objects) that yields the positions and orientations of all joints in the GFR:

$$[p^{(j)}, R^{(j)}] = f_{\text{obj}}(\beta, p^{(b)}, R^{(b)}, \{R^{(j)}\}) \tag{4}$$

where $z_{\text{phy}} = [p^{(b)}, R^{(b)}, \{R^{(j)}\}_{j=1}^{23}]$ denotes the desired kinematic quantities of the object of interest, explicitly produced by the "physical-space" branch of the hybrid IMU encoder. The object-specific parameters $\vec{\beta}(Z_{\text{phyA}})$ (e.g., bone lengths in the human SMPL model) are adapted by a learnable DNN, CondProj, whose outputs depend on the physics-aware latent feature Z_{phyA}, thereby making inference over object parameters explicitly conditioned on the current IMU inputs.

Unfortunately, purely physics-based modeling cannot resolve everything, because the IMU placement and its relative motion with respect to the object of interest are generally unknown and are nearly impossible to model explicitly (e.g., a phone's movement in a pocket or bag, or a smartwatch's motion on a wrist over time). Rather than hand-crafting explicit parameterized models for these factors, we adopt a data-driven approach that directly predicts the IMU placement and its relative motion with respect to a joint using a learnable DNN, CondProj, which takes the physics-aware latent feature Z_{PhyA} as input:

$$[p^{(\text{imu}\to\text{obj})}, R^{(\text{imu}\to\text{obj})}] = f_{\text{imu}\to\text{obj}}(Z_{\text{PhyA}}) \tag{5}$$

We detail how the IMU placement and its relative motion are parameterized with DNNs, and how this relative motion is disentangled from the object's movement, in Section 3.2. Therefore, the IMU's motion in the GFR is given by

$$\begin{aligned}[p^{(\text{imu})}, R^{(\text{imu})}] =& f_{\text{imu}\to\text{obj}}(Z_{\text{PhyA}}) + \\ & f_{\text{obj}}(\beta(Z_{\text{PhyA}}), p^{(b)}, R^{(b)}, R^{(j)})\end{aligned} \tag{6}$$

which corresponds to composing the object's motion (4) with the learned IMU's motion relative to the object (5). As shown in Figure 1, by further transforming the IMU's motion into the LFR using the forward IMU model (3), we thus obtain our auto-adaptive physics decoder.

3.1.3 Hybrid IMU Encoder. Although our auto-adaptive physics decoder enables interpretable IMU sensing without manual labels and uses data-driven modeling to avoid inaccurate and labor-intensive hand-crafted physical models, standard autoencoder-style training still relies on reconstruction error in the data space as its primary supervision signal. IMU measurements are notoriously noisy, which makes a purely data-space reconstruction loss much less effective.

Our key idea is therefore to decompose the encoder into two cascaded components: the first produces a noise-reduced latent space that preserves task-relevant information, and the second maps this latent representation into a physics-aware space tailored to the auto-adaptive physics decoder. By enforcing reconstruction in the denoised latent space rather than the raw IMU data space, we mitigate the impact of sensor noise. We refer to these two encoder components as IMUProj and PhyAdapter, respectively.

To obtain a noise-reduced latent space, we first pre-train IMUProj using the standard masked autoencoder recipe, a strategy that has proven effective for denoising and representation learning on a variety of time-series data and tasks [6, 41, 70]. We adopt a miniature ViT architecture [87], reducing the depth from 12 to 6 transformer blocks (12M parameters), and make minor adaptations for IMU data, including masking along both the temporal and sensor dimensions and applying positional embeddings only in the temporal domain. We deliberately omit sensor-wise positional embeddings because sensor placement is not known a priori (e.g., the IMU may be in a phone in a pocket or a watch on a wrist). PhyAdapter is implemented as a shallow MLP that maps features from this noise-reduced latent space into the physics and physics-aware spaces.

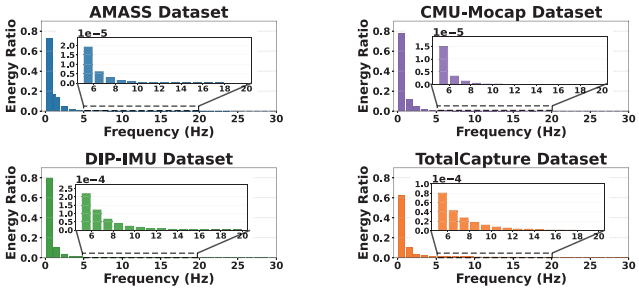


Figure 2: Frequency energy distribution of human motion across four widely used public datasets.

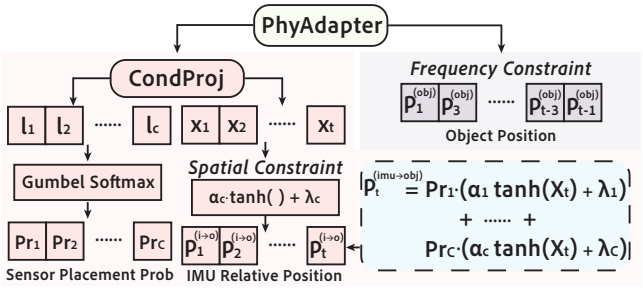


Figure 3: Probabilistic frequency–spatial constraints for motion disentanglement.

During physical self-supervised learning, PhyAdapter is trained jointly with our physics decoder, while IMUProj remains frozen.

3.2 Disentangling IMU and Object Motion

During IMU sensing, we typically have no prior knowledge of where the sensors are attached, their initial position and orientation, or how they move relative to the object over time. In the previous section, we modeled this IMU–object relative motion using the learned forward mapping $f_{\text{imu}\to\text{obj}}(\cdot)$ obtained via a data-driven approach. However, data-driven learning is not a panacea: it remains challenging to disentangle the sensor's relative motion from the underlying object motion during training.

To address this challenge, we first structurally decompose the sensor's relative motion. In typical scenarios, multiple sensors of known device types (e.g., smartphone, smartwatch, earbuds) operate jointly. Each device type is associated with a small set of plausible operating regions on the body. For example, a smartwatch may be worn on the left or right wrist, and a smartphone may be carried in the left/right pocket, held in the left/right hand, or placed in a backpack. We refer to the discrete choice over these regions as the *sensor placement*.

Each candidate placement is associated with an anchor joint in the kinematic-tree model, denoted by ϕ_m for the m-th IMU. For instance, the left/right wrist joint for a smartwatch on the left/right wrist, and the left/right hip joint for a smartphone in the corresponding trouser pocket. Given a sensor placement with anchor joint ϕ_m, we decompose the relative motion of the m-th IMU (denoted imu_m) into two components: (i) the initial relative kinematic state (position and orientation) with respect to ϕ_m, $[p_0^{(\text{imu}_m\to\phi_m)}, R_0^{(\text{imu}_m\to\phi_m)}]$, and (ii) the time-varying motion relative to this initial state, $[\Delta p_k^{(\text{imu}_m\to\phi_m)}, \Delta R_k^{(\text{imu}_m\to\phi_m)}]$. Thus, the kinematic state of imu_m in the global reference frame at time step k can be written as follow, taking the positional part as an example:

$$p_k^{(\text{imu}_m)} = p_0^{(\text{imu}_m\to\phi_m)} + \sum_{\tau=1}^{t} \Delta p_\tau^{(\text{imu}_m\to\phi_m)} + p_k^{(\phi_m)} \tag{7}$$

where $p_k^{(\phi_m)}$ is obtained from the kinematic-tree forward-kinematics function $f_{\text{obj}}(\cdot)$.

However, the core challenge is to disentangle the sensor's relative motion from the human body motion. Our key insight is that these two types of motion are more naturally separable in the joint frequency–spatial domain. On the one hand, human motion is typically band-limited, as established in biomechanics and kinesiology [7, 32, 51]. We further confirm this empirically by analyzing several open motion-capture datasets with diverse ground-truth human motion trajectories. As shown in Figure 2, across all these datasets a 25 Hz cutoff frequency captures more than 99% of the human motion energy.

On the other hand, the sensor's relative motion is spatially bounded. For example, for a smartwatch worn on the wrist, rotation about the wrist joint is typically limited to about 30°; displacement away from the forearm surface is on the order of a finger's width (e.g., ∼1 cm); and sliding motion along the forearm is confined to a small range (e.g., ∼3 cm). Similarly, for a smartphone in a trouser pocket, both its 3D rotation and translation are constrained within a modest angular and spatial range (e.g., ∼ 40° and a few centimeters).

In summary, human body motion is frequency-band limited but not strongly spatially constrained (beyond the natural link limits of the SMPL kinematic tree), whereas the sensor's relative motion is spatially limited but not necessarily band-limited. As shown in Figure 3, by explicitly encoding these frequency–spatial priors in the output heads of the PhyAdapter and CondProj networks, we substantially reduce the learning complexity and allow physical self-Supervised training to more naturally disentangle object motion from sensor motion.

We enforce the frequency constraint by limiting the output rate over time according to the Nyquist criterion—producing outputs at 50 Hz for a 25 Hz cutoff frequency, while the raw IMU is sampled at 100 Hz. Spatial constraints are imposed via bounded activations, typically a scaled and shifted tanh of the form $\alpha \cdot \tanh(\cdot) + \lambda$, which restricts predicted translations and rotations to plausible ranges.

