PrimeIMU: High-Frequency Video-to-IMU Synthesis via Physics-Guided Simulation and Hybrid U-Net Refinement

ANONYMOUS AUTHOR(S)

Inertial Measurement Units (IMUs) are fundamental sensing components across robotics, drones, and other embodied systems, providing high-frequency inertial signals essential for control, stabilization, state estimation, and dynamics modeling. However, acquiring large-scale, high-sampling-rate IMU datasets for diverse articulated platforms remains challenging: deploying, calibrating, and synchronizing physical IMU sensors is costly, time-consuming, and often constrained by hardware limitations and operational requirements. To alleviate these challenges, recent work has explored synthesizing IMU signals from either model-synthesized motions or vision-derived kinematic estimates. While motion-synthesis approaches scale well, they lack real-world visual grounding. Vision-driven approaches offer such grounding but are fundamentally constrained by the limited temporal resolution of visual inputs and the inherent mismatch between kinematic estimates and inertial measurements.

Building on these observations, we propose PrimeIMU, a framework that generates IMU signals from vision-derived kinematic trajectories and physics-simulated IMU signals, using a hybrid U-Net that learns to map simulated signals to real sensor measurements. Extensive experiments show that PrimeIMU (1) improves the quality of inertial signal synthesis, (2) generalizes to unseen motion patterns while preserving realistic signal characteristics, (3) enables synthetic-only training of downstream models with performance close to real-sensor training and yields further gains when used to augment real data, and (4) adapts across datasets and sensor configurations with minimal fine-tuning to support downstream tasks under domain shift.

CCS Concepts: • **Computing methodologies** → **Machine learning**.

Additional Key Words and Phrases: IMU Signal Synthesis

ACM Reference Format:

Anonymous Author(s). 2018. PrimeIMU: High-Frequency Video-to-IMU Synthesis via Physics-Guided Simulation and Hybrid U-Net Refinement. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 23 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Inertial Measurement Unit (IMU) data are widely used across articulated mechanical and robotic systems [18, 19, 25, 34], including aerial drones [7, 26] and other embodied platforms [1, 8]. Their high-frequency inertial measurements capture fine-grained motion and dynamic behavior [19, 34], supporting tasks such as motion analysis, dynamic behavior characterization [25], operational assessment [38], and model-based simulation of articulated structures [24]. Achieving such applications at scale requires large, high-frequency IMU datasets with reliable annotations. However, collecting such data remains highly challenging, as the process is time-consuming, costly, and demands substantial domain expertise [5, 11, 15, 23, 27, 36, 39].

To address the scarcity of labeled IMU data, existing research has explored synthetic IMU generation for data augmentation through two primary approaches. The first leverages *large language models* (LLMs) to generate diverse

Manuscript submitted to ACM

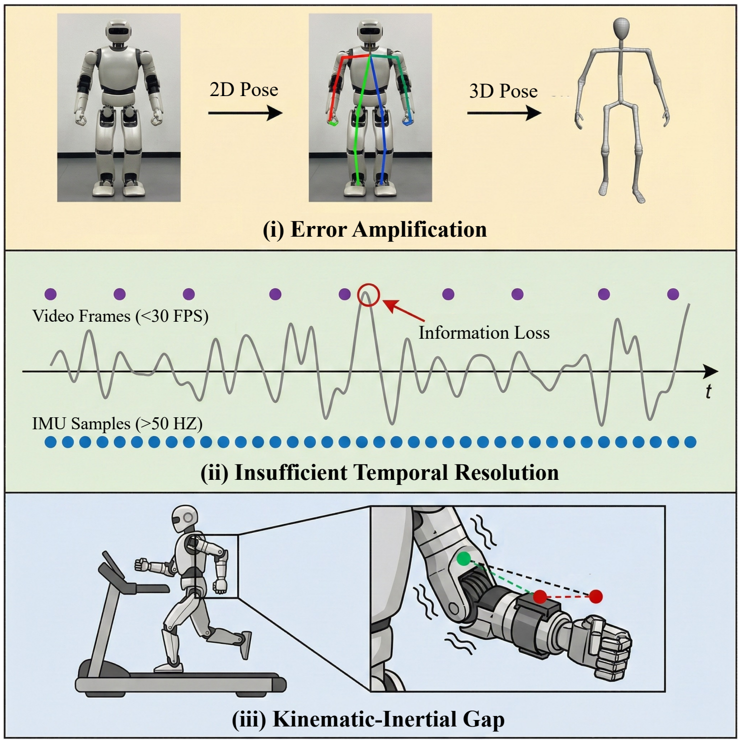

Fig. 1. Illustration of three fundamental issues in current vision-driven IMU synthesis pipelines: (i) error amplification introduced by kinematic estimation, (ii) insufficient temporal resolution when projecting low-frame-rate video into high-frequency IMU signals, and (iii) discrepancies between idealized kinematic models and the motion sensed by an IMU mounted on a moving structure.

motion descriptions, which are converted into 3D articulated trajectories via motion synthesis and then transformed into virtual IMU signals through motion-to-IMU modeling [15, 16, 41]. The second leverages *vision-driven models* to extract 2D kinematic cues from visual data, lift them to 3D articulated representations, and synthesize virtual IMU signals via kinematic modeling [13, 14, 29]. While LLM-based approaches offer scalability, they lack visual grounding and may produce IMU signals with unrealistic dynamics. Vision-driven methods provide visual grounding but still fall short of real IMU quality for three key reasons (Figure 1): (i) errors in visual kinematic estimation propagate and amplify during differentiation, producing noisy or spiky inertial signals; (ii) visual frame rates (typically 15–30 FPS) remain far below IMU sampling rates (> 50 Hz), leading to temporal under-resolution that cannot be recovered through interpolation; and (iii) a kinematic–inertial gap persists between idealized articulated motion and the inertial response of physical IMU hardware, whose measurements reflect mounting dynamics and other non-idealities. These limitations reduce the realism of synthesized signals and limit the utility of synthetic data.

Addressing these challenges would provide significant value for both research and practical deployment, as it would enable the development of IMU-driven models with little or no reliance on large amounts of real sensor data. Consequently, one crucial question has arisen: *Is it feasible to synthesize IMU signals that faithfully capture physical sensor behavior while supporting IMU-based downstream applications?*

IMU Generation Method	Realism-Oriented Designs				
	real.	vis.+phys.	temp.res.	dev.adpt.	kin.t.freq.
IMUGPT [16]	✗	✗	✗	✗	✗
IMUGPT 2.0 [15]	✗	✗	✗	✗	✗
UniMTS [41]	✗	✗	✗	✗	✗
Video2IMU [14]	✗	✗	✗	✗	✗
Vi2IMU [29]	✗	✓	✗	✗	✗
IMUTube [13]	✗	✓	✗	✗	✗
PrimeIMU	✓	✓	✓	✓	✓

Table 1. Comparison of PrimeIMU with State-of-the-Art IMU generation methods on realism-oriented designs, where *real.*: explicitly aims for realistic IMU generation (beyond pure augmentation); *vis.+phys.*: combines visual grounding with physical modeling; *temp.res.*: addresses limited temporal resolution from video frame rates; *dev.adpt.*: adapts synthesis to device-specific characteristics (e.g., FPS, IMU sampling rates); *kin.t.freq.*: incorporates kinematic, temporal, and frequency-domain information during synthesis.

To overcome these challenges, we propose PrimeIMU, a synthesis framework that incorporates key design components summarized in Table 1. PrimeIMU synthesizes high-fidelity, high-frequency IMU signals by using physics-based IMU simulation on video-derived kinematics and refining the simulated signals with a hybrid U-Net to obtain high sensor-fidelity IMU signals. Our contributions are:

- **Realistic, Sensor-Faithful Video-to-IMU Synthesis.** We present PrimeIMU, a video-driven IMU synthesis pipeline focused on generating realistic and sensor-faithful inertial signals. The framework integrates physics-based IMU simulation with a hybrid U-Net refinement module, jointly addressing three key limitations of prior work: noise amplification in pose-derived kinematics, temporal under-resolution in video inputs, and the mismatch between kinematic modeling and the behavior of physical IMU sensors.
- **Generalization to Unseen Motion Patterns.** PrimeIMU generalizes to motion patterns not observed during training, indicating that the method captures inertial structure that extends beyond the training distribution.
- **Synthetic-Only Training of IMU-Based Models.** Downstream models trained solely on PrimeIMU-generated IMU signals achieve performance close to that of models trained on real sensor data, demonstrating that high-fidelity synthetic IMUs can effectively substitute for real measurements when sensor collection is limited.
- **Augmentation Benefits with Real IMU Data.** PrimeIMU-generated signals introduce sensor-faithful patterns that complement real sensor measurements, leading to measurable gains in downstream tasks when used as augmentation.
- **Cross-Dataset and Cross-Device Adaptation.** PrimeIMU accommodates domain shifts arising from different datasets and IMU hardware, requiring minimal fine-tuning to adapt to new sensor and sampling configurations.

2 Related Work

2.1 Data Scarcity and Learning Paradigms

Although IMUs are widely deployed across articulated mechanical and robotic systems, the availability of large, diverse, and well calibrated inertial datasets remains limited. Constructing datasets that capture a broad range of motion patterns, operating conditions, and device configurations requires controlled hardware deployment, precise calibration, and repeated data collection. These requirements impose substantial effort and often result in datasets that are restricted in scope and lack the diversity needed for robust inertial modeling. Variations in device characteristics and mounting conditions further introduce distribution shifts that make cross system generalization challenging [10, 17]. Existing collections therefore remain narrow in operational coverage despite the ubiquity of MEMS IMUs.

Annotation quality further affects the usability of real world IMU datasets. Establishing reliable ground truth typically requires structured data collection protocols or manual processing of continuous recordings, which can lead to inconsistencies in activity boundaries and labeling conventions [5, 11, 15, 23, 27, 36, 39]. Such variation limits the scale and uniformity of supervised training corpora.

To mitigate dataset limitations, prior work explores self supervised pretraining [12, 28], few shot and domain adaptation [3, 9], metric based learning [6], adversarial robustness approaches [2], and cross dataset transfer [30]. While these strategies improve model generalization, they remain ultimately bounded by the diversity and coverage of real IMU data. This motivates the development of realistic synthetic IMU generation, where synthesized signals aim to reproduce device level inertial behavior and scale across motion types and hardware configurations. Such synthetic data can complement limited real measurements and support downstream inertial modeling when large scale physical data collection is impractical.

2.2 Cross Modality IMU Synthesis

Due to the difficulty of collecting large annotated IMU datasets, recent work has increasingly explored the synthesis of inertial signals from other modalities. Existing approaches generally fall into text driven and video driven categories. Text driven pipelines such as IMUGPT [16] and IMUGPT 2.0 [15] use large language models to generate motion descriptions that are mapped to parametric trajectories and subsequently converted into synthetic IMU sequences. UniMTS [41] enriches the process with linguistic priors before translating them into skeletal motions. These approaches scale easily but lack visual grounding and often deviate from the statistics of real sensor measurements.

