Beyond 200 Hz: An Evaluation of Low-Rate IMU Sampling for Pedestrian Inertial Odometry

PAVLO HRECHKO, University of Twente, The Netherlands

Mobile and wearable devices with limited energy resources increasingly rely on inertial measurement unit (IMUs) for pedestrian localisation. This is necessary in situations where the global navigation satellite system (GNSS) or Wi-Fi signals are not available. Although state-of-the-art neural inertial odometry models such as RoNIN achieve meter-level accuracy at high sampling rates such as 200 Hz [9], operating at these rates significantly drain battery life and computational resources. This paper investigates the minimum viable IMU sampling rate capable of maintaining acceptable pedestrian inertial odometry accuracy. To maintain comparable localisation accuracy at lower sampling rates, two compensation strategies are investigated: (i) temporal upsampling of low-rate IMU data, and (ii) training neural odometry models directly with downsampled data. Extensive experiments conducted using the RoNIN dataset and custom-collected iOS pedestrian IMU data show a rapid increase in drift as sampling rate decreases. Temporal upsampling, even with sophisticated Kalman–RTS smoothing, fails to recover lost high-frequency information, resulting in severe localisation errors (>170 m). Direct retraining of RoNIN at reduced frequencies (10–150 Hz) significantly outperformed naive interpolation methods, yet accuracy remained below the 200 Hz baseline. Findings indicate that 40 Hz represents a practical lower bound for applications tolerant of moderate drift (~10 m after several minutes), while rates below 30 Hz lead to unacceptable error (≥ 14 m after several minutes).

CCS Concepts: • **Computing methodologies** → **Model verification and validation**.

Additional Key Words and Phrases: IMU, inertial odometry, neural networks, RoNIN, low-rate sampling, pedestrian localisation, upsampling, downsampling, energy efficiency

1 INTRODUCTION

Energy-constrained mobile and wearable devices are relying more and more on IMUs. This occurs when GNSS or Wi-Fi positioning are unavailable. Recent state-of-the-art neural inertial odometry like RoNIN [9] has achieved meter-level accuracy at a 200 Hz sampling rate, however, operating at this sampling rate rapidly exhausts both the sensor's energy budget and the computational resources required for always-on systems. Lowering the sample rate appears to be the solution, but doing so causes the sensor to produce unreliable data and rapidly degrade the quality of localisation. This paper therefore asks: *what is the minimum IMU sampling rate that still yields acceptable pedestrian accuracy, and how can signal-processing or learning compensate for information lost at low rates (e.g., below 30 Hz)?* This study is structured around three central research questions:

RQ1: Given a start reference, what is the lowest IMU sampling rate that keeps the drift $\leq X$ m after 30 minutes?

RQ2: Can upsampling IMU streams collected at lower sampling rates enable an unchanged 200 Hz RoNIN model to achieve the same drift performance as with native high-rate input?

RQ3: To what extent can a RoNIN-style network trained directly on downsampled data (e.g., 20–100 Hz) localise?

The rest of the paper is organized as follows. In Section 2, we review prior work. In Section 3, we discuss background on IMU data processing. Section 4 details the methodology, outlining two main experimental scenarios: (i) evaluating whether upsampling low-rate IMU data can recover information lost due to reduced sampling rates, and (ii) assessing the effectiveness of training the neural odometry model directly on downsampled data. Section 5 presents the results of experiments, analysing the impact of different sampling rates and compensation methods on localisation accuracy and drift. Section 6 provides the conclusion and discusses directions for future work.

2 RELATED WORK

This section reviews the evolution of IMU-only localisation chronologically.

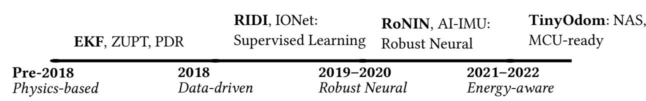

Figure 1. Timeline of IMU-only localisation development.

Early Challenges in Inertial Dead Reckoning (Pre-2018): Traditional inertial navigation systems (INS) estimate position and orientation using double integration of accelerometer and gyroscope data. IMUs operating with low-grade microelectromechanical systems (MEMS) induce small biases that drift as a cubic function of time [8]. High-grade IMUs reduce the drift, but mobile devices are constrained by power and cost limitations [8]. Traditional approaches use extended Kalman filters (EKF) that track position, velocity, attitude, and sensor biases [8] and use magnetometer readings to correct for heading, however the indoor environment often has disturbances that can affect performance [8]. Pedestrian dead-reckoning (PDR) incorporates zero-velocity updates (ZUPT) that set the velocity to zero whenever the foot is in contact with the ground, helping to minimize drift at each step [8]. Step-and-heading approaches detect foot strikes, estimate step length, and accumulate 2D motion vectors, thus trading off 3D detail for less drift [8].

Data-Driven IMU Odometry Emerges (2018): Robust IMU Double Integration (RIDI), which uses supervised learning to estimate velocity from raw IMU data, was introduced in [14]. Accelerations are corrected to remove gravity, prior to the integration, improving the accuracy of velocity estimates [14]. RIDI also classifies smartphone placement to select appropriate regression models, mitigating drift without the use of foot-mounted sensors.

TScIT 43, July 4, 2025, Enschede, The Netherlands

Concurrent with RIDI, Chen et al. proposed IONet, an end-to-end recurrent neural network that directly outputs position and heading changes from raw IMU sequences. IONet uses a Long Short-Term Memory (LSTM) architecture to process the sequential data [3]. IONet learns sensor error dynamics, handling arbitrary orientations, gait variations, and sampling jitter, outperforming traditional PDR and INS on OxIOD benchmarks.