However, some of the aforementioned design choices (i.e., the parameters of the spatial constraints and the anchor joint) depend on the sensor placement, which is itself unknown a priori. We therefore treat sensor placement as a learnable variable that is automatically adapted by the corresponding CondProj head in the physics decoder. Consequently, the sensor's relative motion $f_{\text{imu}\to\text{obj}}(\cdot)$ is modeled in a probabilistic, expectation-based formulation. Given each input IMU with a known device type (e.g., watch, phone, or earbuds), CondProj outputs a set of logits over the corresponding candidate placements. To obtain a probability distribution over

Sensor	Placement	Rotation	Translation
Earbud	Designated ear	Fixed	Fixed
Watch	Left wrist	±30° (wrist axis)	±1 cm (away from forearm), ±3 cm (along forearm)
	Right wrist		
Phone	Left hand	0°	±0.1 cm
	Right hand		
	Left trouser pocket	±40° (3 axes)	±3 cm
	Right trouser pocket		
	Backpack	Unlimited	±10 cm

Table 1: Default spatial limitations and configurations for each sensor type and its candidate placements.

sensor placements $\{\mathrm{Pr}_c\}$, we apply the Gumbel–Softmax reparameterization [26] rather than a standard softmax, which encourages approximately discrete placement selection/sampling instead of "averaging" over all candidates, and likewise improves the subsequent relative-motion estimation. Moreover, we make the spatial-constraint parameters $\{\alpha, \lambda\}$ in the bounded activation functions depend explicitly on the inferred placement, enabling more fine-grained constraints instead of having to accommodate the worst case (e.g., an IMU in a backpack). The default configurations are listed in Table 1 and are used for all experiments. These constraints are chosen based on empirical experience rather than tuned for optimality; deriving tighter, data-driven bounds from large-scale statistics is left for future work. Putting this together, the IMU's relative motion is estimated as follows, illustrated here for the translational component at time step k.

$$\begin{aligned} \Delta p_{k,c}^{\mathrm{imu}_m \to \phi_{m,c}} &= \alpha_c \cdot \tanh(\Delta x_k^{\mathrm{imu}_m}) + \lambda_c \\ \Delta p_k^{\mathrm{imu}_m \to \mathrm{obj}} &= \sum_{c=1}^{C} \mathrm{Pr}_c \cdot \Delta p_{k,c}^{\mathrm{imu} \to \phi_{m,c}} \end{aligned} \tag{8}$$

where $\Delta x_k^{\mathrm{imu}_m}$ is the CondProj output for $\Delta p_k^{\mathrm{imu} \to \mathrm{obj}}$ before the activation function, and $\phi_{m,c}$ denotes the anchor joint associated with imu_m under the c-th placement. In this way, the IMU–object relative-motion function $f_{\mathrm{imu} \to \mathrm{obj}}(\cdot)$ is constructed so that sensor motion is naturally disentangled from object motion through the imposed probabilistic frequency–spatial constraints.

3.3 Multi-View Kinematic Tree

Even with known sensor deployments and sensors tightly attached to the object of interest, IMU-based tasks such as full-body motion capture with only 2–6 IMUs remain highly challenging due to their inherently ill-posed nature, i.e., the number of kinematic states of interest far exceeds the number of sensed states. This difficulty is further amplified by our goal of achieving such sensing without any manual labels.

Empirically, we find that directly using a standard kinematic tree as the object/human forward kinematics model $f_{\mathrm{obj}}(\cdot)$ leads to poor performance, especially when fewer than 6 IMUs are available. The key reason is that physical self-supervised learning imposes a very different supervision pattern from conventional fully supervised training or fine-tuning. In a standard kinematic tree, multiple kinematic chains are formed from the root joint (e.g., the pelvis) to the leaf joints. Under supervised learning, every joint is directly supervised, so each kinematic chain receives dense learning signals along its entire length. In contrast, under physical self-supervised learning, supervision is available only at anchor joints where IMUs

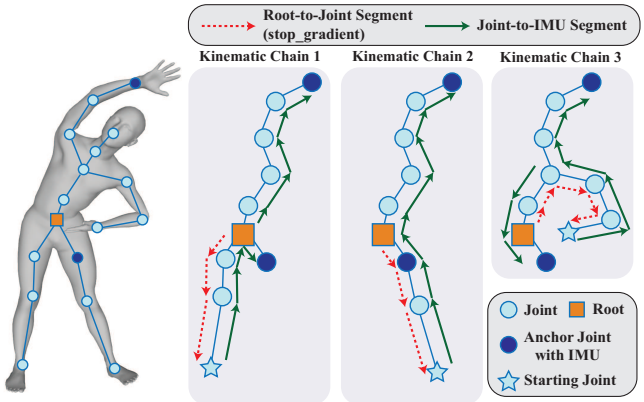


Figure 4: Multi-view kinematic tree. Three "detour" kinematic chains are shown, originating from the right foot, left foot, and left wrist.

are attached. As a result, many joints may receive little or no effective supervision—particularly when anchor joints are not leaves, or when some kinematic chains contain no anchor joints at all, which is common when using fewer than six IMUs for human pose estimation.

To address this, we propose a *multi-view kinematic tree* with an associated forward-kinematics function that more effectively exploits and propagates physics-based supervision to all joints. Unlike a conventional kinematic tree, which defines kinematic chains only from the root to the leaves, the multi-view kinematic tree constructs chains from every joint to every IMU-anchored joint. Thus, for a model with N joints and M attached IMUs, the multi-view kinematic tree yields $N \times M$ kinematic chains, each terminating at an IMU anchor joint.

We denote $f_{\mathrm{obj}}^{(j,\phi_m)}(\cdot)$ as the forward-kinematics mapping along the chain from joint j to anchor joint ϕ_m in the global reference frame (GFR). A issue is that the global kinematic state (i.e., the object's global translation and orientation over time in the GFR) is typically represented only at the root joint. Naïvely replicating global states at every joint would introduce redundant variables and additional consistency constraints. Instead, we introduce a "detour" kinematic chain that is split into two segments: the first segment runs from the root to joint j, denoted $f_{\mathrm{obj}}^{(\mathrm{root} \to j)}(\cdot)$, and the second segment from joint j to ϕ_m, denoted $f_{\mathrm{obj}}^{(j \to \phi_m)}(\cdot)$. The first segment is used to express the global kinematic state from the perspective of joint j.

However, naively using this two-segment chain still poses challenges for supervision propagation. As illustrated in Figure 4, the root-to-joint segment (shown in red) can partially overlap, with opposite direction, the joint-to-IMU segment. Since this forward mapping is used as part of a differentiable computation graph, gradients flowing along these overlapping segments can cancel out, weakening the effective supervision. To address this, we insert a `stop_gradient` operation on the first segment: the root-to-joint path is used only to propagate the global kinematic state in the forward pass, but is excluded from gradient backpropagation. Note that when the joint coincides with the root, neither the "detour" kinematic chain nor the `stop_gradient` segment is needed;

the global kinematic state can still be directly and correctly supervised. Therefore, the multi-view forward-kinematics function $f_{\text{obj}}(\beta, p^{(b)}, R^{(b)}, \{R^{(j)}\})$ is defined as

$$f_{\text{obj}}^{(j,\phi_m)}(\cdot) = \texttt{stop\_grad}(f_{\text{obj}}^{(\text{root}\to j)}(\cdot)) + f_{\text{obj}}^{(j\to\phi_m)}(\cdot) \tag{9}$$

yielding a total of $N \times M$ kinematic chains for an N-joint, M-IMU system and enabling sparse physics-based supervision to be stably propagated to all parameters of the object model.

3.4 Uncertainty-Aware Formulation

Despite the above efforts, IMU sensing remains challenging due to the underdetermined nature of the problem when only a limited number of IMUs are available. A single IMU time series can correspond to multiple, distinct human motion trajectories, making the solution inherently ambiguous. This uncertainty is intrinsic and cannot be ignored. Moreover, it rarely follows a Gaussian distribution and is often multi-modal or highly irregular. To address this, we explicitly model joint rotations in the object model as probability distributions and adopt an uncertainty-aware formulation in constructing the multi-view kinematic chains.

While Gaussian distributions are the most common choice in DNN-based formulations [33], they are restricted to unimodal behavior. Instead, we propose to directly learn the underlying distribution $P(x)$ without imposing a strong prior. For a random variable x with support $[x_0, x_n]$, its expectation is $\bar{x} = \int_{-\infty}^{+\infty} x \cdot P(x)dx = \int_{x_0}^{x_n} x \cdot P(x)dx$. To obtain a tractable formulation, we discretize this range into evenly spaced points $x_0, x_1, \ldots, x_n$ with interval Δ. Under the discrete constraint $\sum_{i=0}^{n} P(x_i) = 1$, the estimated regression value becomes $\bar{x} = \sum_{i=0}^{n} x_i \cdot P(x_i)$. In other words, a regression task can be reformulated as a classification problem over discretized bins, enabling a flexible distributional representation without imposing restrictive parametric priors.

To enable uncertainty-aware multi-view kinematic chains, we replace all kinematic states $\{\beta, p^{(b)}, R^{(b)}, \{R^{(j)}\}\}$ in the forward-kinematics function $f_{\text{obj}}(\cdot)$ with the discrete distribution formulation described above. Consequently, all multi-view kinematic-chain equations in (9) are reformulated in a probabilistic manner, yielding a stochastic process reminiscent of a Kalman filter that propagates distributions over the state space. Although the transition equations in the multi-view kinematic chains are linear, the resulting distributions are non-Gaussian. To handle this, we borrow the idea of particle filtering and employ Monte Carlo sampling to realize uncertainty-aware multi-view kinematic chain generation.