Video driven pipelines incorporate visual grounding by extracting motion cues directly from videos. IMUTube [13] estimates 2D poses, lifts them to 3D, and differentiates the trajectories to approximate inertial signals. Video2IMU [14] regresses IMU sequences directly from 2D poses, and Vi2IMU [29] targets wrist mounted sensors in structured communication settings. These methods typically produce more realistic inertial signatures than text based pipelines.

Despite these advantages, video driven IMU synthesis still faces fundamental limitations. Error in 2D and 3D pose estimation is amplified during differentiation, leading to unstable inertial predictions. The mismatch between video frame rates and IMU sampling rates limits the synthesis of high frequency motion. Moreover, visually derived kinematic trajectories do not fully capture the inertial dynamics measured by physical IMU sensors, which depend on forces and motions not represented in purely kinematic models. These challenges indicate that effective IMU synthesis requires jointly leveraging visual kinematics and physics-based inertial modeling to produce signals that better approximate the behavior of real IMU sensors.

3 Method

Our objective is to develop a framework that synthesizes high-frequency, physically plausible Inertial Measurement Unit (IMU) data from standard RGB video inputs. The core challenge lies in bridging the significant domain gap between low-frequency visual data and high-frequency inertial signals, while also addressing the noise and inaccuracies inherent in video-based motion capture. To this end, we propose a novel two-stage method, PrimeIMU, as depicted in Figure 2. The first stage uses vision-based kinematic estimation and physics-based modeling to produce an initial IMU signal that captures coarse motion trends but remains noisy and low in temporal resolution. The second stage applies a generative

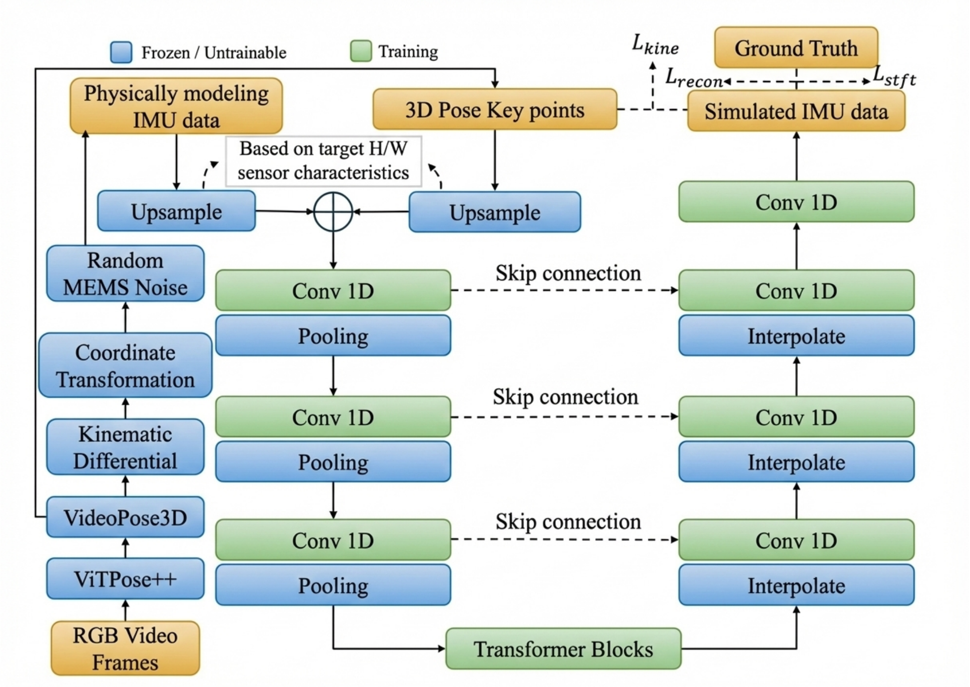

Fig. 2. Overview of the PrimeIMU pipeline for synthesizing high-frequency IMU signals from RGB videos. (**Stage 1: Physics-based Simulation**) Each RGB frame is processed by ViTPose++ to estimate 2D keypoints, which are then lifted to 3D joint trajectories using VideoPose3D. The 3D poses are converted into low-frequency simulated IMU signals through kinematic differentiation, coordinate transformation, and MEMS noise modeling. (**Stage 2: Deep Generative Refinement**) A Hybrid U-Net with a transformer bottleneck takes the upsampled simulated IMU together with the upsampled pose sequence, refines them via convolutional–transformer blocks with skip connections, and generates realistic high-frequency IMU signals. Training is guided by a composite loss that combines time-domain reconstruction, frequency-domain STFT alignment, and kinematic consistency to enforce both fidelity and physical plausibility.

refinement Hybrid Unet module that denoises the signal, reconstructs high-frequency dynamics, and enforces kinematic consistency at the target sampling rate.

3.1 Stage 1: Physics-Guided IMU Simulation

The first stage of our framework converts the input video into a coarse IMU estimate by extracting 3D kinematic trajectories and applying a physics-guided inertial simulation model, producing a low-frequency approximation of the target sensor signals.

3.1.1 3D Pose Estimation. Given an RGB video sequence, our first step is to reconstruct the 3D trajectory of key object joints. We adopt a robust two-step lifting approach. First, we process each video frame with ViTPose++ [40], a powerful vision transformer-based model, to obtain accurate 2D keypoint coordinates for each object in the scene. Subsequently, the resulting 2D pose sequence is fed into VideoPose3D [22], a temporal convolutional network that leverages motion context to lift the 2D coordinates into a coherent 3D skeleton sequence. The output of this step is a sequence of 3D joint positions $P_{3D} \in \mathbb{R}^{T_v \times K \times 3}$, where T_v is the number of video frames and K is the number of keypoints.

3.1.2 Physics-driven Kinematic Modeling. With the 3D joint trajectories, we simulate the readings of a 6-axis IMU (3-axis accelerometer, 3-axis gyroscope) attached to a specific object segment. This process involves three key modules.

Kinematic Differentiation. This module computes the linear acceleration and angular velocity from the pose data. Let the 3D position of the distal joint of a segment (e.g., the wrist) at time t be denoted by the vector $\mathbf{p}(t)$. The global linear velocity $\mathbf{v}(t)$ and acceleration $\mathbf{a}_{\text{global}}(t)$ are obtained by taking the first and second time derivatives of the position, respectively:

$$\mathbf{v}(t) = \frac{d\mathbf{p}(t)}{dt}, \quad \mathbf{a}_{\text{global}}(t) = \frac{d^2\mathbf{p}(t)}{dt^2}. \tag{1}$$

Directly differentiating discrete, noisy position data drastically amplifies noise. Therefore, we first apply a Savitzky-Golay filter [31] to the position sequence $\mathbf{p}(t)$ to obtain a smoothed trajectory before differentiation, which is implemented using finite differences.

The orientation of the object segment is more robustly defined by the vector connecting the proximal joint $\mathbf{p}_{\text{prox}}(t)$ to the distal joint $\mathbf{p}_{\text{dist}}(t)$. We use this vector to construct a time-varying local coordinate frame for the segment, represented by a sequence of orientation quaternions $q(t)$. The local angular velocity $\boldsymbol{\omega}_{\text{local}}(t)$ is then derived from the temporal evolution of these quaternions. Given the relationship $\dot{q}(t) = \frac{1}{2}\begin{bmatrix} 0 \\ \boldsymbol{\omega}_{\text{local}}(t) \end{bmatrix} \otimes q(t)$, where $\otimes$ denotes quaternion multiplication, the angular velocity can be computed as:

$$\begin{bmatrix} 0 \\ \boldsymbol{\omega}_{\text{local}}(t) \end{bmatrix} = 2 \cdot \dot{q}(t) \otimes q(t)^{-1}, \tag{2}$$

where $q(t)^{-1}$ is the quaternion conjugate and $\dot{q}(t)$ is computed via finite differences.

Coordinate System Transformation. An IMU measures acceleration and angular velocity in its own local coordinate frame. The angular velocity $\boldsymbol{\omega}_{\text{local}}(t)$ is already in this frame. However, the linear acceleration $\mathbf{a}_{\text{global}}(t)$ must be transformed. Using the orientation quaternion $q(t)$, we rotate the global acceleration vector into the local frame and subtract the effect of gravity, $\mathbf{g} = [0, 0, -9.81]^T \text{m/s}^2$, which is also transformed into the local frame:

$$\mathbf{a}_{\text{local}}(t) = R(q(t))^{-1}(\mathbf{a}_{\text{global}}(t)) - R(q(t))^{-1}\mathbf{g}, \tag{3}$$

where $R(q(t))$ is the rotation matrix corresponding to quaternion $q(t)$. This step is crucial for simulating realistic accelerometer readings that include gravitational components.

Random MEMS Noise Simulation. Real IMU sensors are subject to various sources of noise. To enhance the realism of our simulated data, we model two primary noise types: a Gaussian white noise component and a random walk bias. The final simulated IMU signal $I_{\text{sim}}(t) = [\mathbf{a}_{\text{sim}}(t), \boldsymbol{\omega}_{\text{sim}}(t)]$ is generated as:

$$\mathbf{a}_{\text{sim}}(t) = \mathbf{a}_{\text{local}}(t) + \mathbf{n}_a(t) + \mathbf{b}_a(t), \tag{4}$$

$$\boldsymbol{\omega}_{\text{sim}}(t) = \boldsymbol{\omega}_{\text{local}}(t) + \mathbf{n}_\omega(t) + \mathbf{b}_\omega(t), \tag{5}$$

where $\mathbf{n}(t)$ is sampled from a zero-mean Gaussian distribution whose standard deviation depends on the sensor's noise density, and $\mathbf{b}(t)$ is a bias term that evolves as a random walk. The output of this stage is a low-frequency simulated IMU signal, $I_{\text{sim}} \in \mathbb{R}^{T_v \times 6}$.

3.2 Stage 2: Hybrid U-Net Generative Refinement

The physically simulated data I_{sim} is noisy and shares the low frequency of the source video. The second stage of our framework uses a generative network to learn the mapping from this low-quality simulation to high-quality, high-frequency IMU data, conditioned on the low-frequency pose.

3.2.1 Network Architecture. We design a Hybrid U-Net architecture that combines the local feature extraction power of convolutions with the global context modeling of transformers. The network takes a low-frequency pose sequence $P_{\text{low}} \in \mathbb{R}^{T_l \times D_{\text{pose}}}$ and a random noise vector $N \in \mathbb{R}^{T_h \times D_{\text{imu}}}$ as input, where T_l and T_h are the lengths of the low- and high-frequency sequences, respectively.