Advances in Robust Neural Inertial Navigation (2019–2020): Herath et al. proposed robust neural inertial navigation (RoNIN) using a large-scale dataset (100 subjects, ~40 h) and temporal convolutional networks (TCN), normalizing predictions to a heading agnostic frame and adding a velocity-based loss to stabilize trajectories [9]. RoNIN outperformed both RIDI and IONet, demonstrating a lower drift on its own dataset as well as on the OxIOD benchmark.

Brossard et al. presented AI-IMU Dead-Reckoning, which combines an invariant EKF with a neural network that dynamically adjusts filter covariance parameters. This approach achieved a translational error of 1.10% on the KITTI benchmark [7] using IMU-only data [1].

Toward Energy-Aware and Lightweight IMU Localisation (2021–2022): Saha et al. introduced TinyOdom, a hardware- and quantization-aware neural architecture search (NAS) framework producing models 31×–134×smaller than prior networks, deployable on MCUs with <128 KB RAM, and maintaining 2.5–12 m drift over 60 s across scenarios [4].

To conclude, IMU-only localisation has developed from physics-based EKFs and ZUPT-restricted PDR to robust data driven methods that learn how to solve drift problems and adapt to a variety of different conditions, and there are also energy-efficient model designs for embedded applications.

3 BACKGROUND

The foundation of this research is the RoNIN framework presented by Herath et al. [9], which represents the state-of-the-art in data-driven pedestrian inertial odometry. RoNIN's contribution is threefold: a large-scale dataset, a novel neural architecture, and a comprehensive evaluation. The experiments in this paper use the RoNIN dataset and model as the primary baseline for evaluating the effects of low-rate sampling.

The RoNIN dataset was collected using a unique two-device protocol: a body-mounted Google Tango device, which is a 3D tracking phone that provides high-accuracy ground truth trajectories, while a second phone recorded the IMU data used for training. This setup allowed subjects to handle the IMU phone naturally (e.g., in a pocket, hand, or bag), capturing over 40 hours of diverse pedestrian motion from 100 subjects. Critically, all data was captured natively at 200 Hz, establishing the high-frequency baseline that this work investigates.

RoNIN's architectural success is attributed to two key design principles that directly address the challenges of inertial navigation:

- **Coordinate Frame Normalization:** To make the model invariant to the phone's heading, RoNIN normalizes all input IMU data into a *heading-agnostic coordinate frame* (HACF). In this frame, the Z-axis always points in the direction of gravity, but the rotation around this axis can be arbitrary. This normalization ensures that the model learns motion patterns independent of how a user is holding or carrying their device, which is important for real-world robustness.
- **Robust Velocity Losses:** Unlike the noisy, instantaneous velocity vectors generated by the ground truth, RoNIN uses more stable loss functions. For its recurrent architectures—LSTM networks and TCN—RoNIN applies a latent velocity loss to ensure the integral of the predicted velocities over a window relates to the positional change. RoNIN also uses a strided velocity loss with its ResNet, predicting change in position over a full one-second interval.

These features, combined with powerful backbones like LSTMs and TCNs, make RoNIN a robust baseline. The experiments in this paper are designed to evaluate how this high-performance, 200 Hz-centric architecture responds when its fundamental data assumptions are challenged by significantly lower sampling rates.

Modern IMU-only localisation pipelines increasingly emphasize signal conditioning and temporal alignment to ensure data quality before integration into neural or classical models.

Upsampling & Smoothing: IMU data streams are occasionally upsampled (for example, to reach a model's fixed input sample rate or to synchronize sensors with heterogeneous sampling frequencies). The most basic form of upsampling is linear/spline interpolation, which is the approach taken by [5], where spline interpolation was used to match a 1.6 kHz gyro for their high speed testing trials. Chen *et al.* used dual-linear interpolation for 400 Hz IMU data to have a time stamp derived from the 30 Hz camera frame. More complex upsampling/smoothing stages include a number of predictive filters, such as a Rauch–Tung–Striebel (RTS) smoother, which performs the backward pass of an EKF to take advantage of system dynamics and infer intermediate states of the system [11]. Considering interpolation as a filtering problem allows the algorithm to use both past and future observations, which will give better performance than naive interpolation methods. Generally, a zero-phase finite impulse response (FIR) is used beforehand to avoid increasing noise in the interpolation. The implementation built in this research follows the best practices: Filtering with FIR is prior to downsampling, and upsampling is done using Kalman-based smoothing with FIR interpolation. This approach ensures that the inputs to subsequent odometry modules remain band-limited and free from excessive noise.

Downsampling must be preceded by an antialias low-pass filter to avoid spectral folding [10]. The frequency spectrum of human motion IMU signals is generally concentrated at low frequencies. More than 95% of the signal energy associated with accelerations of the walking pattern is concentrated below 10–15 Hz [6]. The components above that range are usually either sensor noise or incidental vibrations. Consequently, aggressive filtering and downsampling of high-frequency components result in minimal information loss while reducing computational requirements. Fan *et al.* showed that there is a decreasing return for the improvements in orientation accuracy found at higher sample rates: walking was adequate at 100 Hz, running was adequate at around 200 Hz, and only very fast cyclic actions had improvements in orientation accuracy up to around ~400 Hz [5]. Increasing the sampling rate beyond 100 Hz did not yield further improvements in accuracy, likely because the additional samples introduced more noise into the measurements[5]. Lower rates therefore save power and wireless bandwidth in wearables [5], echoing other findings that low-bandwidth motion can be reliably captured once anti-alias filtering is applied [6, 10].

Timestamp Alignment: Precise alignment of time stamps is critical, since time offsets as small as one millisecond can degenerate abruptly after integration. Pipelines typically either resample one of the streams onto another via interpolation [2] or estimate offsets directly from spatiotemporal calibration using the sensor configuration of the IMUs [2]. In most consumer grade systems, simple interpolation is appropriate: sensor readings are shifted or interpolated to the nearest timestamp in common, providing a consistent and sensible input into the fusion algorithm. Thus, the IMU preprocessing workflow involves three steps: (i) low-pass filtered and downsampled, (ii) upsampled/smoothed on occasion, and (iii) absolute timestamp integrity maintained. That step uses known spectral analyses and studies of sampling rates [5, 6] to minimize systematic errors and random noise drift, and thus provides a clean representation of the IMU readings intended for processing in the IMU-only localisation methods described previously.