However, Monte Carlo sampling from discrete distributions is not differentiable. To overcome this, we again use the Gumbel–Softmax reparameterization trick [26], which provides a differentiable approximation to one-hot categorical samples and thus preserves end-to-end trainability.

4 Experiment Setup

4.1 Data Collection

Data Collection and Wearing Conditions. We recruited four participants. In the Motion Capture (MoCap) setting, we instrumented six limb segments with Xsens IMUs. Participants also wore a smartwatch on the wrist, carried a smartphone in a hip pocket, and used an earbud on the head. Ground-truth motion was recorded with four Kinect v2 sensors positioned around the capture volume.

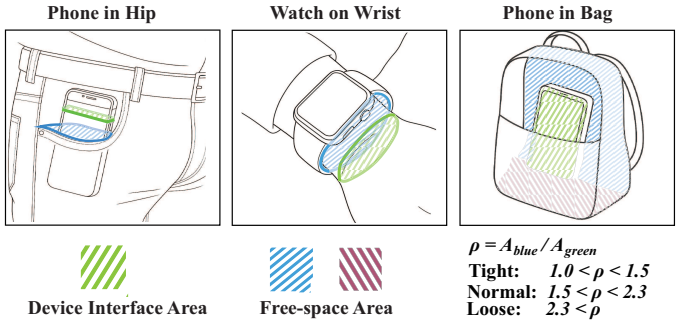


Figure 5: Definition of wearing tightness. On a cross-sectional plane, free-space area (A_{blue}) and device–interface area (A_{green}) define $\rho = A_{\text{blue}}/A_{\text{green}}$, categorized as Tight ($1.0 < \rho < 1.5$), Normal ($1.5 < \rho < 2.3$), or Loose ($\rho > 2.3$).

In the Inertial Tracking setting, the participants carried a smartphone and IMU streams were recorded as participants followed predefined paths. The waypoint timestamps provided reference positions for temporal and spatial alignment.

To capture how sensor motion affects downstream sensing performance, we also collect data under varying wearing tightness conditions. As illustrated in Fig. 5, each placement is associated with a cross-sectional plane on which we define a free-space region F (blue) and a device–interface region D (green). This plane corresponds to the pocket opening for the phone-at-hip case, the wristband loop for the watch, or the supporting internal face for the backpack. The looseness ratio $\rho = A_F/A_D$ is computed on that plane and used to categorize wearing conditions as tight, normal, or loose.

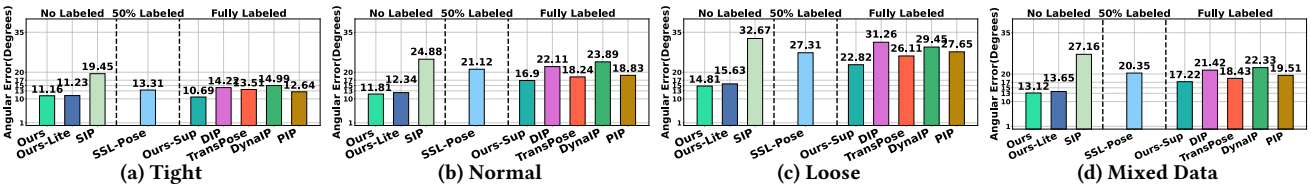
4.2 Baseline Models and Datasets

We compare our approach with three types of baselines: (1) physics-based optimization methods that use no labels, (2) self-supervised methods that use partial labels, and (3) fully supervised methods that rely on complete labels.

4.2.1 *Motion Capture.*

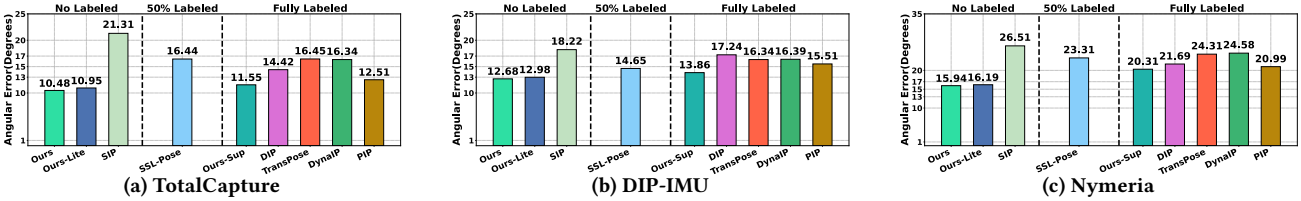
- **Sparse IMU Pose (SIP) [66]: Label-free physics-based optimization that fits a statistical body model to IMU readings to recover motion.**
- **SSL-Pose** [16]: Self-supervised Transformer, pretrained with a masked autoencoder objective and then fine-tuned.
- **Physical Inertial Poser (PIP)** [78]: Fully supervised RNN with an integrated physics-based motion optimizer.
- **TransPose** [79]: Fully supervised RNN with multi-stage inference.
- **Dynamic Inertial Poser (DynaIP)** [84]: Fully supervised biRNN with part-based pose estimation.
- **Deep Inertial Poser (DIP)** [25]: Fully supervised biRNN with synthetic IMU data augmentation.

Other than our own data collection, we also evaluate on three public MoCap datasets, DIP-IMU, TotalCapture, and Nymeria, a large-scale daily-activity dataset, which provide synchronized IMU streams and ground-truth poses over diverse motions. We report four standard metrics: (1) SIP Error: mean orientation error of upper arms and legs



(a) Tight (b) Normal (c) Loose (d) Mixed Data

Figure 6: Angular error across different tightness conditions. All other baselines assume ground-truth sensor placements and skeleton parameters.



(a) TotalCapture (b) DIP-IMU (c) Nymeria

Figure 7: Angular error on datasets with 6 sensors.

Method	Our Data				TotalCapture				DIP-IMU				Nymeria				Label
	SIP Err(°)	Ang Err(°)	Pos Err(cm)	Mesh Err(cm)	SIP Err(°)	Ang Err(°)	Pos Err(cm)	Mesh Err(cm)	SIP Err(°)	Ang Err(°)	Pos Err(cm)	Mesh Err(cm)	SIP Err(°)	Ang Err(°)	Pos Err(cm)	Mesh Err(cm)	
SIP	33.84	27.16	21.43	23.91	23.29	21.31	12.62	14.85	23.45	18.22	13.62	14.37	31.45	26.51	23.59	20.46	No
SSL-Pose	24.90	22.54	18.62	20.25	29.90	24.54	13.46	16.25	26.64	23.60	10.48	15.72	31.21	27.54	22.33	24.61	10%
	21.62	20.46	12.32	14.36	26.45	21.46	10.32	13.36	26.45	22.33	9.11	12.88	27.52	24.10	20.38	19.65	20%
	21.32	20.35	10.11	12.56	19.32	16.44	8.13	9.85	16.46	14.65	7.11	9.45	24.19	23.31	14.69	16.88	50%
DIP	23.58	21.42	10.95	12.04	16.58	14.42	8.44	9.04	18.42	17.24	9.16	11.97	25.44	21.69	12.51	17.63	Full
DynaIP	24.55	22.33	13.51	11.27	19.86	16.55	12.44	12.21	17.55	16.34	6.51	7.27	27.30	24.58	16.92	18.44	Full
TransPose	19.25	18.43	9.64	10.91	14.31	16.45	8.64	7.91	15.84	16.39	8.44	9.39	30.98	24.31	18.46	19.45	Full
PIP	22.44	19.51	11.68	9.45	14.44	13.51	7.68	8.45	16.03	15.51	9.86	10.82	26.79	20.99	14.51	13.24	Full
Ours-Sup	19.35	17.22	9.69	9.32	15.01	11.55	7.89	8.31	8.59	13.86	7.99	9.09	23.04	20.31	16.32	15.96	Full
Ours-Lite	15.69	13.65	7.58	8.90	12.69	10.95	7.31	6.59	13.91	16.54	10.06	12.31	17.64	16.19	9.56	10.44	No
Ours	15.15	13.12	6.84	7.45	12.66	10.48	5.91	5.99	11.13	12.68	4.69	5.62	16.91	15.94	8.95	9.63	No

Table 2: MoCap results with 6 IMUs on three datasets.

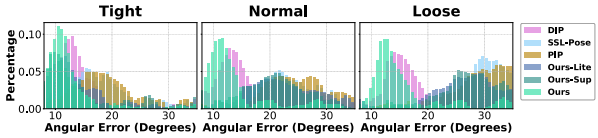


Figure 8: Angular error distributions shown for tight, normal, and loose wearing conditions.

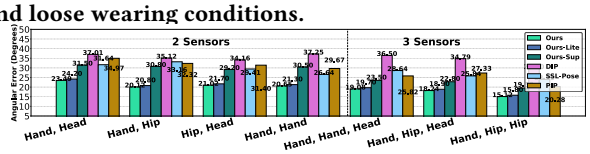


Figure 9: Angular errors under popular sensor placement combinations.

in the global frame (°); (2) Angular Error: mean joint rotation deviation (°); (3) Position Error: mean Euclidean joint-position distance (cm); (4) Mesh Error: mean vertex distance between reconstructed and ground-truth meshes (cm).