The pose sequence P_{low} is first upsampled to the target length T_h via linear interpolation and then projected into a high-dimensional embedding space. This pose embedding is added to an embedding of the input noise N. The resulting sequence $X \in \mathbb{R}^{T_h \times D_{\text{model}}}$ is the input to the U-Net.

Encoder. The encoder consists of a series of 1D convolutional blocks. Each block contains convolutional layers with residual connections, followed by a max-pooling layer that halves the temporal resolution. The outputs of each block before pooling are saved as skip connections.

Bottleneck. At the lowest temporal resolution, the feature map is passed to a Transformer block. This allows the model to capture long-range temporal dependencies and global patterns in the motion sequence. Positional encodings are added to the features before the Transformer block to provide temporal context.

Decoder. The decoder mirrors the encoder's structure. At each stage, the feature map is upsampled using linear interpolation. The upsampled features are concatenated with the corresponding skip connection from the encoder path and passed through a 1D convolutional block. This process progressively refines the features while restoring the original temporal resolution. A final 1D convolutional layer projects the output features back to the dimension of the IMU data, yielding the generated high-frequency signal $I_{\text{gen}} \in \mathbb{R}^{T_h \times 6}$.

3.3 Composite Loss Design

The network is trained end-to-end by minimizing a composite loss function that evaluates the generated signal in both the time and frequency domains, while also enforcing physical plausibility. The ground truth is a sequence of real, high-frequency IMU data $I_{\text{gt}} \in \mathbb{R}^{T_h \times 6}$.

Reconstruction Loss ($\mathcal{L}_{recon}$). To ensure the generated signal matches the ground truth in the time domain, we use a standard L1 loss:

$$\mathcal{L}_{\text{recon}} = \|I_{\text{gen}} - I_{\text{gt}}\|_1. \tag{6}$$

This loss penalizes deviations in the signal's amplitude and temporal structure.

STFT Loss ($\mathcal{L}_{stft}$). Matching the frequency content is crucial for realistic IMU signals. We employ a multi-resolution Short-Time Fourier Transform (STFT) loss. This loss is the sum of two components: a spectral convergence term and a log STFT magnitude term, computed over multiple FFT resolutions:

$$\mathcal{L}_{\text{stft}} = \sum_{r \in R} \left(\frac{\| \, |S_r(I_{\text{gen}})| - |S_r(I_{\text{gt}})| \, \|_F}{\| \, |S_r(I_{\text{gt}})| \, \|_F} + \| \log |S_r(I_{\text{gen}})| - \log |S_r(I_{\text{gt}})| \|_1 \right), \tag{7}$$

where $S_r(\cdot)$ denotes the STFT operator at resolution r, $|\cdot|$ is the magnitude, and $\|\cdot\|_F$ is the Frobenius norm. This loss ensures that the spectral characteristics of the generated signal align with the ground truth.

Kinematic Consistency Loss ($\mathcal{L}_{kine}$). This novel loss term enforces that the generated IMU data is physically consistent with the input motion. We take the acceleration component of the generated high-frequency IMU signal, $I_{\text{gen}}^{\text{accel}} \in \mathbb{R}^{T_h \times 3}$, and apply a differentiable double integrator to recover the corresponding high-frequency 3D pose trajectory, $\hat{P}_{\text{high}}$. An initial position from the ground-truth low-frequency pose is used as the integration constant. This predicted trajectory is then downsampled via interpolation to the original low frequency, yielding $\hat{P}_{\text{low}}$. The kinematic loss is the Mean Squared Error (MSE) between this predicted low-frequency pose and the ground-truth low-frequency pose P_{low} that was an input to the network:

$$\mathcal{L}_{\text{kine}} = \|\text{Downsample}(\hat{P}_{\text{high}}) - P_{\text{low}}\|_2^2. \tag{8}$$

This loss acts as a powerful regularizer, ensuring the network generates signals that adhere to the laws of motion.

Total Loss. ($\mathcal{L}_{total}$). The final training objective is a weighted sum of the three loss components:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{recon}}\mathcal{L}_{\text{recon}} + \lambda_{\text{stft}}\mathcal{L}_{\text{stft}} + \lambda_{\text{kine}}\mathcal{L}_{\text{kine}}, \tag{9}$$

where λ_{recon}, λ_{stft}, and λ_{kine} are hyper-parameters that balance the contribution of each term.

4 Experiments

4.1 Setup

4.1.1 Datasets. We conduct experiments on two publicly available multimodal benchmarks that provide both video and inertial sensor modalities for activity recognition: *(i)* **UTD-MHAD** [4] contains synchronized RGB videos, depth videos, skeleton joint positions, and 6-axis inertial signals (accelerometer and gyroscope) recorded from a Kinect camera and a single wearable inertial sensor. Eight subjects performed 27 different actions, including sports (e.g., basketball shoot, tennis swing), hand gestures (e.g., wave, clap, draw shapes), daily activities (e.g., sit-to-stand, knock on door), and exercises (e.g., squat, lunge, arm curl). Each subject repeated every action four times, yielding a total of 861 sequences. The inertial sensor was placed on the right wrist or thigh depending on the action, and all modalities were temporally synchronized and manually segmented, making UTD-MHAD a compact yet diverse benchmark for multimodal activity recognition. *(ii)* **MM-Fit** [33] contains synchronized multi-view RGB-D videos, skeleton poses, and wearable inertial signals (accelerometer, gyroscope, and magnetometer). Ten subjects performed ten types of resistance exercises, including squats, lunges, push-ups, sit-ups, curls, rows, presses, raises, and jumping jacks. Each exercise consisted of three sets of ten repetitions, resulting in 21 sessions, 616 sets, and 6160 annotated repetitions. All modalities were time-synchronized using a calibration jump and annotated with exercise types, set boundaries, and repetition counts, providing a challenging and diverse benchmark for multimodal activity recognition in workout scenarios.

4.1.2 Baseline. Because no prior state-of-the-art video-to-IMU methods publicly release their code, we construct a reproducible baseline by following the methodological descriptions provided in recent work, in particular, IMUTube [13] and UniMTS [41]. IMUTube is specifically designed for video-to-IMU synthesis, and we therefore re-implement its pipeline as faithfully as the available details permit. Although UniMTS is not a video-to-IMU approach, several of its design components for multimodal motion-to-sensor generation inform aspects of our IMUTube re-implementation. Unless otherwise stated, all references to the *Baseline* refer to this reconstructed IMUTube pipeline.

4.1.3 IMU Generation Quality Evaluation Metrics. We assess the fidelity of generated IMU signals by comparing them to ground-truth recordings using both error- and correlation-based measures. Error-based metrics (RMSE/MAE) quantify magnitude discrepancies but may overlook trend misalignment, while correlation-based metrics (R^2 and Pearson) ensure temporal and physical plausibility: *(i)* **Root Mean Square Error (RMSE)** emphasizes large deviations between generated and real signals, making it sensitive to spikes; small values indicate stable predictions, while large values suggest spiky or unstable outputs. *(ii)* **Mean Absolute Error (MAE)** captures the average absolute deviation from ground truth; lower values reflect better overall alignment, while higher values indicate consistent drift. *(iii)* **Coefficient of Determination (R^2)** measures the proportion of variance in the real signal explained by the generated one; values close to 1 indicate faithful reproduction of dynamics, values near 0 imply mean-level prediction, and negative values denote worse-than-mean performance. *(iv)* **Pearson Correlation** quantifies linear trend alignment between generated and ground-truth signals; values near +1 denote strong positive alignment, values near 0 indicate no relation, and values near -1 imply reversed, physically implausible trends.

4.1.4 Classification Performance Evaluation Metrics. We adopt two standard metrics for action recognition evaluation: *(i)* **Accuracy (Acc)** is the overall Top-1 classification accuracy, i.e., the fraction of correctly predicted activity labels across all samples. *(ii)* **Macro F1 (F1)** is the unweighted average of per-class F1 scores, giving equal importance to each class and thus providing robustness to class imbalance.

4.1.5 Classification Models. We evaluate the effectiveness of synthetic IMU data on five representative models spanning both classical machine learning and deep learning paradigms: *(i)* **Random Forest** [21] is an ensemble of decision trees that serves as a strong non-deep-learning baseline for IMU classification. *(ii)* **Support Vector Machine (SVM)** [35] is a kernel-based margin classifier that provides a competitive traditional baseline for time-series data. *(iii)* **DeepConvLSTM** [20] integrates convolutional layers for local feature extraction with LSTMs for temporal sequence modeling and is widely used in action recognition tasks. *(iv)* **DeepConvLSTM_Attention** [32] extends DeepConvLSTM by adding a self-attention mechanism to capture long-range temporal dependencies through adaptive weighting of time steps. *(v)* **Transformer** [37] is a fully attention-based architecture that models long-range dependencies without recurrence or convolutions and has recently gained popularity for sequential action recognition tasks. Together, these models form a diverse evaluation suite for assessing the utility of synthetic IMU data across classical and deep learning action recognition classifiers.

4.1.6 Configuration. For the IMU generation stage, the sampling frequency of the IMU data from the physics-based simulation is set to match the frame rate of the source video. Our U-Net model is trained for 100 epochs with a learning rate of 1×10^{-3} for both datasets. To evaluate the quality of our synthetic data, we use activity recognition as a downstream task to measure the performance gap between models trained on our generated data versus real IMU data. All neural classifiers in this downstream setting are trained for 50 epochs with a learning rate of 1×10^{-3}.

4.2 Comparison with the Baseline Method on IMU Generation Quality

We compare PrimeIMU with a kinematics-only baseline on two multimodal benchmarks, UTD-MHAD and MM-Fit, using four complementary criteria (MAE, RMSE, R^2, and Pearson). Table 2 summarizes results per axis for accelerometer and gyroscope signals.

Overall trends. Across both datasets, PrimeIMU produces signals that align much more closely with real sensors in terms of variance explained and temporal trend agreement. On UTD-MHAD, our method achieves very high R^2

Table 2. Comparison of our PrimeIMU with the re-implemented IMUTube Baseline on the UTD-MHAD and MM-Fit datasets. We report per-axis results for accelerometer (X/Y/Z) and gyroscope (X/Y/Z) signals using four complementary metrics: mean absolute error (MAE↓), root mean square error (RMSE↓), coefficient of determination (R^2 ↑), and Pearson correlation (↑). Across both datasets, PrimeIMU consistently yields lower errors and substantially higher correlations, indicating stronger fidelity to real sensor measurements compared to the baseline.