4 METHODOLOGY

This section describes our approach to study energy-efficient IMU-only localisation methods focused on both low-rate sensing with temporal upsampling and the implications of low sampling rates for neural inertial odometry. The approach had two experimental routes. The first involves determining if temporally upsampled low-rate IMU data could adequately run a high-rate RoNIN model without modifications. The second focuses on determining the performance of RoNIN models directly retrained only on downsampled IMU data across all different rates. Figure 2 provides a visual overview of these two pipelines. The research method involves: data collection, data preprocessing, implementation and comparison of upsampling methods, inference and retraining of the RoNIN model, and assessing trajectory accuracy and efficiency.

4.1 Data Collection

iOS IMU Dataset (Utrecht 2025). Two identical Apple iPhones, taped together, simultaneously recorded pedestrian motion data during an urban walk in Utrecht, the Netherlands. One iPhone recorded triaxial accelerometer and gyroscope data at 40 Hz using the SensorLog app. The other recorded at 100 Hz, enabling direct comparison and subsequent upsampling of the 40 Hz data to 100 Hz for validation. This dual-device setup ensured that both low- and high-frequency datasets captured the same motion events in parallel, allowing for accurate assessment of upsampling techniques and their impact on localisation accuracy.

Android IMU Dataset. The research used the official RoNIN Android application, following the procedures described in the original RoNIN publication [9]. All recordings were subsequently preprocessed to conform to the data format specified by the RoNIN framework. For further details regarding the data collection protocol, refer to [9].

4.2 Data Preprocessing

Parsing and Formatting. To manage the two data sources we implemented specialized loaders. One loader parses the single-table CSV format from the iOS SensorLog app, while another aligns the multiple raw sensor streams produced by the Android RoNIN framework.

- **SensorLogSequence (iOS).** The raw SensorLog CSV exports use long header names and store user–acceleration, gravity, quaternion, and GPS samples in a *single* table.
  (1) Detects the timestamp column, converts nanoseconds to seconds when necessary.
  (2) Renames SensorLog headers to concise labels (e.g. `motionUserAccelerationX(G)`$\longrightarrow$ `accX`).
  (3) Adds the gravity vector back to motionUserAcceleration to reconstruct the raw accelerometer signal and then convert the result from multiples of g to SI units (m/s^2).
  (4) Projects GPS latitude/longitude to a local tangent Mercator frame using `Pyproj` [12].
  (5) Rotates gyroscope and accelerometer vectors into a global gravity-aligned frame via device quaternions; gravity sign is flipped so that $+z$ is up.
  (6) Outputs: timestamps, $[\,\dot{\omega}_x, \dot{\omega}_y, \dot{\omega}_z, a_x, a_y, a_z\,]$, ground-truth 2-D velocity from GPS speed and heading, and local $x-y$ position for evaluation.
- **AlignedSensorLogSequence (Android, GPS-adjusted RoNIN).** Adapted from the original RoNIN framework, this loader processes Android IMU and GPS data as follows:
  (1) Aligns raw IMU streams (gyroscope, accelerometer, magnetometer) onto a unified timeline.
  (2) Replaces the Tango-based ground truth with phone GPS data.
  (3) Converts GPS latitude, longitude, and altitude to local East-North-Up (ENU) coordinates for positional ground truth.
  (4) Uses GPS-derived velocity estimates as target values.
  (5) Enables realistic positioning without dependence on specialized hardware like the Tango device.

Both loaders construct a consistent 6-channel IMU feature vector (gyroscope and accelerometer) for each timestamp. After preprocessing, each experiment yields three aligned arrays: *features* ($N\times6$), *targets* (2-D velocity), and auxiliary data (timestamps, orientation quaternions, and ground-truth position).

4.3 Upsampling Techniques

We selected a 40 Hz sampling rate to emulate a realistic low-power configuration, as this is a common lower bound for reliable human motion capture. Additionally, we tested a 100 Hz rate to examine an intermediate, moderate-power setting that still challenges neural inertial odometry models.

We tested three interpolation strategies to reconstruct high-frequency IMU streams from the original data:

- **Linear Interpolation**: Missing samples were interpolated by piecewise linear interpolation between original 40Hz data points using the function `interp1d` in the SciPy package [13].