4.2.2 Inertial Tracking.

- **MUSE** [61]: Label-free, sensor-fusion method that prioritizes magnetometer measurements over gravity for orientation tracking.
- **PDR** [28]: Label-free, step-based pedestrian dead reckoning approach.
- **EKF**: Label-free, classical kalman filtering-based method that recursively estimates position and orientation.
- **LIMU-BERT** [70]:Self-supervised BERT-like architecture with pre-training and fine-tuning.
- **SSLHAR-LOC** [57]: Self-supervised CNN, pretrained with a masked autoencoder objective and then fine-tuned.
- **RoNIN** [24]: Fully supervised hybrid LSTM/TCN model.
- **CTIN** [58]: Fully supervised hybrid ResNet/Transformer.
- **TLIO** [45]: Fully supervised hybrid LSTM/TCN integrated with an Extended Kalman Filter.
- **IONet** [11]: Fully supervised RNN model.

Other than our own data collection, we also evaluate on two public datasets, **SHL** [1] and **OxIOD** [12], which cover diverse motion patterns and environments for evaluating generalization. We use *Location Error* as evaluation metric.

4.2.3 Our Variants. We additionally include two variants of our method for comprehensive comparison:

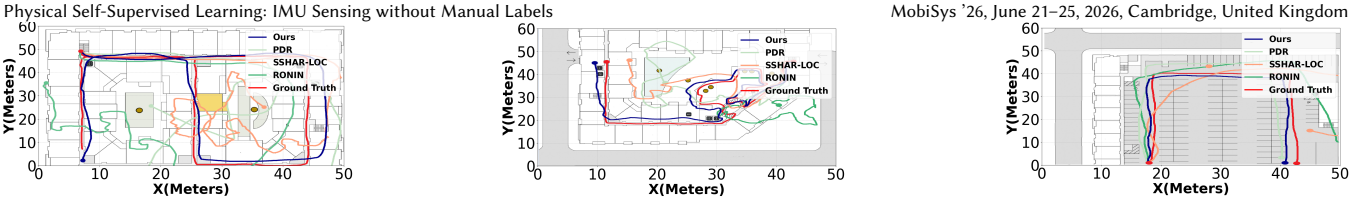
- **Ours-Sup**: A supervised variant of our model trained fully with pose labels (without sensor relative motion modeling).
- **Ours-Lite**: A lightweight compressed variant for efficient inference on embedded devices, obtained by reducing Transformer blocks from 6 to 3, attention heads from 8 to 4, latent dimension from 512 to 256, and using FP16 instead of FP32.

5 Evaluation

In this section, we evaluate our method on two tasks: motion capture and inertial tracking. We first present overall performance, then examine cross-domain generalization. We next provide an ablation study of our physical self-supervised learning, and finally report execution time and results on downstream task adaptation.

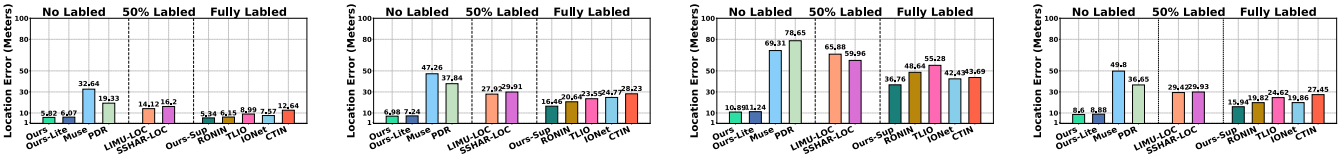
5.1 Overall Performance

5.1.1 Motion Capture Performance. We evaluate our framework on both our self-collected dataset and several public MoCap benchmarks. On our dataset, Figure 6 shows that the performance gap is small under tight attachment, but becomes much larger under normal and loose conditions. While our supervised variant is slightly better in the tight setting, supervised methods deteriorate sharply



(a) Walking Outdoor (b) Running Indoor (c) Running Outdoor

Figure 10: Inertial tracking trajectory estimation. Baselines diverge within minutes due to accumulated error.



(a) Tight (b) Normal (c) Loose (d) Mixed Data

Figure 11: Location error across different tightness conditions.

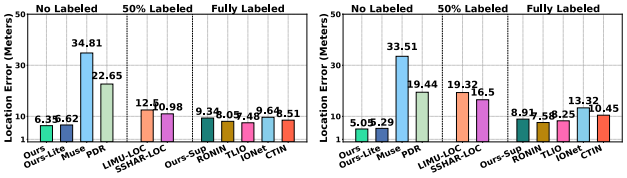
Method	Our Data					SHL					Oxiod					Label
	2mins	6mins	9mins	12mins	15mins	2mins	6mins	9mins	12mins	15mins	2mins	6mins	9mins	12mins	15mins	
Muse	22.42	49.80	66.72	80.04	215.30	15.40	34.81	65.40	151.60	350.70	16.40	33.51	78.81	97.20	254.20	No
PDR	19.25	36.65	36.91	45.00	57.66	11.32	22.65	38.24	46.31	82.26	8.39	19.44	28.89	35.74	77.64	No
LIMU-LOC	36.11	59.98	75.84	98.53	212.00	32.54	64.45	93.42	124.31	251.51	10.76	27.34	65.29	139.28	172.17	10%
	29.15	43.41	78.54	89.32	162.3	23.51	34.87	48.32	66.94	97.54	9.44	26.61	31.58	37.66	61.22	20%
	18.44	29.42	69.37	88.01	165.9	6.31	12.50	18.55	22.60	46.55	7.54	19.32	22.74	38.70	62.61	50%
SSHAR-LOC	39.62	62.47	85.77	144.68	237.54	35.56	75.67	135.24	182.31	348.89	13.57	27.82	39.21	58.94	112.18	10%
	32.24	58.87	76.45	121.55	189.54	27.57	33.24	49.26	87.53	189.4	12.53	27.16	39.78	58.54	98.68	20%
	11.21	29.93	25.26	35.04	49.24	6.56	10.98	16.52	21.64	29.74	6.64	16.50	28.35	35.39	64.94	50%
RONIN	13.64	19.82	24.94	39.40	51.22	5.12	8.05	11.38	14.95	25.11	4.15	7.58	12.14	19.84	29.96	Full
TLIO	17.91	24.62	33.45	65.54	91.95	6.00	7.48	9.46	14.34	19.56	6.01	8.25	12.44	16.25	27.61	Full
IONet	11.45	19.86	12.89	14.88	22.34	6.22	9.64	12.15	15.65	28.81	5.32	13.32	21.46	33.64	39.85	Full
CTIN	12.11	27.45	32.54	41.49	76.84	5.97	8.51	17.64	28.53	33.07	8.32	10.45	23.5	32.0	45.3	Full
Ours-Sup	11.37	15.94	33.24	38.36	49.19	4.98	9.34	18.61	27.45	42.30	5.45	8.91	16.49	27.51	44.72	Full
Ours-Lite	5.12	8.88	10.31	17.86	21.65	4.69	6.62	10.23	13.44	15.51	4.08	5.29	7.06	9.68	12.23	No
Ours	4.45	8.60	9.31	14.02	18.64	4.01	6.35	8.52	9.83	12.55	3.96	5.05	6.32	7.92	9.60	No

Table 3: Inertial tracking results on three datasets

as attachment becomes less controlled, whereas our full framework remains stable. This indicates that the main advantage of our method comes from stronger robustness to sensor motion. On public datasets (Figure 7), sensors are worn more consistently and the performance gap narrows but remains clear, with an even larger gap on Nymeria, where the daily-activity setting is less controlled. Even controlled environments cannot fully eliminate sensor motion, and our label-free approach still outperforms both supervised and self-supervised baselines. Table 2 summarizes results across three benchmarks, where our method achieves the best performance on all metrics, following the same overall trend. Our lightweight variant also shows similar performance, supporting efficient deployment on embedded devices.

Figure 8 further shows angular error distributions across tight, normal, and loose conditions. When errors exceed 25°, motion becomes visually unstable; under loose attachment, SOTA baselines stay below this threshold in fewer than 5% of cases, whereas our framework keeps over 90% of poses within 25° across all regimes. As looseness increases, baselines develop heavy tails, while ours and our lite variant remain compact, indicating much stronger robustness.

We also evaluate sparse-sensor configurations—common in practice when only a phone, smartwatch, or earbud is available. Such minimal setups are ill-posed and highly sensitive to sensor relative motion. As shown in Figure 9, the performance gap between our method and the baselines becomes even larger than in the six-sensor setting: our full model and its Lite variant remain the most robust,



(a) SHL (b) Oxiod

Figure 12: Inertial Tracking comparison on public datasets with 6 minutes trajectory slice.