Dataset	Method	Metric	Accel X	Accel Y	Accel Z	Gyro X	Gyro Y	Gyro Z
UTD-MHAD	Baseline	MAE ↓	0.8395	7.6634	4.6674	1.3719	1.4953	2.1858
		RMSE ↓	1.0821	8.2838	5.3449	1.9902	2.2392	2.9975
		R^2 ↑	-670.6101	-65171.0273	-66641.8406	-1.1429	-1.3515	-2.9680
		Pearson ↑	-0.1797	0.1374	-0.1966	0.0143	0.0507	-0.1329
	PrimeIMU (Ours)	MAE ↓	**0.0306**	**0.0176**	**0.0189**	**3.0549**	**3.1581**	**2.9964**
		RMSE ↓	**0.0399**	**0.0230**	**0.0262**	**4.2513**	**4.0785**	**3.8244**
		R^2 ↑	**0.8706**	**0.9785**	**0.9583**	**0.9842**	**0.8805**	**0.8852**
		Pearson ↑	**0.9886**	**0.9977**	**0.9916**	**0.9972**	**0.9957**	**0.9983**
MM-Fit	Baseline	MAE ↓	2.1401	8.0887	6.4013	0.6339	0.7826	1.4799
		RMSE ↓	3.1713	9.7756	8.3973	0.9037	1.1181	2.2551
		R^2 ↑	-1754265773	-1540101166	-7826811739	-3045290257	-17280099211	-416054159758
		Pearson ↑	-0.0018	0.0016	-0.0032	-0.005	0.0005	-0.0047
	PrimeIMU (Ours)	MAE ↓	**0.2808**	**0.5844**	**0.2074**	**0.0691**	**0.0341**	**0.0495**
		RMSE ↓	**0.3656**	**0.6386**	**0.2764**	**0.0890**	**0.0442**	**0.0655**
		R^2 ↑	**-0.1784**	**-6.4791**	**0.3422**	**-0.1988**	**0.5433**	**-1.8088**
		Pearson ↑	**0.8963**	**0.9275**	**0.9333**	**0.9455**	**0.9526**	**0.9200**

(0.87–0.98) and Pearson correlations (0.99 level) for all six axes, whereas the baseline yields negative R^2 and weak or even negative correlations. On MM-Fit—a more diverse, device-heterogeneous setting—the baseline effectively collapses (near-zero Pearson and extremely negative R^2), while PrimeIMU consistently delivers strong correlations (0.90–0.95) and substantially lower errors.

Accelerometer fidelity. UTD-MHAD shows the clearest gap in accelerometry: PrimeIMU reduces MAE from 0.84–7.66 (baseline) to 0.018–0.031, and RMSE from 1.08–8.28 to 0.023–0.040, alongside near-perfect Pearson ($\geq$ 0.99). The baseline's negative R^2 indicates it fails to model even coarse amplitude/variance, while our hybrid refinement closely tracks both magnitude and dynamics. On MM-Fit, the baseline again shows almost no correlation to real signals; PrimeIMU improves Pearson to 0.90–0.93 and reduces MAE/RMSE by large margins on all axes (e.g., Accel-Z MAE 6.40 ⟶ 0.21).

Gyroscope behavior and scale. For UTD-MHAD gyroscope, PrimeIMU attains excellent R^2 (0.88–0.98) and Pearson (0.996–0.998), indicating highly faithful temporal patterns and variance modeling. Interestingly, its absolute error (MAE/RMSE) is sometimes larger than the baseline on certain axes (e.g., Gyro X/Y/Z MAE $\sim$ 3.0–3.16 vs. baseline 1.37–2.19). Given the simultaneously high R^2/Pearson, this discrepancy points to a residual *scale or bias* mismatch rather than dynamics errors. Such effects are consistent with device- or placement-specific angular rate scaling and soft-tissue artifacts; they are typically addressable with a lightweight post-hoc calibration (e.g., per-axis affine re-scaling) without altering downstream recognition performance. On MM-Fit, PrimeIMU clearly dominates the baseline across all gyro axes in both error (e.g., Gyro X RMSE 0.90 ⟶ 0.089) and correlation (Pearson $\approx$ 0.95), reflecting stronger device/domain robustness.

Stability across datasets. The contrast between datasets highlights the intended design of PrimeIMU: when the baseline is exposed to domain shifts (camera FPS, subject style, sensor sampling, placement), its signals lose both spectral structure and temporal alignment (strongly negative R^2, Pearson ≈ 0 on MM-Fit). By fusing low-frequency kinematics with learned high-frequency priors and training under joint time–frequency–kinematic objectives, PrimeIMU preserves activity-relevant dynamics and maintains high correlations under shift.

Takeaways. (1) Temporal fidelity: near-ceiling Pearson/R^2 on UTD-MHAD and strong correlations on MM-Fit indicate that PrimeIMU captures the correct motion dynamics where the baseline fails. (2) Amplitude alignment: large MAE/RMSE reductions for accelerometers on both datasets show improved magnitude realism; remaining gyro scale offsets on UTD-MHAD are minor and correctable. (3) Domain robustness: the gains are most pronounced under cross-device variability (MM-Fit), where the baseline degenerates but PrimeIMU remains reliable.

Overall, the results corroborate our design goals: by combining kinematic guidance with learned frequency-aware refinement, PrimeIMU generates signals that are markedly more realistic and sensor-faithful than those from a physics-only baseline. The improvements are evident across both accelerometer and gyroscope axes, where variance explained and temporal correlations reach near-ceiling levels on UTD-MHAD and remain strong even under the cross-device variability of MM-Fit. These findings confirm that integrating physics priors with data-driven refinement not only enhances realism but also preserves robustness under domain shift.

4.3 Classification Performance using Generated IMU Data: Baseline vs. Ours

Table 3. Classification results on the UTD-MHAD dataset comparing the baseline generator and PrimeIMU across four training–testing settings: GT→GT (train/test on real IMU), GEN→GEN (train/test on generated IMU), GT→GEN (train on real IMU, test on generated IMU), and GEN→GT (train on generated IMU, test on real IMU). Reported metrics are Accuracy (Acc) and macro-F1; ↑ indicates higher is better.

Model	**Train**	**Test**	**Baseline**		**PrimeIMU (Ours)**	
			Acc ↑	F1 ↑	Acc ↑	F1 ↑
Random Forest	GT	GT	0.8023	0.7962	0.8023	0.7962
	GEN	GEN	0.6734	0.6737	**0.8166**	**0.8044**
	GT	GEN	0.0201	0.0041	**0.7880**	**0.7813**
	GEN	GT	0.0516	0.0223	**0.7937**	**0.7847**
SVM	GT	GT	0.7106	0.6908	0.7106	0.6908
	GEN	GEN	0.4785	0.4711	**0.7106**	**0.6882**
	GT	GEN	0.0602	0.0042	**0.7221**	**0.7052**
	GEN	GT	0.0287	0.0021	**0.6991**	**0.6747**
DeepConvLSTM	GT	GT	0.7564	0.7612	0.7564	0.7612
	GEN	GEN	0.4871	0.4793	**0.7421**	**0.7303**
	GT	GEN	0.0372	0.0057	**0.7564**	**0.7537**
	GEN	GT	0.0201	0.0124	**0.7278**	**0.7119**
DeepConvLSTM_Attention	GT	GT	0.8940	0.8907	0.8940	0.8907
	GEN	GEN	0.7736	0.7744	**0.9083**	**0.9033**
	GT	GEN	0.0315	0.0024	**0.8854**	**0.8750**
	GEN	GT	0.0401	0.0168	**0.8863**	**0.8813**
Transformer	GT	GT	0.8625	0.8624	0.8625	0.8624
	GEN	GEN	0.7679	0.7717	**0.8625**	**0.8563**
	GT	GEN	0.0315	0.0023	**0.8453**	**0.8396**
	GEN	GT	0.0659	0.0236	**0.8453**	**0.8375**

To assess the quality and downstream utility of generated IMU signals, we evaluate classification performance under four training–testing configurations:

Table 4. Classification results on the MM-Fit dataset comparing the baseline generator and PrimeIMU across four training–testing settings: GT→GT (train/test on real IMU), GEN→GEN (train/test on generated IMU), GT→GEN (train on real IMU, test on generated IMU), and GEN→GT (train on generated IMU, test on real IMU). Reported metrics are Accuracy (Acc) and macro-F1; ↑ indicates higher is better.

<table>
<tr><th>Model</th><th>Train</th><th>Test</th><th>Baseline
Acc ↑ | F1 ↑</th><th>PrimeIMU (Ours)
Acc ↑ | F1 ↑</th></tr>
<tr><td>Random Forest</td><td>GT</td><td>GT</td><td>0.4034 | 0.3833</td><td>0.4034 | 0.3833</td></tr>
<tr><td></td><td>GEN</td><td>GEN</td><td>0.2149 | 0.1613</td><td><b>0.3556</b> | <b>0.3345</b></td></tr>
<tr><td></td><td>GT</td><td>GEN</td><td>0.1596 | 0.1287</td><td><b>0.3499</b> | <b>0.3073</b></td></tr>
<tr><td></td><td>GEN</td><td>GT</td><td>0.1664 | 0.1446</td><td><b>0.3614</b> | <b>0.3342</b></td></tr>
<tr><td>SVM</td><td>GT</td><td>GT</td><td>0.3572 | 0.3106</td><td>0.3572 | 0.3106</td></tr>
<tr><td></td><td>GEN</td><td>GEN</td><td>0.1166 | 0.0729</td><td><b>0.3512</b> | <b>0.3522</b></td></tr>
<tr><td></td><td>GT</td><td>GEN</td><td>0.1088 | 0.1483</td><td><b>0.3103</b> | <b>0.3013</b></td></tr>
<tr><td></td><td>GEN</td><td>GT</td><td>0.1357 | 0.1138</td><td><b>0.3492</b> | <b>0.3086</b></td></tr>
<tr><td>DeepConvLSTM</td><td>GT</td><td>GT</td><td>0.3686 | 0.3488</td><td>0.3686 | 0.3488</td></tr>
<tr><td></td><td>GEN</td><td>GEN</td><td>0.2461 | 0.1890</td><td><b>0.3429</b> | <b>0.3063</b></td></tr>
<tr><td></td><td>GT</td><td>GEN</td><td>0.1437 | 0.1140</td><td><b>0.3218</b> | <b>0.2777</b></td></tr>
<tr><td></td><td>GEN</td><td>GT</td><td>0.0988 | 0.0770</td><td><b>0.3322</b> | <b>0.3031</b></td></tr>
<tr><td>DeepConvLSTM_Attention</td><td>GT</td><td>GT</td><td>0.3541 | 0.3374</td><td>0.3541 | 0.3374</td></tr>
<tr><td></td><td>GEN</td><td>GEN</td><td>0.2851 | 0.2176</td><td><b>0.3446</b> | <b>0.3044</b></td></tr>
<tr><td></td><td>GT</td><td>GEN</td><td>0.1587 | 0.1485</td><td><b>0.2993</b> | <b>0.2667</b></td></tr>
<tr><td></td><td>GEN</td><td>GT</td><td>0.1624 | 0.1267</td><td><b>0.3144</b> | <b>0.2957</b></td></tr>
<tr><td>Transformer</td><td>GT</td><td>GT</td><td>0.3787 | 0.3631</td><td>0.3787 | 0.3631</td></tr>
<tr><td></td><td>GEN</td><td>GEN</td><td>0.2871 | 0.2106</td><td><b>0.3442</b> | <b>0.2926</b></td></tr>
<tr><td></td><td>GT</td><td>GEN</td><td>0.0743 | 0.0759</td><td><b>0.3086</b> | <b>0.2719</b></td></tr>
<tr><td></td><td>GEN</td><td>GT</td><td>0.1376 | 0.1143</td><td><b>0.3255</b> | <b>0.3030</b></td></tr>
</table>

- **GT→GT**: training and testing on real IMU data, serving as the reference upper bound.
- **GEN→GEN**: training and testing on generated data, probing the internal consistency and informativeness of synthetic signals.
- **GT→GEN**: training on real IMU but testing on generated data, measuring whether synthetic signals are realistic to real-trained classifiers.
- **GEN→GT**: training on generated data but testing on real IMU, the most realistic deployment setting when labeled sensor data are scarce or unavailable.