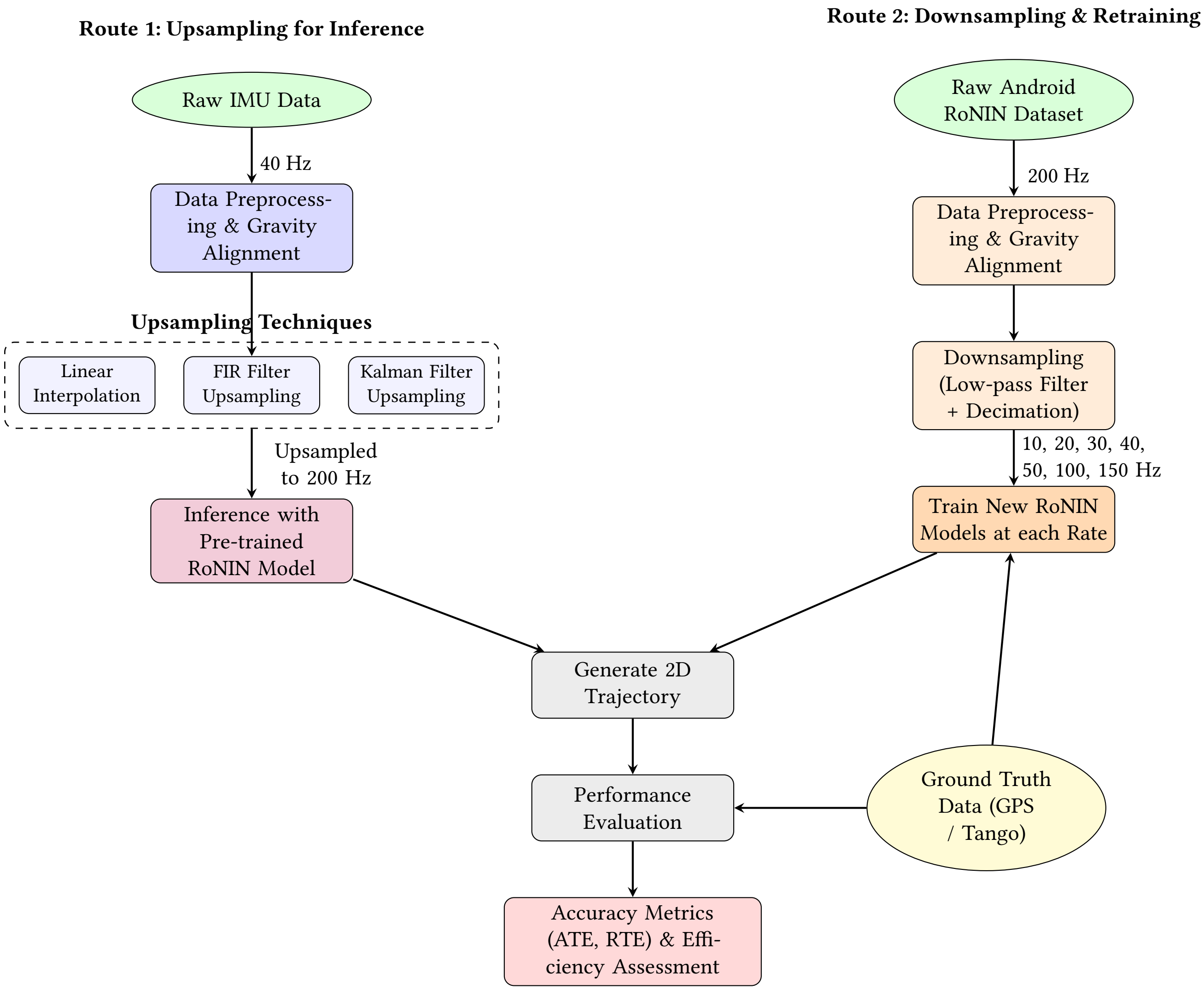

Figure 2. Diagram of the IMU-based localisation methodology. It shows the two experimental routes: (i) upsampling low-rate iOS data to run on a pre-trained high-rate RoNIN model, and (ii) downsampling high-rate Android data to retrain and evaluate new RoNIN models at various sampling rates. Both pipelines result in a 2D trajectory which is evaluated against ground truth data.

- **FIR Filter-Based Upsampling**: Zero-stuffing was employed by inserting zeros between samples to perform upsampling and then applying a polyphase FIR low-pass filter designed using the `firwin` and `upfirdn` functions in the SciPy package [13]. The cutoff frequency was set to 20Hz producing a smooth band-limited signal.
- **Kalman Filter-Based Upsampling**: A discrete four-state Kalman filter with Rauch–Tung–Striebel (RTS) smoothing [11] was created per signal axis. The filter state contained position, velocity, acceleration and bias, with the filter predicting intermediate points at the higher sampling rate (0.01s steps for 100Hz, and 0.005s for 200Hz) and being updated every 0.025s based on the actual (observed) 40Hz measurements.

4.4 Model Inference and Evaluation

The pre-trained RoNIN ResNet model, originally trained on 200Hz data, was used for predictions. For evaluation on lower-rate data (e.g., 40Hz or 100Hz), first we temporally upsampled input sequences to 200Hz using the corresponding interpolation or filtering technique, ensuring compatibility with the model's expected input rate. Windowing was then performed to match the original model configuration (e.g., 400 samples per 2 seconds window), and model outputs were integrated to generate 2D trajectories.

4.5 Downsampling

This part of our work assesses how the IMU sampling rate affects inertial odometry performance. Our methodology is based on a downsampling analysis of the well-known RoNIN inertial dataset [9]. The first phase of this analysis involves recreating the original RoNIN training to arrive at a high-frequency baseline of 200Hz. Following a spectral analysis of the IMU signals in the RoNIN dataset [9], the process involved designing an appropriate lowpass filter and implementing a decimation scheme to generate multiple datasets at reduced sampling frequencies (10Hz, 20Hz, 30Hz, 40Hz, 50Hz, 100Hz, and 150Hz). The overall objective of the analysis is to identify the lowest sampling rate to which IMUs can be downsampled, in which the model could still maintain an overall acceptable performance before its performance rapidly decreased.

Downsampling Procedure: Each trajectory underwent the standard RoNIN preprocessing procedure, this involved calibrating out sensor biases, transforming raw accelerations to linear acceleration by removing gravity, and rotating all IMU measurements to the global (gravity-aligned) frame of reference. Following this preprocessing, down-sampling occurres. An FIR low-pass filter with zero phase lag, designed using a Hamming window, was applied first using the `firwin` and `filtfilt` functions from the SciPy package [13]. Decimation was then performed by selecting every *n*th sample from the filtered high-rate signal to obtain the desired lower frequency. The cutoff frequency was set to 15 Hz, based on the results of the spectral analysis of human motion, which was done with the RoNIN dataset, showing that over 95% of the signal energy is concentrated below 15 Hz. This choice ensures that as much motion information as possible is retained, while effectively reducing aliasing. These same procedures represent a substantial improvement over what occurred in the RoNIN study which only relied on the smoothing properties of the device.