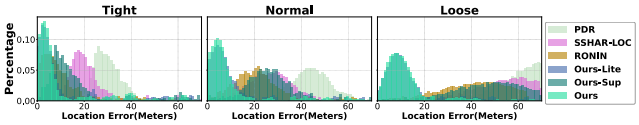
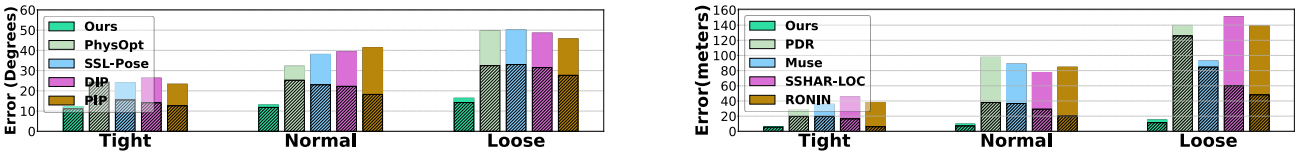


Figure 13: Location error distribution with three tightness conditions.

while the supervised variant no longer does, further confirming the effectiveness of our label-free formulation under sparse and unstable sensing. This robustness comes from explicitly modeling sensor–object relative motion, which limits degradation, and from multi-view propagation, which recovers limb trajectories without direct sensor coverage, providing reliable supervision under sparse and noisy placements.

5.1.2 Inertial Tracking Performance. We evaluate inertial tracking on our dataset and two public benchmarks (Figures 10–11).



(a) Motion Capture (b) Inertial Tracking

Figure 14: Leave-one-condition-out evaluation by wearing tightness. Non-shaded bars show out-of-domain performance when a tightness condition is unseen during training; shaded bars show in-domain performance after fine-tuning with data from that condition.

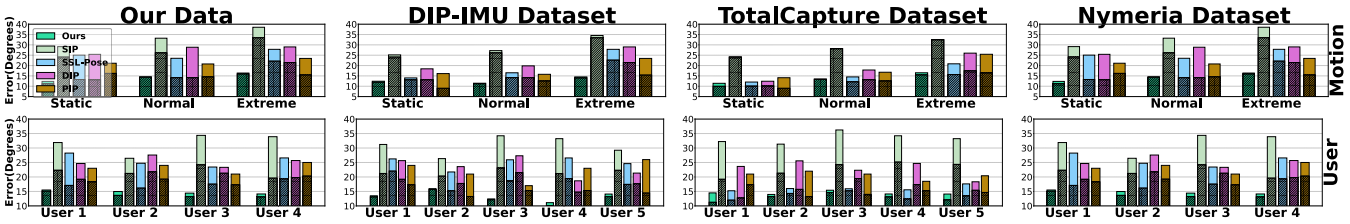


Figure 15: Motion capture results under additional leave-one-scenario-out evaluations, covering both motion and user scenarios. Non-shaded bars show out-of-domain performance on the held-out scenario; shaded bars show in-domain performance after fine-tuning on that scenario.

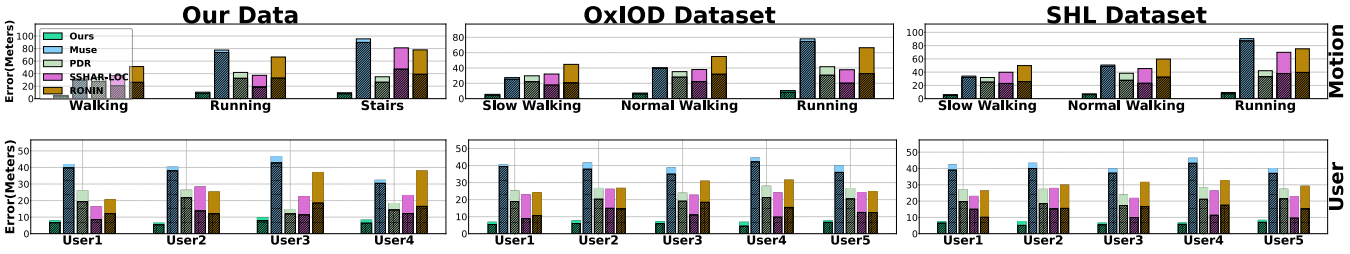


Figure 16: Inertial tracking results under additional leave-one-scenario-out evaluations, covering both motion and user scenarios. Non-shaded bars show out-of-domain performance on the held-out scenario; shaded bars show in-domain performance after fine-tuning on that scenario.

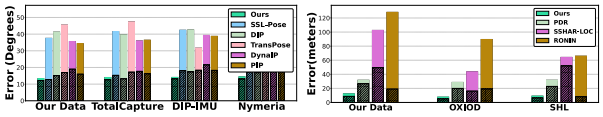


Figure 17: Leave-One-Dataset-Out Training to Evaluate Model Performance in Unseen Environments.

Our method achieves the lowest trajectory error across all tightness conditions, reducing drift by roughly 1.5–2× on six-minute trajectories, despite using no labels. Consistent with the MoCap results, the key gap again appears as attachment becomes less controlled: methods that implicitly rely on rigid attachment degrade rapidly, while our framework remains stable by explicitly predicting sensor–object motion. Classical PDR accumulates substantial drift, and self-supervised baselines improve with labels, yet even with 50% supervision their errors are still nearly 2× ours. Similar trends hold on public datasets. Error distributions (Figure 13) show that fewer than 10% of our trajectories exceed 15 m after six minutes, whereas competing methods have much heavier tails.

5.2 Cross-Domain Generalization Analysis

Real-world deployments involve broad variability in wearing tightness, motion intensity, sensor placement, and user behavior. To evaluate robustness under unseen conditions, we adopt a leave-one-scenario-out protocol in which an entire condition is excluded during training and used only for testing. This provides a direct measure of generalization without additional labels or fine-tuning.

To assess robustness to attachment variability, we first hold out each wearing condition (tight, normal, loose). As shown in Figure 14, *non-shaded bars show out-of-domain performance when a tightness condition is unseen during training; shaded bars show in-domain performance after fine-tuning with data from that condition.* Baseline methods degrade notably, especially under loose attachment, while our method remains stable across all three cases.

For motion capture, we further hold out (i) users and (ii) motion categories. These two settings stress different aspects of generalization. When motion type changes, the underlying physical equations remain valid, but physical-based methods are highly sensitive to noise; our noise-reduced latent space retains the generality of the physics while improving robustness to disturbances. When user

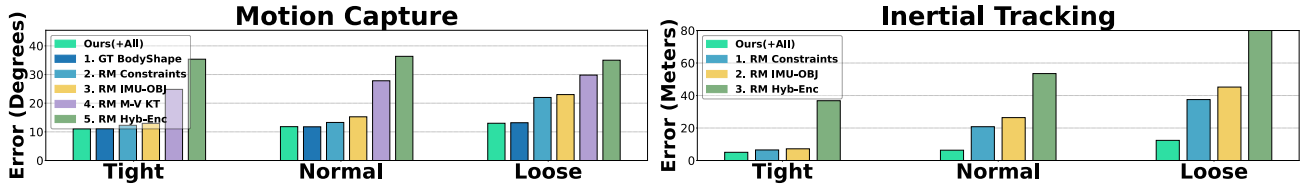
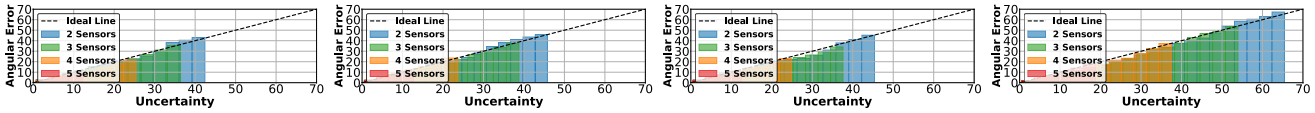


Figure 18: Stepwise ablation of our model under different sensor-attachment conditions. Starting from the full model (Ours(+All)), we (1) replace the learned body-shape module with ground-truth body shape (GT BodyShape), (2) further remove frequency-spatial constraints (RM Constrain), (3) additionally remove IMU–object relative-motion prediction (RM IMU–OBJ), (4) drop the multi-view kinematic tree (RM M-V KT), and (5) finally remove the hybrid encoder (RM Hyb-Enc).



(a) 16 Samples **(b) 8 Samples** **(c) 4 Samples** **(d) 2 Samples**

Figure 19: Output uncertainty vs. predictive error for different sample counts. All cases remain close to the ideal, indicating well-calibrated uncertainty.

identity changes, personalized physical parameters shift; our data-driven modeling infers these parameters directly from the input, enabling adaptation to new subjects. As shown in Figure 15, our method shows minimal degradation across both cases, whereas supervised baselines degrade substantially and still fail to predict unseen users or motions accurately even on the large-scale Nymeria dataset. For inertial tracking, we follow analogous splits over users and motion categories. According to Figure 16, baselines struggle under unseen motions or new users, while our model exhibits consistent performance across all held-out conditions, mirroring the trends observed in motion capture.

Finally, in a leave-one-dataset-out evaluation (Figure 17), prior methods suffer large drops on unseen datasets, whereas our label-free framework maintains significantly higher accuracy, demonstrating robust cross-dataset generalization.

5.3 Ablation Study

5.3.1 Ablation Study of Model Components. In this section, we will perform a ablation study by step-wisely removing each model componments.Starting from the full model (Ours(+All)), we (1) replace the learned body-shape module with ground-truth body shape (GT BodyShape), (2) further remove frequency-spatial constraints (RM Constrain), (3) additionally remove IMU–object relative-motion prediction (RM IMU–OBJ), (4) drop the multi-view kinematic tree (RM M-V KT), and (5) finally remove the hybrid encoder (RM Hyb-Enc). As shown in Figure 18, for the motion capture task, replacing ground-truth body-shape parameters with our predicted ones yields almost identical performance, indicating that the model has successfully captured the underlying complex kinematic structure. Under the tight-attachment setting, removing either the frequency-spatial constraint or the IMU–object relative-motion inference causes only marginal degradation. In contrast, removing the multi-view kinematic tree leaves several joints with little or no supervision, leading to a clear increase in error. Eliminating the hybrid encoder further amplifies this error by preventing effective noise suppression. Overall, the trend indicates that as sensor attachment becomes looser, accurate IMU–object relative-motion inference becomes increasingly critical. For inertial tracking, which lacks a skeletal or kinematic prior, we ablate only the frequency-spatial constraint, the relative-motion modeling, and the hybrid encoder. Under the loose-attachment condition, removing any of these modules again leads to substantial performance degradation.