Tables 3 and 4 report results on UTD-MHAD and MM-Fit across classical (Random Forest, SVM) and neural (DeepConvLSTM, Transformer) classifiers, comparing the baseline generator and PrimeIMU.

GEN→GT: synthetic training, real deployment. This setting is most relevant to practice, since it corresponds to training action recognition models purely on synthetic IMU and deploying them directly on real devices. With baseline synthetic signals, performance drops catastrophically: on UTD-MHAD, all models collapse to near-random accuracies (e.g., Random Forest 0.052/0.022, SVM 0.029/0.002, Transformer 0.066/0.024). These results indicate that baseline synthetic data lack the discriminative dynamics necessary for real-world generalization. In contrast, PrimeIMU nearly closes the gap to GT→GT. For instance, on UTD-MHAD, Transformer achieves 0.845/0.838 (GEN→GT) compared to 0.863/0.862 (GT→GT), and DeepConvLSTM_Attention yields 0.886/0.881 vs. 0.894/0.891. Even simple classifiers benefit: Random Forest reaches 0.794/0.785, and SVM 0.699/0.675, despite being trained entirely on synthetic inputs.

On MM-Fit, which introduces cross-domain challenges such as heterogeneous devices, sensor placements, and activity styles, PrimeIMU still provides substantial improvements. For example, Random Forest improves from 0.166/0.145 (baseline) to 0.361/0.334, and Transformer from 0.138/0.114 to 0.326/0.303. These results demonstrate that PrimeIMU enables synthetic-only training to produce models competitive with those trained on scarce, expensive sensor datasets, even under domain shift.

GEN⟶GEN: internal consistency of synthetic data. Training and testing fully within the generated domain examines whether the synthetic data distribution is coherent and sufficiently informative for learning. The baseline generator fails this test: on UTD-MHAD, accuracies remain below 0.67 across models, reflecting internal inconsistency and unrealistic temporal structure. In contrast, PrimeIMU achieves strong GEN⟶GEN performance across classifiers, often approaching the GT⟶GT ceiling. Notably, on UTD-MHAD, DeepConvLSTM_Attention even surpasses its real-data reference (0.908/0.903 vs. 0.894/0.891), suggesting that our generated signals are not only realistic but sometimes less noisy than raw sensor measurements. On MM-Fit, GEN⟶GEN accuracies remain in the 0.30–0.35 range, significantly outperforming the baseline. These findings confirm that PrimeIMU produces self-consistent synthetic domains rich enough to support high-quality training without collapse.

GT⟶GEN: realism under real-trained models. Evaluating real-trained models on synthetic data probes whether generated signals are perceived as realistic by classifiers trained on true sensors. Baseline signals fail completely: all models collapse to near-zero performance (e.g., Transformer 0.032/0.002, DeepConvLSTM 0.037/0.006). In contrast, PrimeIMU yields stable performance closely aligned with GT⟶GT. On UTD-MHAD, Transformer achieves 0.845/0.840 (GT⟶GEN) vs. 0.863/0.862 (GT⟶GT), and Random Forest 0.788/0.781 vs. 0.802/0.796. Similar trends appear on MM-Fit: although accuracies are lower due to domain shift, models retain meaningful performance when applied to synthetic inputs. These results suggest that PrimeIMU significantly narrows the anatomical–inertial gap, producing signals that real-trained models interpret as plausible sensor readings.

GT⟶GT: reference ceiling. As expected, training and testing on real IMU data produces the highest absolute accuracies, establishing the empirical upper bound. The key observation, however, is how closely PrimeIMU in GEN⟶GT approaches this ceiling. Across UTD-MHAD, the performance difference is typically within 1–2 percentage points across all models. This near-equivalence is striking given that no real IMU data are used for training, highlighting that our generator preserves the discriminative motion cues required for downstream action recognition.

Takeaways. Taken together, these experiments yield three insights. First, classifiers trained on PrimeIMU signals generalize to real data almost as well as those trained on ground truth. Second, improvements extend to the more challenging MM-Fit benchmark, where PrimeIMU consistently outperforms the baseline under domain shift. Third, scalable, privacy-preserving, and cost-effective synthetic IMU generation offers a practical substitute for large-scale sensor collection across healthcare, fitness, and wearable computing scenarios.

4.4 Data Augmentation

We examine whether synthetic IMU data generated by PrimeIMU can improve action recognition performance when used as an augmentation source alongside ground-truth IMUs. Table 5 reports classification accuracy and macro F1 on UTD-MHAD for five representative models, spanning classical classifiers and deep sequential architectures.

Table 5. UTD-MHAD dataset data augmentation results.

Model	Training	Testing	Accuracy	F1 Score (Macro)
RandomForest	GT	GT	0.8023	0.7962
	GT+GEN	GT	**0.8381**	**0.8266**
SVM	GT	GT	0.7106	0.6908
	GT+GEN	GT	**0.7550**	**0.7413**
DeepConvLSTM	GT	GT	0.7564	0.7612
	GT+GEN	GT	**0.8840**	**0.8743**
DeepConvLSTM_Attention	GT	GT	0.8940	0.8907
	GT+GEN	GT	**0.9284**	**0.9247**
Transformer	GT	GT	**0.8625**	**0.8624**
	GT+GEN	GT	0.8467	0.8483

Overall trends. Across four out of five models, adding synthetic data (GT+GEN) improves performance relative to training on real data alone. RandomForest and SVM see moderate but consistent gains (+0.03–0.05 in both accuracy and F1), while deep recurrent networks achieve substantial boosts (DeepConvLSTM: 0.7564 → 0.8840 accuracy; DeepConvLSTM_Attention: 0.8940 → 0.9284). The only exception is the Transformer, where augmentation slightly decreases performance, likely reflecting sensitivity to distributional variance in synthetic signals. Overall, these results suggest that PrimeIMU signals are effective in enriching training distributions and mitigating overfitting for most architectures.

Classical models. For RandomForest, augmenting with synthetic data raises accuracy from 0.8023 to 0.8381 and macro F1 from 0.7962 to 0.8266. Similarly, SVM improves from 0.7106 to 0.7550 accuracy and 0.6908 to 0.7413 F1. These consistent gains indicate that even shallow classifiers can benefit from the additional variability introduced by PrimeIMU, which provides new decision boundaries beyond the limited coverage of real training data.

Deep sequential models. DeepConvLSTM exhibits the largest improvement: accuracy rises from 0.7564 to 0.8840 and F1 from 0.7612 to 0.8743, representing over 12% absolute gains. DeepConvLSTM_Attention also benefits, improving by nearly 4% accuracy and macro F1. These results show that temporal models are particularly adept at absorbing the fine-grained temporal cues encoded in synthetic IMUs, leveraging them to learn richer motion dynamics that generalize better to unseen real signals.

Transformer behavior. The Transformer shows a slight decrease with augmentation (0.8625 → 0.8467 accuracy; 0.8624 → 0.8483 F1). This mixed outcome suggests that when a model already achieves high baseline performance, synthetic data may introduce domain variance that interacts poorly with the model's reliance on global sequence representations. Unlike recurrent networks, which can exploit added local temporal diversity, Transformers may require more careful integration of synthetic and real samples (e.g., curriculum training or weighting strategies).

Takeaways. (1) Consistent gains for most models: PrimeIMU augmentation improves performance for RandomForest, SVM, and both LSTM variants, confirming its utility as a training signal. (2) Large benefits for deep RNNs: Recurrent networks exploit the synthetic diversity most effectively, achieving double-digit improvements in accuracy and F1. (3)

Model sensitivity: The slight drop for Transformers highlights that augmentation is not universally beneficial and may require model-specific strategies.

Overall, these results demonstrate that PrimeIMU can function not only as a substitute for scarce real data but also as a viable augmentation source that strengthens recognition performance in most settings.

4.5 Evaluation of Unseen Activity Generation Quality

To further examine generalization beyond the training distribution, we evaluate the ability of PrimeIMU to generate realistic signals for unseen activities. We simulate this scenario by progressively removing activity classes from training and then testing on those held-out classes. Four configurations are considered: excluding 1, 3, 7, and 13 classes. Performance is reported across accelerometer and gyroscope axes using MAE, RMSE, R^2, and Pearson correlation. Table 6 summarizes the results.

Table 6. Performance of PrimeIMU on unseen activity generation with different numbers of removed classes during training

Configuration	Dimension	MAE↓	RMSE ↓	R^2 ↑	Pearson ↑
w/o 1 class	Accel-X	0.0278	0.0318	-4.5762	0.8223
	Accel-Y	0.0319	0.0347	0.5526	0.9680
	Accel-Z	0.0178	0.0351	-0.9815	0.7919
	Gyro-X	3.1941	4.1490	0.8473	0.9549
	Gyro-Y	2.1869	2.7867	0.4120	0.9200
	Gyro-Z	3.3438	4.5424	0.5921	0.9442
w/o 3 classes	Accel-X	0.0200	0.0249	0.3460	0.9206
	Accel-Y	0.0152	0.0215	0.7880	0.9614
	Accel-Z	0.0206	0.0253	0.3325	0.9298
	Gyro-X	6.3002	6.7170	0.5306	0.9809
	Gyro-Y	6.1725	6.8002	-3.6676	0.8741
	Gyro-Z	3.6351	4.5474	0.2053	0.9547
w/o 7 classes	Accel-X	0.0470	0.0639	-0.6917	0.8887
	Accel-Y	0.0321	0.0457	0.6769	0.9643
	Accel-Z	0.0328	0.0440	0.3705	0.9258
	Gyro-X	5.5519	7.3369	0.4505	0.9772
	Gyro-Y	4.5878	5.9229	-0.7792	0.9254
	Gyro-Z	9.1852	10.4794	-0.7322	0.9777
w/o 13 classes	Accel-X	0.0489	0.0651	0.2695	0.9594
	Accel-Y	0.0501	0.0647	0.5779	0.9779
	Accel-Z	0.0466	0.0592	0.1732	0.9558
	Gyro-X	9.3833	11.8290	-0.8209	0.9755
	Gyro-Y	7.5312	9.5950	-0.6213	0.9545
	Gyro-Z	9.6361	11.5002	-1.4849	0.9751

Overall trends. As the number of unseen classes increases, performance gradually degrades, reflecting the increased difficulty of generalizing to motion patterns absent during training. Nevertheless, even under the most extreme setting (13 classes removed), PrimeIMU maintains relatively strong Pearson correlations (~0.95–0.98 across most axes), indicating that the generated signals still preserve realistic temporal dynamics. In contrast, R^2 declines more sharply and becomes negative for some gyroscope axes, suggesting amplitude mismatches under severe activity shifts.