Model Training at Lower Rates: We trained individual RoNIN models from scratch at each downsampled rate, namely, 10, 20, 30, 40, 50, 100, 150 Hz. The original RoNIN ResNet-model architecture and training procedure remained unchanged, with only the input sequence length adjusted to match the new sampling rate. Thus, a fixed-duration (e.g., 2-second) input window at 200 Hz contained 400 samples, whereas at 40 Hz, it contained only 80 samples. Each model was trained independently, beginning from random weight initialisation, thus eliminating any potential bias from the original pre-trained model. This provided an unbiased measure of how well the model could adapt specifically to low-rate IMU inputs.

Performance Evaluation: To assess the trajectory precision for each model trained on downsampled data, we measure Absolute Trajectory Error (ATE) and Relative Trajectory Error (RTE), following the evaluation protocol established in the original RoNIN benchmark [9]. ATE measures the overall drift in position over the entire trajectory, while RTE measures the drift average over shorter segments of the trajectory. Evaluation relies on ground truth trajectories from the RoNIN dataset, originally recorded with a high-accuracy Tango device. To assess horizontal errors in the trajectories, the ground truth trajectories were rendered flat on a 2D plane.

Efficient downsampling was accomplished using signal processing methods. Retraining at each sampling rate enabled an explicit and unbiased evaluation of the trade-offs between IMU sampling rate and inertial odometry accuracy. The next section summarises all the quantitative results from the experiments described in this paper.

5 RESULTS

5.1 RQ1: Effect of IMU Sampling Rate on RoNIN Performance

Controlled experiments showed the influence of different IMU sampling rates on inertial-odometry drift and accuracy. Two model types were evaluated:

(a) the original *pre-trained* RoNIN model (trained on the full RoNIN dataset at 200 Hz);
(b) four RoNIN models re-trained from scratch on IMU data downsampled to 150, 100, 50, and 40 Hz.

All models share the same RoNIN-ResNet backbone and differ only in input sampling rate. The RoNIN dataset (IMU and ground-truth trajectories logged at 200 Hz) served as the source of training and testing data [9].

Table 2 summarises average ATE and RTE on the *seen* and *unseen* RoNIN test sequences. Lower IMU rates, such as 30 Hz, 20 Hz, and 10 Hz, consistently produced larger drift. The 200 Hz pre-trained model achieved the lowest errors, with a median drift of less than 5 meters after several minutes of walking and the 40 Hz model showing more than double the drift on unseen sequences; intermediate rates fell between these extremes. A more detailed breakdown and discussion of model retraining at additional sampling rates (30, 20, 10 Hz) is provided in Section 5.3.

Answer to RQ1. On the RoNIN benchmark (4–10 minutes sequences), only the native 200 Hz configuration kept drift within meter-level bounds (approximately 3 m ATE, 2.5 m 1-minute RTE). A modest reduction to 150 Hz already doubled the accumulated error to roughly 6 m, while further reductions—100 Hz (about 6.5 m), 50 Hz (about 7 m), 40 Hz (about 6.4 m), and 30 Hz or below (8–9 m)—pushed the drift beyond the 5 m target commonly regarded as acceptable for indoor pedestrian tracking. Consequently, maintaining drift within 5 m with only a single start reference proved achievable only at 200 Hz, if a more relaxed 10 m ceiling is acceptable, 40 Hz serves as a practical lower bound, with slower rates resulting in rapidly increasing drift.

5.2 RQ2: Temporal Super-Resolution (Upsampling Low-Rate IMU to 200 Hz)

To address RQ2, we upsampled low-rate IMU sequences (40 Hz and 100 Hz recordings form the walk) to 200 Hz using three approaches: simple linear interpolation, zero-stuffing with FIR low-pass filtering, and Kalman Rauch–Tung–Striebel (RTS) smoothing. These 200 Hz reconstructions were then fed into the original pre-trained RoNIN model (which expects 200 Hz input) without any retraining. Table 1 summarises the results of the localisation drift.

The experimental findings clearly demonstrated that none of the three upsampling methods achieved the high-rate accuracy desired by RoNIN. The drift noted was well over the target accuracy level. In the case of the 40 Hz sequences upsampled to 200 Hz, the ATE was between 199 m (linear interpolation) and 215 m (Kalman smoothing), while the ATE for the original 200 Hz data recorded along the same trajectory was approximately 3.24 m ATE. The 100 Hz sequences were marginally better, and had an ATE around 175 m ATE in the