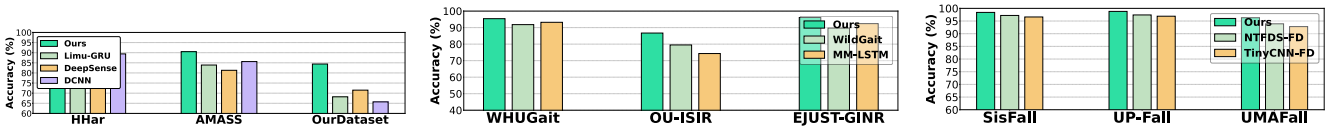
5.3.2 Effect of Uncertainty Sampling Density. To implement our uncertainty-aware multi-view kinematic chain, we use Monte Carlo sampling with particle-filter–style propagation. This raises a practical trade-off: high sampling densities increase computation, while too few samples can destabilize uncertainty estimates. To examine this, we study how sampling density affects reliability.

Figure 19 plots output uncertainty versus angular error under different sampling densities. As the number of samples decreases, the error distribution shifts upward, especially in sparse-sensor settings. However, the calibration curves remain close to the ideal diagonal, indicating that our uncertainty estimates stay reliable over a broad range of sampling densities. Mean error increases substantially only when the sample count drops to two.

5.4 Downstream Task Extension

Once accurate MoCap poses are available, a wide range of downstream tasks become substantially easier. We further evaluate three representative downstream tasks: human activity recognition (HAR), gait recognition, and fall detection. Following common practice in skeleton-based learning, we feed our reconstructed joint sequences into standard downstream backbones, including ST-GCN [72] for HAR, GaitPT [10] for gait recognition, and a lightweight skeleton-based 3D-CNN [52] for fall detection. This setup allows us to directly assess whether our label-free pose estimation provides stable and informative motion representations beyond pose reconstruction itself.

Baselines. We compare against representative task-specific baselines that operate directly on raw IMU signals. For HAR, we use



(a) HAR Task (b) Gait Recognition Task (c) Fall Detection Task

Figure 20: Performance on three downstream tasks: HAR, gait recognition, and fall detection. These results show that the underlying motion predicted by our framework can serve as a generic skeleton representation, allowing simple integration with existing skeleton-based models and straightforward extension to diverse downstream

Cutoff Freq. (Hz)	Ang Err (°)	Spatial Scale (k)	Ang Err (°)
20	13.42	0.50×	13.29
25 (default)	13.12	0.75×	13.08
30	12.98	1.00×	13.12
35	13.51	1.50×	13.31
40	13.46	2.00×	13.79

Table 4: Sensitivity analysis of frequency and spatial priors. The spatial scale k is applied to the default spatial motion bounds specified in Table 1.

LIMU-GRU [70], DCNN [73], and DeepSense [74]. For gait recognition, we use Deep Learning-Based Gait Recognition Using Smartphones in the Wild [90] and Multi-Model Long Short-Term Memory Network [65]. For fall detection, we use NT-FDS [67] and TinyCNN-FD [81].

Results and Comparison. As shown in 20, the overall trend is consistent across all three tasks. On cleaner and more controlled data, direct IMU-based baselines remain competitive. However, under less constrained real-world conditions, their performance degrades much more noticeably, while our method remains robust. This suggests that *if a model cannot disentangle sensor-induced disturbances, even high-level downstream tasks become difficult for neural networks.*

5.5 Sensitivity Analysis of Frequency and Spatial Priors

We further evaluate the sensitivity of the frequency and spatial priors in Table 4. The results show a broad performance plateau, indicating that our framework is not highly sensitive to either hyperparameter. Varying the cutoff frequency from 20–40 Hz changes the angular error only slightly, from 12.98° to 13.51°. Similarly, scaling the spatial bound from 0.5× to 2.0× keeps the error largely stable, ranging from 13.08° to 13.79°, with noticeable degradation appearing only under an overly loose bound of 3.0×.

These results suggest that the proposed priors serve as effective structural regularizers rather than fragile hand-tuned constraints. They provide useful physical guidance while preserving sufficient flexibility across different motions and sensing conditions, further supporting the robustness and practicality of our framework.

5.6 Time Efficiency Evaluation

Table 5 reports the runtime of all methods on an iPhone 16 Pro Max and an ARM Cortex-M7 MCU for a 6-second input window. Ours-Lite runs in 0.032–0.065s on the iPhone and 2.065–3.462s on the MCU for both tracking and mocap, while still achieving the lowest error. These results show that our framework can support real-time deployment on resource-constrained embedded devices.

Mocap			
Model	**iPhone 16 Pro Max**	**ARM Cortex-M7 MCU**	**Error**
Ours-Lite	0.065s	3.462s	13.65°
PIP	0.120s	4.641s	19.50°
SSL-Pose	0.266s	15.329s	27.31°
Tracking			
Model	**iPhone 16 Pro Max**	**ARM Cortex-M7 MCU**	**Error**
Ours-Lite	0.032s	2.065s	8.88m
RoNIN	0.192s	9.645s	19.82m
LIMU	0.044s	5.36s	45.31m

Table 5: Runtime comparison on edge devices.

6 Conclusion

This paper presents a physics self-supervised learning framework for IMU sensing, addressing both inertial tracking and motion capture without requiring manual labels. By combining a learnable physics decoder with a noise-aware neural encoder, our method preserves the structural advantages of physical modeling while improving robustness to sensing noise, placement variation, and user diversity. The framework further incorporates probabilistic frequency–spatial constraints, multi-view kinematic propagation, and uncertainty-aware modeling to better handle the ambiguity and heterogeneity inherent in real-world IMU sensing. Extensive experiments on both self-collected and public datasets show that our method consistently outperforms supervised and self-supervised baselines, especially under realistic and less controlled conditions. In addition, the results on edge devices demonstrate that our lightweight variant can support practical deployment with low runtime overhead. Together, these findings suggest that physics self-supervised learning provides a scalable and practical solution for robust IMU sensing in real-world deployments.

7 Disclaimer

This paper was prepared for informational purposes with contributions from the Global Technology Applied Research center of JPMorgan Chase & Co. (JPMC) and is not a product of its, or its affiliates', Research Departments. JPMC and its affiliates make no representations or warranties, express or implied, regarding the completeness, accuracy, or reliability of the information herein, and accept no liability for its use or any related outcomes. This document does not constitute investment advice, financial research, or a recommendation or offer to buy or sell any security, financial instrument, product, or service.

8 Acknowledgements

This work is partially supported by the National Science Foundation under grants IIS-2107200, and CNS-2038923.

References

[1] 2018. Sussex-Huawei Locomotion and Transportation Dataset. doi:10.21227/7vtt-8c19

[2] Hamad Ahmed and Muhammad Tahir. 2016. Accurate attitude estimation of a moving land vehicle using low-cost MEMS IMU sensors. *IEEE Transactions on Intelligent Transportation Systems* 18, 7 (2016), 1723–1739.

[3] Karan Ahuja, Andy Kong, Mayank Goel, and Chris Harrison. 2020. Direction-of-voice (dov) estimation for intuitive speech interaction with smart devices ecosystems. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 1121–1131.

[4] Eric R Bachmann, Robert B McGhee, Xiaoping Yun, and Michael J Zyda. 2001. Inertial and magnetic posture tracking for inserting humans into networked virtual environments. In *Proceedings of the ACM symposium on Virtual reality software and technology*. 9–16.

[5] Billur Barshan and Hugh F Durrant-Whyte. 1995. Inertial navigation systems for mobile robots. *IEEE transactions on robotics and automation* 11, 3 (1995), 328–342.

[6] Konstantinos Benidis, Syama Sundar Rangapuram, Valentin Flunkert, Yuyang Wang, Danielle Maddix, Caner Turkmen, Jan Gasthaus, Michael Bohlke-Schneider, David Salinas, Lorenzo Stella, et al. 2022. Deep Learning for Time Series Forecasting: Tutorial and Literature Survey. *Comput. Surveys* 55, 6 (2022), 1–36.

[7] Brenda Bigland-Ritchie. 1981. EMG/force relations and fatigue of human voluntary contractions. *Exercise and Sport Sciences Reviews* 9, 1 (1981), 75–118.

[8] Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. 2021. Physics-informed neural networks (PINNs) for fluid mechanics: A review. *Acta Mechanica Sinica* 37, 12 (2021), 1727–1738.

[9] Shengze Cai, Zhicheng Wang, Sifan Wang, Paris Perdikaris, and George Em Karniadakis. 2021. Physics-informed neural networks for heat transfer problems. *Journal of Heat Transfer* 143, 6 (2021), 060801.

[10] Andy Catruna, Adrian Cosma, and Emilian Radoi. 2024. GaitPT: Skeletons are All You Need for Gait Recognition. In *18th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2024)*. IEEE, 1–10.