Accelerometer dynamics. Accelerometer signals remain highly consistent across conditions. For small removals (1 or 3 classes), MAE and RMSE stay very low (~0.02–0.04), and Pearson correlations exceed 0.92 across axes. Even with 13 classes excluded, correlations remain high (0.96–0.98), and R^2 values stay positive on two out of three axes. These results suggest that low-frequency kinematic cues transfer well across unseen activities, enabling PrimeIMU to capture realistic translational dynamics with minimal degradation.

Gyroscope robustness. Gyroscope signals show more sensitivity to unseen activities, especially as more classes are removed. With only one class held out, correlations are already strong (0.92–0.95) and R^2 positive across axes. As removals increase to 7 or 13 classes, MAE and RMSE grow substantially (e.g., Gyro-Z RMSE 10.48), and R^2 becomes negative on multiple axes, indicating scale or bias mismatches. Despite this, Pearson correlations remain consistently high (0.92–0.98), confirming that temporal dynamics are preserved even when magnitude fidelity suffers. This pattern suggests that angular rate scaling is more task-specific and may require lightweight calibration for fully unseen motions.

Takeaways. (1) Strong temporal fidelity: High Pearson correlations across all settings confirm that core motion dynamics are preserved, even for activities unseen during training. (2) Amplitude challenges in gyroscopes: Declining R^2 and larger errors under heavy class removal indicate residual scale mismatches, likely due to activity-specific angular rates. (3) Generalization potential: These results demonstrate that PrimeIMU not only fits training classes but also extends to new activities with realistic signal quality, supporting its scalability for diverse real-world applications.

4.6 Cross-Dataset Transfer Learning for IMU Generation and Classification

A central test of synthetic IMU generation is whether models trained on one dataset can generalize to new domains with different devices, sampling rates, and activity styles. We evaluate this by pretraining on UTD-MHAD and transferring to MM-Fit with varying proportions of target-domain data (10%, 20%, 30%, 40%). Table 7 reports signal-level fidelity, while Table 8 presents downstream action recognition classification results.

Signal-level adaptation across domains. Table 7 demonstrates that PrimeIMU adapts rapidly to the new domain with minimal supervision. Even at 10% fine-tuning, generated signals already achieve strong temporal fidelity: Pearson correlations exceed 0.92 on almost all axes (e.g., Gyro-X 0.976, Accel-Z 0.944), confirming that activity-relevant dynamics transfer smoothly across datasets. However, variance explanation (R^2) remains limited or negative on some axes (Accel-Y −5.55, Gyro-Y −9.41), pointing to residual amplitude mismatches. With 20% fine-tuning, fidelity improves substantially: Accel-Z attains MAE 0.1146, RMSE 0.1451, and Pearson 0.974, with R^2 turning positive (0.52); Gyro-X achieves MAE 0.0342, $R^2 = 0.76$, and Pearson 0.983. These values approach saturation, as additional data (30–40%) yield only marginal or inconsistent gains. For example, Gyro-Z improves from $R^2 = 0.18$ at 30% to negative values at 40%, despite stable Pearson around 0.96, suggesting that excessive fine-tuning may overfit idiosyncratic device biases. Across all axes, correlations remain robust ($\rho > 0.90$) regardless of R^2, showing that PrimeIMU consistently captures the correct temporal structure even when magnitude scaling is imperfect. Together, these results highlight three points: (i) strong low-data efficiency, with 10–20% target data sufficient to recover domain-specific dynamics; (ii) robustness of temporal alignment, evidenced by consistently high Pearson values; and (iii) residual amplitude or scale mismatches in gyroscopes, which are easily correctable with lightweight calibration.

Downstream classification with synthetic signals. Signal fidelity translates directly into action recognition performance, as shown in Table 8. Several consistent patterns emerge. First, classifiers trained on generated data and evaluated on real sensors (GEN→GT) achieve accuracy and F1 scores remarkably close to the real-data ceiling (GT→GT). For example,

Table 7. Cross-dataset IMU generation quality when transferring from UTD-MHAD to MM-Fit. Models are pretrained on UTD-MHAD and fine-tuned with different proportions of MM-Fit data (10%, 20%, 30%, 40%). We report signal-level metrics across six sensor dimensions: mean absolute error (MAE), root mean squared error (RMSE), coefficient of determination (R^2), and Pearson correlation. These generated signals serve as the basis for the classification experiments in Table 8.

Dimension	Data Proportion	MAE	RMSE	R^2	Pearson
Accel-X	10%	0.1448	0.1934	0.4315	0.9261
	20%	0.1515	0.1993	**0.7276**	**0.9571**
	30%	0.1477	0.1946	0.4309	0.9247
	40%	**0.1294**	**0.1797**	0.6439	0.9416
Accel-Y	10%	0.4493	0.4911	-5.5504	0.8799
	20%	0.5704	0.7809	-0.0863	0.6772
	30%	0.2259	0.2699	**0.2087**	**0.9511**
	40%	**0.1935**	**0.2626**	-0.3392	0.9116
Accel-Z	10%	0.2059	0.2499	-0.0960	0.9444
	20%	**0.1146**	**0.1451**	0.5209	**0.9743**
	30%	0.1904	0.2321	0.3915	0.9528
	40%	0.1191	0.1619	**0.7264**	0.9533
Gyro-X	10%	0.0500	0.0634	0.2247	0.9760
	20%	**0.0342**	**0.0418**	**0.7645**	**0.9827**
	30%	0.0558	0.0679	-0.6523	0.9575
	40%	0.0667	0.0783	0.0155	0.9747
Gyro-Y	10%	0.0801	0.0863	-9.4137	0.9598
	20%	**0.0266**	**0.0318**	0.2922	0.9738
	30%	0.0442	0.0506	-0.3174	0.9728
	40%	0.0288	0.0352	**0.5606**	**0.9787**
Gyro-Z	10%	0.0538	0.0675	-1.5817	0.9440
	20%	**0.0446**	**0.0513**	**0.6121**	**0.9746**
	30%	0.0531	0.0585	0.1843	0.9622
	40%	0.0591	0.0680	-5.4008	0.9633

with 20% fine-tuning, the Transformer reaches 0.345/0.317 (Acc/F1), nearly matching the GT→GT result of 0.379/0.363. DeepConvLSTM shows a similar trend, achieving 0.360/0.333 under GEN→GT versus 0.369/0.349 under GT→GT. Even classical models remain competitive: Random Forest attains 0.366/0.340 at 10% and rises slightly to 0.371/0.348 at 40%, while SVM maintains GEN→GT performance around 0.350, only marginally below its GT→GT ceiling of 0.357. These results confirm that discriminative activity patterns are faithfully preserved in PrimeIMU signals across architectures.

Second, GEN→GEN experiments verify internal consistency: models trained and tested entirely on synthetic data sustain performance close to GT→GT. For instance, SVM achieves 0.353/0.353 at 30% fine-tuning, essentially matching its real-data counterpart. This shows that the synthetic distribution is not only realistic relative to real sensors but also coherent in itself, allowing classifiers to generalize within the generated domain.

Third, GT→GEN experiments measure realism as perceived by real-trained models. Unlike baseline methods where performance collapses, PrimeIMU maintains stable results: Random Forest achieves 0.368/0.323 at 30%, close to its GT→GT reference of 0.403/0.383, and Transformer achieves 0.335/0.285, not far from 0.379/0.363. This indicates that synthetic signals are sufficiently aligned with real sensor characteristics to be interchangeable at inference time.

Table 8. Action recognition on MM-Fit using synthetic IMU signals generated in Table 7. We evaluate four train/test configurations: (1) GT→GT, training and testing on ground-truth IMU; (2) GEN→GEN, training and testing on PrimeIMU-generated IMU; (3) GT→GEN, trained on real and tested on synthetic; and (4) GEN→GT, trained on synthetic and tested on real. Results are reported under different fine-tuning proportions of MM-Fit (10%, 20%, 30%, 40%).

Model	**Train**	**Test**	**10%**		**20%**		**30%**		**40%**	
			Acc	F1	Acc	F1	Acc	F1	Acc	F1
Random Forest	GT	GT	0.4034	0.3833	0.4034	0.3833	0.4034	0.3833	0.4034	0.3833
	GEN	GEN	0.3490	0.3241	**0.3508**	**0.3278**	0.3481	0.3268	0.3495	0.3255
	GT	GEN	0.3582	0.3176	0.3569	0.3182	0.3591	0.3222	**0.3675**	**0.3289**
	GEN	GT	0.3663	0.3404	0.3625	0.3382	0.3681	0.3455	**0.3711**	**0.3477**
SVM	GT	GT	0.3572	0.3106	0.3572	0.3106	0.3572	0.3106	0.3572	0.3106
	GEN	GEN	0.3508	0.3518	0.3512	0.3531	**0.3530**	**0.3533**	0.3516	0.3525
	GT	GEN	**0.3112**	**0.3020**	0.3081	0.2989	0.3081	0.2990	0.3055	0.2917
	GEN	GT	0.3505	0.3084	0.3503	0.3086	**0.3524**	**0.3116**	0.3524	0.3110
DeepConvLSTM	GT	GT	0.3686	0.3488	0.3686	0.3488	0.3684	0.3423	0.3684	0.3423
	GEN	GEN	**0.3556**	0.3009	0.3424	**0.3134**	0.3530	0.3003	0.3235	0.2805
	GT	GEN	0.3174	0.2708	0.3213	0.2792	**0.3363**	**0.2866**	0.3275	0.2800
	GEN	GT	0.3427	0.3177	0.3337	0.3123	**0.3604**	**0.3328**	0.3461	0.3192
DeepConvLSTM_Attention	GT	GT	0.3541	0.3374	0.3642	0.3440	0.3568	0.3405	0.3568	0.3405
	GEN	GEN	**0.3481**	**0.3103**	0.3231	0.2775	0.3429	0.2834	0.3371	0.3023
	GT	GEN	0.3160	0.2724	0.3156	0.2736	0.3182	0.2733	**0.3191**	**0.2777**
	GEN	GT	**0.3385**	**0.3151**	0.3274	0.3058	0.3335	0.3127	0.3293	0.3052
Transformer	GT	GT	0.3787	0.3631	0.3787	0.3507	0.3740	0.3507	0.3740	0.3507
	GEN	GEN	0.3240	0.2694	**0.3490**	**0.3089**	0.3389	0.2849	0.3411	0.2998
	GT	GEN	0.3310	0.2813	0.3279	0.2716	**0.3349**	**0.2847**	0.3292	0.2806
	GEN	GT	0.3314	0.3014	0.3312	0.3059	**0.3448**	**0.3168**	0.3280	0.3027

Finally, across all configurations, performance stabilizes quickly with little additional gain beyond 20% fine-tuning. This plateau mirrors the signal-level results and emphasizes that the majority of cross-domain variance is captured early, making further data annotation costly but unnecessary.