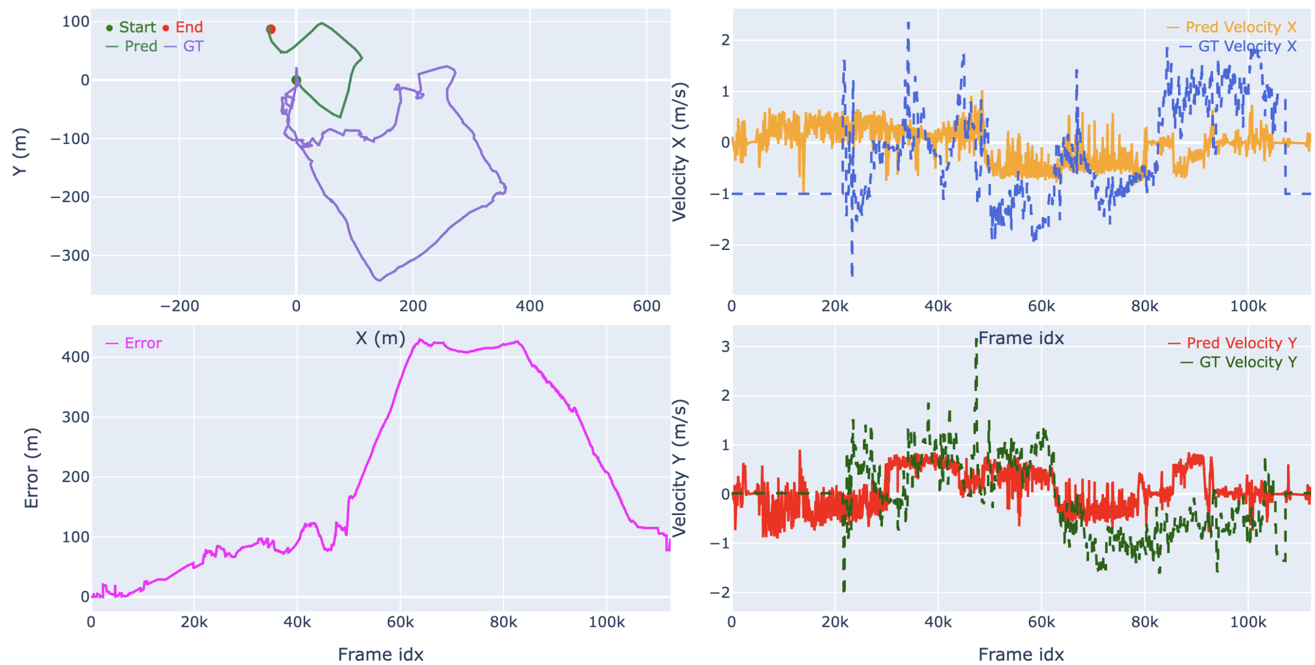

Figure 3. Example of acceleration data upsampled from 40 Hz to 200 Hz using FIR.

upsampling phase, but still had performance well below the 200 Hz native data.

Table 1. RQ2 localisation accuracy after upsampling low-rate IMU to 200 Hz and feeding the unchanged RoNIN model. Lower is better.

Input	**Upsampler**	**ATE [m]**	**RTE [m]**
40 Hz ⟶ 200 Hz	FIR	203.4	70.2
	Kalman	215.1	71.0
	Linear	199.0	69.8
100 Hz ⟶ 200 Hz	FIR	175.4	68.8
	Kalman	177.0	69.2
	Linear	173.6	68.6
Native 200 Hz (baseline)	–	**3.24**	**2.45**
40 Hz (direct, no upsampling)	–	160.32	138.40

The results also show negligible differences in final accuracy between the three interpolation methods. All interpolation techniques have similar trajectory errors within a few meters, suggesting that the primary limitation comes from the fundamental lack of high-frequency information in the lower-rate IMU data, rather than the interpolation method itself. Moreover, the relative trajectory error (RTE) seems to have a small positive effect from data upsampling in the short-term splits—from around 138 m (40 Hz direct) to around 70 m. Although, the short-term smoothing effect was not enough to address the large, long-term extent of drift. The trajectories from the upsampled data also had significant differences from the actual path taken, see Figure 3.

These results demonstrate that temporal upsampling has important limitations when the task is to retrieve lost high-frequency inertial information. Once IMU data is undersampled, high-frequency detail of a movement is lost irretrievably and to which interpolation cannot authentically recover. Even the more sophisticated Kalman smoothing, utilizing predictive motion models are unable to produce authentic high-frequency events.

In terms of computational feasibility, all interpolation methods tested required a moderate amount of processing time, with linear interpolation being the fastest, FIR filtering moderately demanding, and Kalman-RTS smoothing the most computationally intensive. However, since none of these methods improved localisation accuracy after upsampling low-rate IMU data, the additional computation did not provide any practical benefit.

Given these negative outcomes, further investigation into RQ3 is justified, specifically exploring the potential of training inertial odometry models directly on low-frequency IMU data, rather than attempting to artificially enhance the data via temporal super resolution.

5.3 RQ3: Low-Rate Model Retraining (Downsampled IMU Data 10–150 Hz)

To address RQ3, the RoNIN inertial odometry model was retrained on progressively downsampled versions of the original 200Hz dataset, specifically at IMU rates of 150Hz, 100Hz, 50Hz, 40Hz, 30Hz, 20Hz, and 10Hz. Each low-rate model underwent evaluation on two test sets: (i) the *seen* set from the RoNIN distribution, and (ii) the *unseen* set consisting of RoNIN unseen sequences not included