[11] Changhao Chen, Xiaoxuan Lu, Andrew Markham, and Niki Trigoni. 2018. Ionet: Learning to cure the curse of drift in inertial odometry. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[12] Changhao Chen, Peijun Zhao, Chris Xiaoxuan Lu, Wei Wang, Andrew Markham, and Niki Trigoni. 2018. OxIOD: The Dataset for Deep Inertial Odometry. arXiv:1809.07491 [cs.RO] https://arxiv.org/abs/1809.07491

[13] Taco Cohen and Max Welling. 2016. Group equivariant convolutional networks. In *International conference on machine learning*. PMLR, 2990–2999.

[14] Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. 2018. Spherical cnns. *arXiv preprint arXiv:1801.10130* (2018).

[15] Xiaoran Fan, Longfei Shangguan, Siddharth Rupavatharam, Yanyong Zhang, Jie Xiong, Yunfei Ma, and Richard Howard. 2021. HeadFi: bringing intelligence to all headphones. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. 147–159.

[16] Jack H. Geissinger and Alan T. Asbeck. 2020. Motion Inference Using Sparse Inertial Sensors, Self-Supervised Learning, and a New Dataset of Unscripted Human Motion. *Sensors* 20, 21 (2020). doi:10.3390/s20216330

[17] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

[18] Mahanth Gowda, Justin Manweiler, Ashutosh Dhekne, Romit Roy Choudhury, and Justin D Weisz. 2016. Tracking drone orientation with multiple GPS receivers. In *Proceedings of the 22nd annual international conference on mobile computing and networking*. 280–293.

[19] Tian Hao, Guoliang Xing, and Gang Zhou. 2013. iSleep: unobtrusive sleep quality monitoring using smartphones. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*. 1–14.

[20] Harish Haresamudram, Apoorva Beedu, Varun Agrawal, Patrick L Grady, Irfan Essa, Judy Hoffman, and Thomas Plötz. 2020. Masked reconstruction based self-supervision for human activity recognition. In *Proceedings of the 2020 ACM International Symposium on Wearable Computers*. 45–49.

[21] Harish Haresamudram, Irfan Essa, and Thomas Plötz. 2021. Contrastive predictive coding for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 2 (2021), 1–26.

[22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 16000–16009.

[23] Yuze He, Chen Bian, Jingfei Xia, Shuyao Shi, Zhenyu Yan, Qun Song, and Guoliang Xing. 2023. Vi-map: Infrastructure-assisted real-time hd mapping for autonomous driving. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*. 1–15.

[24] Sachini Herath, Hang Yan, and Yasutaka Furukawa. 2020. Ronin: Robust neural inertial navigation in the wild: Benchmark, evaluations, & new methods. In *2020 IEEE international conference on robotics and automation (ICRA)*. IEEE, 3146–3152.

[25] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J. Black, Otmar Hilliges, and Gerard Pons-Moll. 2018. Deep inertial poser: learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Trans. Graph.* 37, 6, Article 185 (Dec. 2018), 15 pages. doi:10.1145/3272127.3275108

[26] Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144* (2016).

[27] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, et al. 2018. Towards environment independent device free human activity recognition. In *Proceedings of the 24th annual international conference on mobile computing and networking*. 289–304.

[28] Antonio R Jimenez, Fernando Seco, Carlos Prieto, and Jorge Guevara. 2009. A comparison of pedestrian dead-reckoning algorithms using a low-cost MEMS IMU. In *2009 IEEE International Symposium on Intelligent Signal Processing*. IEEE, 37–42.

[29] Denizhan Kara, Tomoyoshi Kimura, Shengzhong Liu, Jinyang Li, Dongxin Liu, Tianshi Wang, Ruijie Wang, Yizhuo Chen, Yigong Hu, and Tarek Abdelzaher. 2024. FreqMAE: Frequency-Aware Masked Autoencoder for Multi-Modal IoT Sensing. In *Proceedings of the ACM on Web Conference 2024*. 2795–2806.

[30] Prerna Khanna, IV Ramakrishnan, Shubham Jain, Xiaojun Bi, and Aruna Balasubramanian. 2024. Hand Gesture Recognition for Blind Users by Tracking 3D Gesture Trajectory. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–15.

[31] Rushil Khurana, Karan Ahuja, Zac Yu, Jennifer Mankoff, Chris Harrison, and Mayank Goel. 2018. GymCam: Detecting, recognizing and tracking simultaneous exercises in unconstrained scenes. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 4 (2018), 1–17.

[32] A. I. King. 1984. A Technical Survey: A Review of Biomechanical Models. *Journal of Biomechanical Engineering* 106, 2 (1984), 97–104. doi:10.1115/1.3138480

[33] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[34] Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W Mahoney. 2021. Characterizing possible failure modes in physics-informed neural networks. *Advances in neural information processing systems* 34 (2021), 26548–26560.

[35] Hyeokhyen Kwon, Catherine Tong, Harish Haresamudram, Yan Gao, Gregory D Abowd, Nicholas D Lane, and Thomas Ploetz. 2020. Imutube: Automatic extraction of virtual on-body accelerometry from video for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 3 (2020), 1–29.

[36] Nicholas D Lane and Petko Georgiev. 2015. Can deep learning revolutionize mobile sensing?. In *Proceedings of the 16th international workshop on mobile computing systems and applications*. 117–122.

[37] Nicholas D Lane, Petko Georgiev, and Lorena Qendro. 2015. Deepear: robust smartphone audio sensing in unconstrained acoustic environments using deep learning. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*. 283–294.

[38] Hyung-Jik Lee and Seul Jung. 2009. Gyro sensor drift compensation by Kalman filter to control a mobile inverted pendulum robot system. In *2009 IEEE International Conference on Industrial Technology*. IEEE, 1–6.

[39] Yuyang Leng, Renyuan Liu, Hongpeng Guo, Songqing Chen, and Shuochao Yao. 2023. Scaleflow: Efficient deep vision pipeline with closed-loop scale-adaptive inference. In *Proceedings of the 31st ACM International Conference on Multimedia*. 1698–1706.

[40] Zikang Leng, Hyeokhyen Kwon, and Thomas Plötz. 2023. Generating virtual on-body accelerometer data from virtual textual descriptions for human activity recognition. In *Proceedings of the 2023 ACM International Symposium on Wearable Computers*. 39–43.

[41] Yuxuan Liang, Haomin Wen, Yuqi Nie, Yushan Jiang, Ming Jin, Dongjin Song, Shirui Pan, and Qingsong Wen. 2024. Foundation Models for Time Series Analysis: A Tutorial and Survey. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 6555–6565. doi:10.1145/3637528.3671451

[42] Renyuan Liu, Yuyang Leng, Kaiyan Liu, Shaohan Hu, Chun-Fu Chen, Peijun Zhao, Heechul Yun, and Shuochao Yao. 2025. DAF: An Efficient End-to-End Dynamic Activation Framework for on-Device DNN Training. In *Proceedings of the 23rd Annual International Conference on Mobile Systems, Applications and Services*. 196–208.

[43] Renyuan Liu, Yuyang Leng, Shilei Tian, Shaohan Hu, Chun-Fu Chen, and Shuochao Yao. 2024. Dynaspa: Exploiting spatial sparsity for efficient dynamic dnn inference on devices. In *Proceedings of the 22nd ACM Conference on Embedded Networked Sensor Systems*. 422–435.

[44] Renyuan Liu, Yuyang Leng, Shilei Tian, Shaohan Hu, Richard Chen, and Shuochao Yao. 2025. On-Device Dynamic DNN Inference through Spatial Sparsity Exploitation. *GetMobile: Mobile Computing and Communications* 29, 3 (2025), 35–38.

[45] Wenxin Liu, David Caruso, Eddy Ilg, Jing Dong, Anastasios I Mourikis, Kostas Daniilidis, Vijay Kumar, and Jakob Engel. 2020. Tlio: Tight learned inertial odometry. *IEEE Robotics and Automation Letters* 5, 4 (2020), 5653–5660.

[46] Yang Liu, Zhenjiang Li, Zhidan Liu, and Kaishun Wu. 2019. Real-time arm skeleton tracking and gesture inference tolerant to missing wearable sensors. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*. 287–299.

[47] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. 2023. SMPL: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 851–866.

[48] Wenjie Luo, Zhenyu Yan, Qun Song, and Rui Tan. 2021. PhyAug: Physics-directed data augmentation for deep sensing model transfer in cyber-physical systems. In *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (co-located with CPS-IoT Week 2021)*. 31–46.

[49] João Luís Marins, Xiaoping Yun, Eric R Bachmann, Robert B McGhee, and Michael J Zyda. 2001. An extended Kalman filter for quaternion-based orientation estimation using MARG sensors. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, Vol. 4. IEEE, 2003–2011.

[50] Hossein Mousavi Hondori and Maryam Khademi. 2014. A review on technical and clinical impact of microsoft kinect on physical therapy and rehabilitation. *Journal of medical engineering* 2014, 1 (2014), 846514.

[51] Jens Bo Nielsen. 2016. Human spinal motor control. *Annual Review of Neuroscience* 39, 1 (2016), 81–101.

[52] Nadhira Noor and In Kyu Park. 2023. A Lightweight Skeleton-Based 3D-CNN for Real-Time Fall Detection and Action Recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*. 2179–2188.