Takeaways. Taken together, the two perspectives—signal fidelity and downstream classification—converge on the same conclusion: PrimeIMU establishes cross-dataset generalization. The framework adapts to new domains with as little as 10–20% target data, producing signals that are both temporally faithful and discriminative for activity recognition. At the signal level, correlations consistently exceed 0.90 while errors shrink rapidly with minimal fine-tuning. At the classification level, GEN→GT performance consistently tracks GT→GT across diverse models, from classical baselines to deep sequence networks. This combination of high-fidelity synthesis and robust downstream transfer positions PrimeIMU as a practical solution for deploying action recognition systems across heterogeneous sensor configurations and activity types without the burden of collecting large-scale labeled IMU data in every new environment.

4.7 Ablation Analysis

We ablate PrimeIMU along two axes: (*i*) input sources—low-frequency pose guidance and physics-inspired simulated IMU initialization; and (*ii*) loss terms—time-domain reconstruction loss $\mathcal{L}_{\text{recon}}$, multi-resolution spectral alignment loss $\mathcal{L}_{\text{stft}}$, and kinematic consistency loss $\mathcal{L}_{\text{kine}}$. We report MAE/RMSE together with R^2 and Pearson to jointly probe amplitude fidelity and temporal trend agreement. Results on UTD-MHAD are summarized in Table 9.

Table 9. Ablation study of PrimeIMU on the UTD-MHAD dataset. We compare the full model with variants where key components are removed: pose sequence input (w/o pose), simulated IMU initialization from the physics stage (w/o phy imu), and individual loss terms for kinematic consistency ($\mathcal{L}_{\text{kine}}$), spectral alignment ($\mathcal{L}_{\text{stft}}$), and time-domain reconstruction ($\mathcal{L}_{\text{recon}}$). Results are reported across six sensor dimensions (Accel-X/Y/Z, Gyro-X/Y/Z) using MAE, RMSE, R^2, and Pearson correlation.

Dimension	Model	MAE	RMSE	R²	Pearson
Accel-X	Full	0.0306	0.0399	0.8706	**0.9886**
	w/o pose	**0.0235**	**0.0319**	**0.9323**	0.9811
	w/o phy imu	0.0722	0.1118	0.3557	0.8347
	w/o $\mathcal{L}_{\text{kine}}$	0.0295	0.0396	0.8902	0.9762
	w/o $\mathcal{L}_{\text{stft}}$	0.0745	0.1117	0.3190	0.9494
	w/o $\mathcal{L}_{\text{recon}}$	0.0401	0.0520	0.1943	0.9708
Accel-Y	Full	**0.0176**	**0.0230**	**0.9785**	**0.9977**
	w/o pose	0.0234	0.0326	0.9585	0.9900
	w/o phy imu	0.0285	0.0384	0.9679	0.9933
	w/o $\mathcal{L}_{\text{kine}}$	0.0275	0.0373	0.9721	0.9918
	w/o $\mathcal{L}_{\text{stft}}$	0.0284	0.0366	0.9725	0.9922
	w/o $\mathcal{L}_{\text{recon}}$	0.0374	0.0475	0.9349	0.9912
Accel-Z	Full	**0.0189**	**0.0262**	**0.9583**	**0.9916**
	w/o pose	0.0276	0.0349	0.9343	0.9911
	w/o phy imu	0.0349	0.0459	0.8822	0.9822
	w/o $\mathcal{L}_{\text{kine}}$	0.0396	0.0496	0.8891	0.9763
	w/o $\mathcal{L}_{\text{stft}}$	0.0250	0.0323	0.9205	0.9884
	w/o $\mathcal{L}_{\text{recon}}$	0.0440	0.0559	0.7398	0.9741
Gyro-X	Full	**3.0549**	**4.2513**	**0.9842**	**0.9972**
	w/o pose	4.5961	5.7856	0.9462	0.9944
	w/o phy imu	4.8794	6.4089	0.9337	0.9910
	w/o $\mathcal{L}_{\text{kine}}$	4.0421	5.6858	0.9814	0.9926
	w/o $\mathcal{L}_{\text{stft}}$	4.1948	5.5147	0.9750	0.9945
	w/o $\mathcal{L}_{\text{recon}}$	9.4388	11.5457	0.8519	0.9862
Gyro-Y	Full	**3.1581**	**4.0785**	0.8006	**0.9957**
	w/o pose	3.5308	4.2360	0.8109	0.9938
	w/o phy imu	4.9996	6.6176	**0.8989**	0.9736
	w/o $\mathcal{L}_{\text{kine}}$	4.3202	5.9407	0.6626	0.9528
	w/o $\mathcal{L}_{\text{stft}}$	4.6992	5.9767	0.7854	0.9808
	w/o $\mathcal{L}_{\text{recon}}$	5.5411	7.0502	0.5790	0.9813
Gyro-Z	Full	**2.9964**	**3.8424**	0.8852	**0.9983**
	w/o pose	4.3181	5.1476	0.8355	0.9965
	w/o phy imu	5.2329	6.6543	0.8841	0.9926
	w/o $\mathcal{L}_{\text{kine}}$	4.8615	5.9656	0.8339	0.9935
	w/o $\mathcal{L}_{\text{stft}}$	5.1138	6.4306	**0.9557**	0.9906
	w/o $\mathcal{L}_{\text{recon}}$	6.9763	8.8849	0.8483	0.9863

Input sources: pose guidance and physics prior are complementary. Removing the physics-based prior (w/o phy imu) consistently harms all accelerometer axes (e.g., Accel-X MAE 0.0722 vs. 0.0306, Pearson 0.8347 vs. 0.9886) and gyroscopes (Gyro-X MAE 4.8794 vs. 3.0549), indicating that the prior supplies crucial high-frequency structure that pure learning cannot easily infer from sparse video kinematics. Conversely, removing pose guidance (w/o pose) mainly affects temporal alignment and gravity-related components: correlations drop across accelerometers (e.g., Accel-X Pearson 0.9811 vs. 0.9886, Accel-Z 0.9911 vs. 0.9916) and gyroscopes (Gyro-X 0.9944 vs. 0.9972), while RMSE grows on most axes. A notable *edge case* is Accel-X where w/o pose yields slightly lower error than the full model (MAE 0.0235 vs. 0.0306, RMSE 0.0319 vs. 0.0399) but also a lower Pearson (0.9811 vs. 0.9886). This pattern—lower absolute error yet weaker correlation—suggests that dropping pose can reduce a constant bias along one axis (benefiting MAE/RMSE) at the expense of *temporal* fidelity. Since downstream action recognition relies more on correct dynamics than a small static offset, the full model remains preferable. Overall, the two inputs are *complementary*: the physics prior contributes device-like high-frequency behavior; pose guidance anchors global motion and gravity direction. Using both yields the best joint fidelity.

Loss terms: each constrains a different failure mode. Time reconstruction loss $\mathcal{L}_{\text{recon}}$ controls amplitude and bias. Removing it causes the largest degradation on gyros (Gyro-X MAE 9.44 vs. 3.05, Gyro-Z MAE 6.98 vs. 3.00) and markedly worse R^2 (e.g., Gyro-X 0.85 vs. 0.98), even when Pearson stays relatively high (0.986). This is the classic "high-correlation, wrong-scale" failure: temporal shape is roughly right but magnitudes drift, which is detrimental for device-faithful synthesis. Spectral alignment $\mathcal{L}_{\text{stft}}$ loss shapes frequency content. Dropping it inflates accelerometer errors (Accel-X MAE 0.0745 vs. 0.0306, Accel-Z RMSE 0.0323 vs. 0.0262) and reduces R^2 (Accel-X 0.319 vs. 0.871), confirming that STFT matching is key to suppressing spurious high-frequency spikes from differentiation and to restoring IMU-like spectra. On gyros, removing $\mathcal{L}_{\text{stft}}$ also increases scale error (Gyro-Z MAE 5.11 vs. 3.00) and slightly lowers Pearson (0.991), indicating a broader role beyond pure accelerometry. Kinematic consistency $\mathcal{L}_{\text{kine}}$ loss enforces agreement with the guiding pose after double integration. Disabling it especially hurts axes that are sensitive to accumulated drift (e.g., Accel-Z RMSE 0.0496 vs. 0.0262; Gyro-Z MAE 4.86 vs. 3.00) and reduces correlations on multiple axes (e.g., Accel-X Pearson 0.976 vs. 0.989). This shows that $\mathcal{L}_{\text{kine}}$ acts as a powerful regularizer against non-physical accelerations and helps the network respect coarse kinematic constraints while learning fine-grained inertial details.

Axis-specific observations and diagnostics. Accelerometers benefit the most from the full composite: with all components enabled, Pearson is near-ceiling across axes (e.g., Accel-Y 0.998) and R^2 is high (0.96–0.98). When $\mathcal{L}_{\text{stft}}$ or the physics prior is removed, accelerometer errors rise sharply, revealing their sensitivity to both spectral realism and proper high-frequency priors. Gyroscopes show very strong temporal alignment even under ablations (Pearson typically > 0.99), but are more prone to *scale* errors without $\mathcal{L}_{\text{recon}}$ or $\mathcal{L}_{\text{stft}}$ (e.g., Gyro-X MAE 9.44 w/o $\mathcal{L}_{\text{recon}}$). This "right-shape/wrong-scale" signature—high Pearson, degraded R^2/MAE—is consistent with device- and placement-specific angular-rate biases and confirms why an explicit amplitude-alignment term is necessary. In practice, any residual per-axis bias can be further reduced by a lightweight post-hoc affine calibration, but the full objective already mitigates most of it.