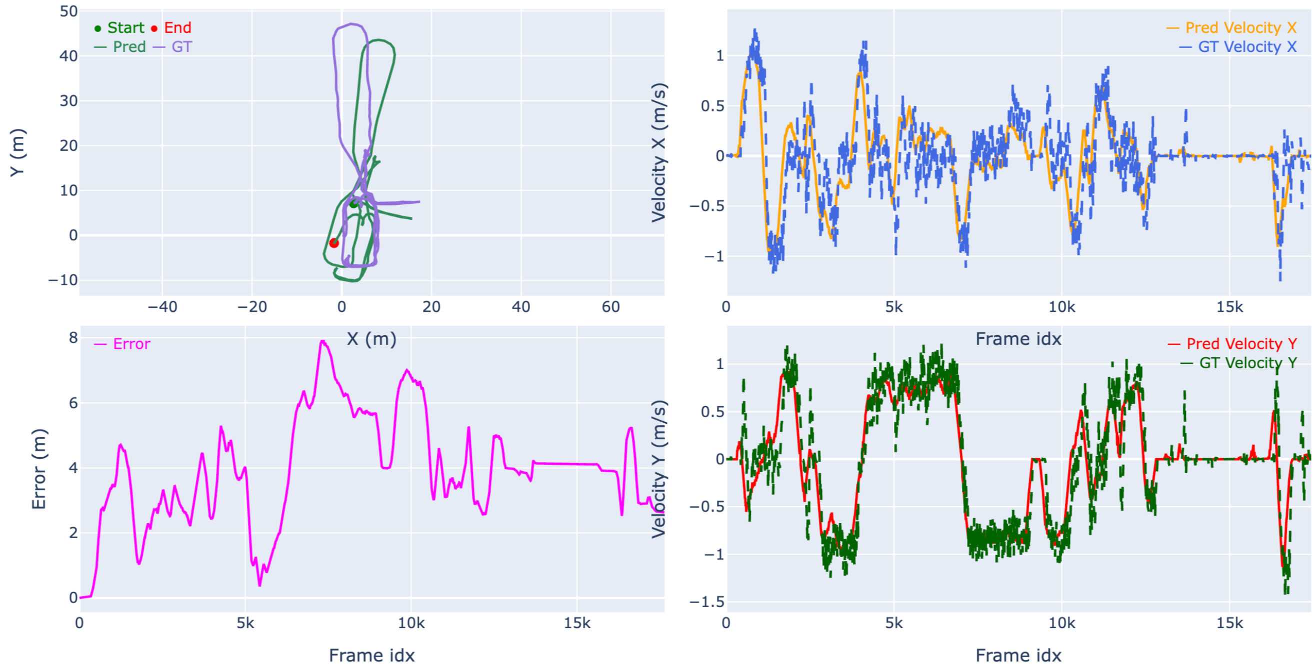

Figure 4. Trajectory and velocity predictions for an unseen test sequence at 40 Hz.

in the training set. The *seen* test assesses performance in the training data distribution and the *unseen* test assesses generalisation.

Table 2 summarises the localisation accuracy for each model in terms of Absolute Trajectory Error (ATE) and Relative Trajectory Error (RTE). The original 200Hz RoNIN model serves as a baseline, achieving approximately 3m ATE on unseen data [9].

Table 2. Performance of RoNIN-style models retrained on downsampled IMU rates. Lower is better for all metrics.

IMU Rate	ATE seen [m]	RTE seen [m]	ATE unseen [m]	RTE unseen [m]
10 Hz	6.97	55.71	8.96	40.91
20 Hz	6.65	16.44	8.06	18.20
30 Hz	5.40	9.32	8.23	13.58
40 Hz	**4.87**	**5.79**	**6.41**	**8.30**
50 Hz	5.33	3.48	7.01	4.90
100 Hz	5.59	4.26	6.52	5.31
150 Hz	5.49	3.31	6.22	4.86

Despite retraining, all lower-rate models exhibited increased drift compared to the 200Hz baseline. On the seen test set, ATE increased significantly from approximately 3m at 200Hz to 5–7m at lower sampling rates. The unseen test set showed a more pronounced performance degradation, with drift at 10Hz reaching nearly 9m ATE and 41 m RTE, reflecting severe degradation at extremely low rates.

Models assessed at intermediate rates showed relatively moderate performance loss, with ATE ranging approximately between 6–7 m. Interestingly, the model assessed at 40 Hz still performed reasonably (6.41 m ATE), although this was still double the baseline drift at 200 Hz. The models assessed at 10, 20, and 30 Hz displayed progressively worse performance with critical thresholds around 40 Hz where accuracy drops sharply.

Figure 4 shows a typical trajectory from the unseen test set using the 40 Hz downsampled model. The predicted path has moderate drift and moves away from the ground truth, highlighting the difficulty of accurate long-term predictions at this rate. The predicted velocity components (Velocity X/Y (m/s)) are relatively accurate over short periods, indicating that the model can still capture quick motion changes at lower rates. However, even with good short-term accuracy, errors add up over time and lead to noticeable drift in longer sequences.

It is probable that minor inconsistencies across sampling rates, such as small relative shifts in performance for medium frequencies are likely caused by stochasticity of training/training on particular sequences rather than a benefit associated with any specific sampling rate.

The drop in accuracy continues even though most human motion IMU signals are below 15 Hz. Several factors may explain the drop:

(1) **Information loss and aliasing**: Without perfect anti-alias filtering, downsampling can remove or distort important high-frequency details.

(2) **Model temporal resolution**: Higher sampling rates make it easier for the model to produce stable and less noisy velocity estimates.
(3) **Training generalisation**: Models trained on lower-rate data may become too used to smoother signals and struggle to handle sudden movements in real-world situations.

In summary, retraining directly on downsampled data proved more effective than naive upsampling (RQ2), with lower-rate models still able to maintain reasonable pedestrian localisation accuracy in many scenarios.

6 FUTURE WORK

This study demonstrates a clear and difficult trade-off between the IMU sampling rate and pedestrian inertial odometry accuracy, showing that currently ubiquitous state-of-the-art architectures, like RoNIN [9], have great difficulty achieving meter-level precision below a 150–200 Hz sampling rate. While retraining on downsampled data is an enormous gain from simply naive upsampling, a meaningful accuracy gap still remains. Future work should shift from just trying to mitigate information loss, but rather developing systems that are naturally more robust against sparse observations. This section suggests two research directions that could help close the current gap between sampling rate and accuracy.

First, the paradigm of constant-rate sampling should be challenged in favour of more intelligent, energy-efficient sensing strategies. This work treated sampling rate as a static parameter, yet human motion is highly dynamic. A promising way is adaptive or event-driven IMU sampling. A device could operate in an ultra-low-power state (e.g., 10–20 Hz) to monitor for motion, changing to a higher rate (e.g., 100 Hz) only upon detecting significant dynamic events like the initiation of walking or turning. This would detect when high-resolution data is needed to minimize drift, while conserving energy during static or low-dynamic periods. This approach requires co-design of the hardware, firmware, and the odometry model, which must be robust to variable and non-uniform input data rates.