[53] Xiaomin Ouyang, Xian Shuai, Jiayu Zhou, Ivy Wang Shi, Zhiyuan Xie, Guoliang Xing, and Jianwei Huang. 2022. Cosmo: contrastive fusion learning with small data for multimodal human activity recognition. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 324–337.

[54] Xiaomin Ouyang, Zhiyuan Xie, Jiayu Zhou, Jianwei Huang, and Guoliang Xing. 2021. Clusterfl: a similarity-aware federated learning system for human activity recognition. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*. 54–66.

[55] Jose Luis Ponton, Haoran Yun, Andreas Aristidou, Carlos Andujar, and Nuria Pelechano. 2023. SparsePoser: Real-time full-body motion reconstruction from sparse data. *ACM Transactions on Graphics* 43, 1 (2023), 1–14.

[56] Hangwei Qian, Tian Tian, and Chunyan Miao. 2022. What makes good contrastive learning on small-scale wearable-based tasks?. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*. 3761–3771.

[57] Setareh Rahimi Taghanaki, Michael J Rainbow, and Ali Etemad. 2021. Self-supervised human activity recognition by learning to predict cross-dimensional motion. In *Proceedings of the 2021 ACM International Symposium on Wearable Computers*. 23–27.

[58] Bingbing Rao, Ehsan Kazemi, Yifan Ding, Devu M Shila, Frank M Tucker, and Liqiang Wang. 2022. Ctin: Robust contextual transformer network for inertial navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 5413–5421.

[59] Angelo M Sabatini. 2006. Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing. *IEEE transactions on Biomedical Engineering* 53, 7 (2006), 1346–1356.

[60] Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. 2021. E (n) equivariant graph neural networks. In *International conference on machine learning*. PMLR, 9323–9332.

[61] Sheng Shen, Mahanth Gowda, and Romit Roy Choudhury. 2018. Closing the gaps in inertial motion tracking. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 429–444.

[62] Sheng Shen, He Wang, and Romit Roy Choudhury. 2016. I am a smartwatch and i can track my user's arm. In *Proceedings of the 14th annual international conference on Mobile systems, applications, and services*. 85–96.

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

[64] Jochen Tautges, Arno Zinke, Björn Krüger, Jan Baumann, Andreas Weber, Thomas Helten, Meinard Müller, Hans-Peter Seidel, and Bernd Eberhardt. 2011. Motion reconstruction using sparse accelerometer data. *ACM Transactions on Graphics (ToG)* 30, 3 (2011), 1–12.

[65] Lam Tran, Thang Hoang, Alexandros Iosifidis, and Moncef Gabbouj. 2021. Multi-Model Long Short-Term Memory Network for Gait Recognition Using Window-Based Data Segment. *IEEE Access* 9 (2021), 23833–23846. doi:10.1109/ACCESS.2021.3057554

[66] Timo Von Marcard, Bodo Rosenhahn, Michael J Black, and Gerard Pons-Moll. 2017. Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. In *Computer graphics forum*, Vol. 36. Wiley Online Library, 349–360.

[67] Marvi Waheed, Hammad Afzal, and Khawir Mehmood. 2021. NT-FDS—A Noise Tolerant Fall Detection System Using Deep Learning on Wearable Devices. *Sensors* 21, 6 (2021), 2006. doi:10.3390/s21062006

[68] Jens Wittenburg. 2013. *Dynamics of Systems of Rigid Bodies*. Vol. 33. Springer-Verlag.

[69] Huatao Xu, Pengfei Zhou, Rui Tan, and Mo Li. 2023. Practically Adopting Human Activity Recognition. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*. 1–15.

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

[71] Shunpei Yamaguchi, Aditya Arun, Takuya Fujiwara, Misaki Sakuta, Ryotaro Hada, Takuya Fujihashi, Takashi Watanabe, Dinesh Bharadia, and Shunsuke Saruwatari. 2024. Experience: Practical Challenges for Indoor AR Applications *(ACM MobiCom '24)*. Association for Computing Machinery, New York, NY, USA, 1030–1044.

[72] Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition. arXiv:1801.07455 [cs.CV] https://arxiv.org/abs/1801.07455

[73] Jian Bo Yang, Minh Nhut Nguyen, Phyo Phyo San, Xiao Li Li, and Shonali Krishnaswamy. 2015. Deep convolutional neural networks on multichannel time series for human activity recognition. In *Proceedings of the 24th International Conference on Artificial Intelligence* (Buenos Aires, Argentina) *(IJCAI'15)*. AAAI Press, 3995–4001.

[74] Shuochao Yao, Shaohan Hu, Yiran Zhao, Aston Zhang, and Tarek Abdelzaher. 2017. Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In *Proceedings of the 26th international conference on world wide web*. 351–360.

[75] Shuochao Yao, Yiran Zhao, Huajie Shao, ShengZhong Liu, Dongxin Liu, Lu Su, and Tarek Abdelzaher. 2018. Fastdeepiot: Towards understanding and optimizing neural network execution time on mobile and embedded devices. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*. 278–291.

[76] Shuochao Yao, Yiran Zhao, Huajie Shao, Chao Zhang, Aston Zhang, Shaohan Hu, Dongxin Liu, Shengzhong Liu, Lu Su, and Tarek Abdelzaher. 2018. Sensegan: Enabling deep learning for internet of things with a semi-supervised framework. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 2, 3 (2018), 1–21.

[77] Shuochao Yao, Yiran Zhao, Aston Zhang, Lu Su, and Tarek Abdelzaher. 2017. Deepiot: Compressing deep neural network structures for sensing systems with a compressor-critic framework. In *Proceedings of the 15th ACM conference on embedded network sensor systems*. 1–14.

[78] Xinyu Yi, Yuxiao Zhou, Marc Habermann, Soshi Shimada, Vladislav Golyanik, Christian Theobalt, and Feng Xu. 2022. Physical inertial poser (pip): Physics-aware real-time human motion tracking from sparse inertial sensors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 13167–13178.

[79] Xinyu Yi, Yuxiao Zhou, and Feng Xu. 2021. TransPose: real-time 3D human translation and pose estimation with six inertial sensors. *ACM Trans. Graph.* 40, 4, Article 86 (July 2021), 13 pages. doi:10.1145/3450626.3459786

[80] Hyungjun Yoon, Hyeongheon Cha, Canh Hoang Nguyen, Taesik Gong, and Sung-Ju Lee. 2022. IMG2IMU: Applying Knowledge from Large-Scale Images to IMU Applications via Contrastive Learning. *arXiv preprint arXiv:2209.00945* (2022).

[81] Xiaoqun Yu, Seonghyeok Park, Doil Kim, Eungjin Kim, Jaewon Kim, Woosub Kim, Yechan An, and Shuping Xiong. 2023. A practical wearable fall detection system based on tiny convolutional neural networks. *Biomedical Signal Processing and Control* 86 (2023), 105325.

[82] Qingzhao Zhang, Xumiao Zhang, Ruiyang Zhu, Fan Bai, Mohammad Naserian, and Z Morley Mao. 2023. Robust real-time multi-vehicle collaboration on asynchronous sensors. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*. 1–15.

[83] Xiyuan Zhang, Xiaohan Fu, Diyan Teng, Chengyu Dong, Keerthivasan Vijayakumar, Jiayun Zhang, Ranak Roy Chowdhury, Junsheng Han, Dezhi Hong, Rashmi Kulkarni, et al. 2023. Physics-Informed Data Denoising for Real-Life Sensing Systems. In *Proceedings of the 21st ACM Conference on Embedded Networked Sensor Systems*. 83–96.

[84] Yu Zhang, Songpengcheng Xia, Lei Chu, Jiarui Yang, Qi Wu, and Ling Pei. 2024. Dynamic Inertial Poser (DynaIP): Part-Based Motion Dynamics Learning for Enhanced Human Pose Estimation with Sparse Inertial Sensors. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1889–1899. doi:10.1109/CVPR52733.2024.00185

[85] Chenyu Zhao, Ciyu Ruan, Jingao Xu, Haoyang Wang, Shengbo Wang, Jiaqi Li, Jirong Zha, Zheng Yang, Yunhao Liu, Xiao-Ping Zhang, and Xinlei Chen. 2024. Foes or Friends: Embracing Ground Effect for Edge Detection on Lightweight Drones. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '24)*. Association for Computing Machinery, New York, NY, USA, 1377–1392.

[86] Yuanqing Zheng, Guobin Shen, Liqun Li, Chunshui Zhao, Mo Li, and Feng Zhao. 2014. Travi-navi: Self-deployable indoor navigation system. In *Proceedings of the 20th annual international conference on Mobile computing and networking*. 471–482.

[87] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. 2021. iBOT: Image BERT Pre-Training with Online Tokenizer. *arXiv preprint arXiv:2111.07832* (2021).

[88] Pengfei Zhou, Mo Li, and Guobin Shen. 2014. Use it free: Instantly knowing your phone attitude. In *Proceedings of the 20th annual international conference on*

Mobile computing and networking. 605–616.

[89] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. 2019. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5745–5753.

[90] Qin Zou, Yanling Wang, Qian Wang, Yi Zhao, and Qingquan Li. 2020. Deep Learning-Based Gait Recognition Using Smartphones in the Wild. *IEEE Transactions on Information Forensics and Security* 15 (2020), 3197–3212. doi:10.1109/TIFS.2020.2985628