Why the full model is still preferred despite a few lower per-axis errors in ablations. Instances like Accel-X (slightly lower MAE without pose) reflect a local trade-off: optimizing *only* MAE/RMSE on one axis can occasionally benefit from removing constraints, but it comes with worse temporal agreement (lower Pearson) and weaker robustness across the remaining axes. Our goal is *sensor-faithful, deployable* synthesis across all channels, not isolated gains. The full model delivers the best *joint* profile—high correlations, strong R^2, and low errors—consistent with its superior downstream classification (Section 4).

Takeaways. (i) *Both inputs matter*: the prior physics supply high-frequency device-like dynamics; pose guidance stabilizes global kinematics and gravity, and the combination is strictly better than either alone. (ii) *All three losses are necessary and complementary*: $\mathcal{L}_{\text{recon}}$ fixes amplitude/bias, $\mathcal{L}_{\text{stft}}$ restores IMU spectra and suppresses spurious spikes, $\mathcal{L}_{\text{kine}}$ prevents nonphysical accelerations and drift. (iii) *Axis behavior is consistent with sensor physics*: accelerometers are spectrum-sensitive; gyros are correlation-robust but scale-sensitive. In general, the full PrimeIMU configuration yields the most faithful reproduction of real sensors across axes and metrics, aligning with the improvements observed in generation fidelity and downstream action recognition performance.

5 Conclusion

We introduced PrimeIMU, a framework for synthesizing high-frequency, high-fidelity IMU signals directly from video. Our study demonstrates that PrimeIMU addresses three long-standing challenges in IMU generation: (i) error propagation from imperfect pose estimation, (ii) the loss of fine-grained motion dynamics caused by low video frame rates, and (iii) the mismatch between kinematic estimates and the measurements by physical IMU sensors. By combining physics-guided IMU simulation with a hybrid U-Net refinement module, PrimeIMU first converts video-derived kinematic trajectories into approximate inertial measurements and then learns to refine these simulated signals to more faithfully capture the dynamics and sensor characteristics of real IMU devices. Extensive experiments show that PrimeIMU (1) improves the fidelity of inertial signal synthesis, (2) generalizes to unseen motion patterns while preserving realistic signal characteristics, (3) enables synthetic-only training of downstream models with performance approaching that of real-sensor training and provides additional gains when used to augment real data, and (4) adapts across datasets and sensor configurations with minimal fine-tuning, supporting robust downstream performance under domain shift.

References

[1] Ghaith Al-refai, Dina Karasneh, Hisham Elmoaqet, Mutaz Ryalat, and Natheer Almtireen. 2025. Surface Classification from Robot Internal Measurement Unit Time-Series Data Using Cascaded and Parallel Deep Learning Fusion Models. *Machines* 13, 3 (2025), 251.

[2] Lei Bai, Lina Yao, Xianzhi Wang, Salil S Kanhere, Bin Guo, and Zhiwen Yu. 2020. Adversarial multi-view networks for activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–22.

[3] Dmitrijs Balabka. 2019. Semi-supervised learning for human activity recognition using adversarial autoencoders. In *Adjunct proceedings of the 2019 ACM international joint conference on pervasive and ubiquitous computing and proceedings of the 2019 ACM international symposium on wearable computers*. 685–688.

[4] Chen Chen, Roozbeh Jafari, and Nasser Kehtarnavaz. 2015. UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In *International Conference on Image Processing*.

[5] Wenqiang Chen, Shupei Lin, Elizabeth Thompson, and John Stankovic. 2021. Sensecollect: We need efficient ways to collect on-body sensor-based human activity data! *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021), 1–27.

[6] Dongzhou Cheng, Lei Zhang, Can Bu, Xing Wang, Hao Wu, and Aiguo Song. 2023. ProtoHAR: Prototype guided personalized federated learning for human activity recognition. *IEEE Journal of Biomedical and Health Informatics* 27, 8 (2023), 3900–3911.

[7] Giovanni Cioffi, Leonard Bauersfeld, Elia Kaufmann, and Davide Scaramuzza. 2023. Learned inertial odometry for autonomous drone racing. *IEEE Robotics and Automation Letters* 8, 5 (2023), 2684–2691.

[8] Asiye Demirtaş, Gökhan Erdemir, and Haluk Bayram. 2024. Indoor surface classification for mobile robots. *PeerJ Computer Science* 10 (2024), e1730.

[9] Siwei Feng and Marco F Duarte. 2019. Few-shot learning-based human activity recognition. *Expert Systems with Applications* 138 (2019), 112782.

[10] Iuri Frosio, Federico Pedersini, and N Alberto Borghese. 2008. Autocalibration of MEMS accelerometers. *IEEE Transactions on instrumentation and measurement* 58, 6 (2008), 2034–2041.

[11] Walid Gomaa and Mohamed A Khamis. 2023. A perspective on human activity recognition from inertial motion data. *Neural Computing and Applications* (2023).

[12] 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.

[13] 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* (2020).

[14] Arttu Lämsä, Jaakko Tervonen, Jussi Liikka, Constantino Álvarez Casado, and Miguel Bordallo López. 2022. Video2IMU: Realistic IMU features and signals from videos. In *International Conference on Wearable and Implantable Body Sensor Networks*.

[15] Zikang Leng, Amitrajit Bhattacharjee, Hrudhai Rajasekhar, Lizhe Zhang, Elizabeth Bruda, Hyeokhyen Kwon, and Thomas Plötz. 2024. Imugpt 2.0: Language-based cross modality transfer for sensor-based human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* (2024).

[16] Zikang Leng, Hyeokhyen Kwon, and Thomas Ploetz. 2023. Generating Virtual On-Body Accelerometer Data from Virtual Textual Descriptions for Human Activity Recognition. In *International Symposium on Wearable Computers*.

[17] Faisal Mohd-Yasin, Can E Korman, and David J Nagel. 2003. Measurement of noise characteristics of MEMS accelerometers. *Solid-State Electronics* 47, 2 (2003), 357–360.

[18] Andreas Mueller. 2019. Modern robotics: Mechanics, planning, and control [bookshelf]. *IEEE Control Systems Magazine* 39, 6 (2019), 100–102.

[19] Bartłomiej Nalepa, Magdalena Pawlyta, Mateusz Janiak, Agnieszka Szczęsna, Aleksander Gwiazda, and Konrad Wojciechowski. 2022. Recreating the Motion Trajectory of a System of Articulated Rigid Bodies on the Basis of Incomplete Measurement Information and Unsupervised Learning. *Sensors* 22, 6 (2022), 2198.

[20] Francisco Javier Ordóñez and Daniel Roggen. 2016. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors* (2016).

[21] Aakash Parmar, Rakesh Katariya, and Vatsal Patel. 2018. A review on random forest: An ensemble classifier. In *International Conference on Intelligent Data Communication Technologies and Internet of Things*.

[22] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 2019. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Conference on Computer Vision and Pattern Recognition*.

[23] Thomas Plötz. 2023. If only we had more data!: Sensor-based human activity recognition in challenging scenarios. In *2023 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. IEEE, 565–570.

[24] Sathian Pookkuttath, Povendhan Arthanaripalayam Palanisamy, and Mohan Rajesh Elara. 2023. AI-Enabled Condition Monitoring Framework for Outdoor Mobile Robots Using 3D LiDAR Sensor. *Mathematics* 11, 16 (2023), 3594.

[25] Yuantian Qin, Zhehang Yin, Quanou Yang, and Kai Zhang. 2024. Dynamics Parameter Identification of Articulated Robot. *Machines* 12, 9 (2024), 595.

[26] Yuheng Qiu, Can Xu, Yutian Chen, Shibo Zhao, Junyi Geng, and Sebastian Scherer. 2025. AirIO: Learning Inertial Odometry with Enhanced IMU Feature Observability. *arXiv preprint arXiv:2501.15659* (2025).

[27] Daniele Ravi, Charence Wong, Benny Lo, and Guang-Zhong Yang. 2016. Deep learning for human activity recognition: A resource efficient implementation on low-power devices. In *2016 IEEE 13th international conference on wearable and implantable body sensor networks (BSN)*. IEEE, 71–76.

[28] Aaqib Saeed, Tanir Ozcelebi, and Johan Lukkien. 2019. Multi-task self-supervised learning for human activity detection. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 2 (2019), 1–30.

[29] Panneer Selvam Santhalingam, Parth Pathak, Huzefa Rangwala, and Jana Kosecka. 2023. Synthetic smartwatch imu data generation from in-the-wild asl videos. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* (2023).

[30] Allah Bux Sargano, Xiaofeng Wang, Plamen Angelov, and Zulfiqar Habib. 2017. Human action recognition using transfer learning with deep representations. In *2017 International joint conference on neural networks (IJCNN)*. IEEE, 463–469.

[31] Ronald W Schafer. 2011. What is a savitzky-golay filter?[lecture notes]. *IEEE Signal Processing Magazine* (2011).

[32] Satya P Singh, Madan Kumar Sharma, Aimé Lay-Ekuakille, Deepak Gangwar, and Sukrit Gupta. 2020. Deep ConvLSTM with self-attention for human activity decoding using wearable sensors. *IEEE Sensors Journal* (2020).

[33] David Strömbäck, Sangxia Huang, and Valentin Radu. 2020. Mm-fit: Multimodal deep learning for automatic exercise logging across sensing devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* (2020).

[34] Ruslan Sultan and Steffen Greiser. 2025. Time and Frequency Domain Analysis of IMU-based Orientation Estimation Algorithms with Comparison to Robotic Arm Orientation as Reference. *Sensors* 25, 16 (2025), 5161.

[35] Shan Suthaharan. 2016. Support vector machine. In *Machine learning models and algorithms for big data classification: thinking with examples for effective learning*. Springer.

[36] Nilay Tufek, Murat Yalcin, Mucahit Altintas, Fatma Kalaoglu, Yi Li, and Senem Kursun Bahadir. 2019. Human action recognition using deep learning methods on limited sensory data. *IEEE Sensors Journal* (2019).

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).

[38] Gregory W Vogl, M Alkan Donmez, Andreas Archenti, and Brian A Weiss. 2016. Inertial measurement unit for on-Machine diagnostics of machine tool linear axes. In *Annual Conference of the PHM Society*, Vol. 8.

[39] Cong Xu, Yuhang Li, Dae Lee, Dae Hoon Park, Hongda Mao, Huyen Do, Jonathan Chung, and Dinesh Nair. 2023. Augmentation robust self-supervised learning for human activity recognition. In *International Conference on Acoustics, Speech and Signal Processing*.

[40] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. 2023. Vitpose++: Vision transformer for generic body pose estimation. *Transactions on Pattern Analysis and Machine Intelligence* (2023).

[41] Xiyuan Zhang, Diyan Teng, Ranak Roy Chowdhury, Shuheng Li, Dezhi Hong, Rajesh Gupta, and Jingbo Shang. 2024. UniMTS: Unified Pre-training for Motion Time Series. In *Advances in Neural Information Processing Systems*.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009