Second, future work should focus on creating models for different application needs and on combining data from multiple types of sensors. This research found that 40 Hz is a practical lower limit for applications that can tolerate about 10 meters of drift. Instead of aiming for a single model that works for all cases, it is more effective to develop several models, each tailored to different accuracy requirements. A 30 Hz model is suitable for casual activity tracking, while first responders require a 200 Hz model for navigation. To improve results at lower rates, it is important to combine IMU data with information from other low-power sensors. For example, adding occasional updates from barometers (for altitude), magnetometers (for heading), or Wi-Fi/BLE signals can help correct drift that builds up with low-frequency IMU data. These steps are essential for achieving accurate, reliable localisation on future mobile and wearable devices with limited power.

7 CONCLUSION

This paper set out to identify the *lowest* IMU sampling rate that can support accurate pedestrian inertial odometry, and to test two broad compensation strategies: temporal upsampling (RQ2) and native low-rate retraining (RQ3). Controlled experiments on the RoNIN benchmark yielded three clear takeaways.

(i) RQ1 – How low can we go? With only a single start reference, drift rises steeply as the sampling rate falls. Native 200 Hz remains the only setting that consistently keeps drift within the tight 5m bound (approximately 3m ATE, 2.5m 1-minutes RTE on unseen sequences). A modest reduction to 150Hz already doubles the error (about 6m ATE), and every further reduction—100Hz (about 6.5m), 50Hz (about 7m), 40Hz (about 6.4m), 30Hz and below (8–9 m)—pushes the trajectory error beyond the 5 m threshold commonly deemed acceptable for indoor tracking. If the application tolerates a more relaxed 10 m ceiling, the results suggest that 40 Hz is a practical lower bound, but anything slower leads to rapidly escalating drift.

(ii) RQ2 – Temporal super-resolution does not *help.* Upsampling 40–100 Hz IMU data to 200 Hz using linear, FIR, or Kalman–RTS interpolation did not improve results: ATE increased to over 170 m. This shows that once important information is lost due to aggressive downsampling, it cannot be recovered by simply increasing the sample rate afterward.

(iii) RQ3 – Native low-rate training helps, but not enough. Retraining RoNIN *from scratch* at each target rate (10–150 Hz) outperformed naive upsampling, yet still fell far short of the 200 Hz baseline. The best low-rate models (40–50 Hz) averaged 6–7 m ATE on unseen data—roughly double the error at full rate—while 20 Hz and 10 Hz deteriorated to 8 m + drift and severe 1-minute RTE spikes. In short, energy savings below 40 Hz come at a sharp accuracy cost with the current architecture.

ACKNOWLEDGMENTS

I used artificial intelligence tools (like OpenAI models) to help improve the grammar, wording, and clarity of this paper. AI also helped with technical issues and code debugging. All ideas, experiments, and conclusions are my own.

In addition, sincere thanks to Undagrid team for their support and encouragement during this project. Special thanks to Rob Kers, Martijn Hilders, and Tsung-Huan Wu for their guidance, feedback, and technical help, which were essential for this paper.

REFERENCES

[1] M. Brossard, A. Barrau, and S. Bonnabel. 2020. AI-IMU Dead-Reckoning. *IEEE Transactions on Intelligent Vehicles* 5, 2 (2020), 274–287. https://doi.org/10.48550/arXiv.1904.06064
[2] T. Chen and et al. 2022. WHUViD: A Stereo-IMU Dataset for Visual-Inertial Odometry. *Remote Sensing* 14, 7 (2022), 1691. https://doi.org/10.3390/rs14092033
[3] Y. Chen, X. Zou, Y. Zhang, S. Wang, C. Xie, Q. Huang, and Y. Wang. 2018. IONet: Learning to Cure the Curse of Drift in Inertial Odometry. In *Proceedings of the AAAI Conference on Artificial Intelligence.* https://doi.org/10.1609/aaai.v32i1.12102
[4] S. S. Saha et al. 2022. TinyOdom: Hardware-Aware Efficient Neural Inertial Navigation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)* 6, 2 (2022), 71:1–71:32. https://doi.org/10.1145/3534594
[5] B. Fan and et al. 2025. Influence of Sampling Rate on Wearable IMU Orientation Accuracy. *Sensors* 25, 7 (2025), 1976. https://doi.org/10.3390/s25071976
[6] S. Fioranelli and et al. 2024. In-sensor Human Gait Analysis with Machine Learning. *Communications Engineering* 1, 1 (2024). https://doi.org/10.1038/s44172-024-00193-5
[7] Andreas Geiger, Philip Lenz, and Raquel Urtasun. 2012. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 3354–3361. https://doi.org/10.1109/CVPR.2012.6248074

[8] R. Harle. 2013. A Survey of Indoor Inertial Positioning Systems for Pedestrians. *IEEE Communications Surveys & Tutorials* 15, 3 (2013), 1281–1293. https://doi.org/10.1109/SURV.2012.121912.00075

[9] S. Herath, H. Yan, and Y. Furukawa. 2020. RoNIN: Robust Neural Inertial Navigation in the Wild. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. https://doi.org/10.48550/arXiv.1905.12853

[10] Hexagon | NovAtel. 2020. *Inertial Navigation Systems and Vibration – Application Note APN-112*. Technical Report. Hexagon | NovAtel.

[11] H. E. Rauch, F. Tung, and C. T. Striebel. 1965. Maximum-Likelihood Estimates of Linear Dynamic Systems. *AIAA Journal* 3, 8 (1965), 1445–1450. https://doi.org/10.2514/3.3166

[12] Howard Snow et al. 2024. *Pyproj: Python interface to PROJ (cartographic projections and coordinate transformations library)*. https://pyproj4.github.io/pyproj/ Version 3.7.1.

[13] P. Virtanen and et al. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* 17, 3 (2020), 261–272. https://doi.org/10.1038/s41592-019-0686-2

[14] H. Yan, Q. Shan, and Y. Furukawa. 2018. RIDI: Robust IMU Double Integration. In *Proceedings of the European Conference on Computer Vision (ECCV)*. https://doi.org/10.48550/arXiv.1712.09004