

Tartan IMU: A Light Foundation Model for Inertial Positioning in Robotics

Shibo Zhao^{1†*}, Sifan Zhou^{1†}, Raphael Blanchard¹, Yuheng Qiu¹, Wenshan Wang¹, Sebastian Scherer¹

¹Carnegie Mellon University

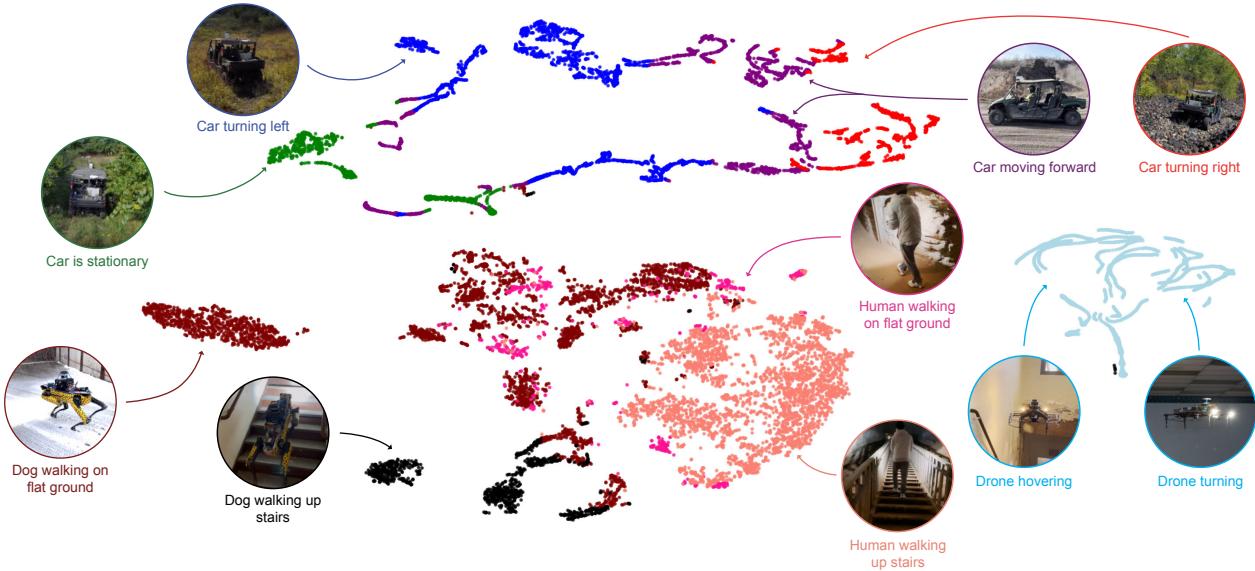


Fig. 1. t-SNE Visualization of our Tartan IMU Foundation Model. Our foundation model is trained on over **100 hours** of diverse IMU data collected from various platforms, including ground vehicles, drones, legged robots, and human motion. This model maps IMU data into a unified high-dimensional shared space, capturing broad, generalizable IMU knowledge.

Abstract

Despite recent advances in deep learning, most existing learning IMU odometry methods are trained on specific datasets, lack generalization, and are prone to overfitting, which limits their real-world application. To address these challenges, we present *Tartan IMU*, a foundation model designed for generalizable, IMU-based state estimation across diverse robotic platforms. Our approach consists of three-stage: First, a pre-trained foundation model leverages over 100 hours of multi-platform data to establish general motion knowledge, achieving 36% improvement in ATE over specialized models. Second, to adapt to previously unseen tasks, we use Low-Rank Adaptation (LoRA), allowing positive transfer with only 1.1 M trainable parameters. Finally, to support robotics deployment, we introduce online test-time adaptation, which eliminates the boundary between training and testing, allowing the model to continuously "learn as it operates" at 200 FPS in real-time.

Project page: <https://superodometry.com/tartanimu>.

1. Introduction

The Inertial Measurement Unit (IMU) is a critical component in robotic systems, widely used across platforms such as autonomous drones, vehicles, legged robots, and VR/AR devices for human interaction. By measuring linear acceleration and angular velocity, the IMU enables accurate motion tracking and reliable control across diverse environments. Compared to other positioning solutions such as visual odometry[47] and LiDAR odometry[48], Inertial odometry (IO) offers greater robustness in challenging conditions as it is unaffected by visual degradation such as low lighting and LiDAR degradation such as smoke [49].

However, current IO solutions rely on double integration to integrate the pose from acceleration and angular velocity, which is susceptible to drift due to factors such as sensor bias, temperature fluctuations, and sensor white noise [19]. Even minor errors can rapidly accumulate during integration, causing significant drift and unbounded er-

*Corresponding author. †Equal contribution.

rors within seconds [10]. To solve this problem, previous research incorporates domain-specific knowledge such as dead reckoning[10] or other sensor constraints[19]. While these approaches show effectiveness in particular application areas, they are still difficult to address the fundamental drift problem of inertial odometry [10].

Recently, learning-based inertial odometry methods have demonstrated promising performance in addressing challenges of inertial positioning. For example, TLIO [25] has been effective in pedestrian motion estimation, IMO for drone motion estimation [14], AI-IMU [6] for vehicle motion estimation, and deep IMU bias estimation for legged robot motion estimation [7]. However, most of these approaches are still difficult when applied in real-world scenarios. The key challenges can be summarized as follows:

Generalizability: Existing models are tailored to either locomotion-specific or device-specific, which often struggle to generalize in out-of-distribution data. To the best of our knowledge, no current method has been evaluated comprehensively during long-term operation and across various motion patterns and devices.

Adaptability: Existing learning models struggle to adapt to new motion types or new platforms. Deploying learning-based IO across various platforms requires that models quickly adapt to new motion modalities, even with limited data. This adaptability is essential for the practical deployment of learning-based IO in real-world applications.

Drawing from the success of large-scale, high-quality, and diverse data in developing general models that outperform specialized models in natural language processing [26] and visual perception [27], we explore this paradigm in the context of IMU odometry. While this approach has proven effective across many domains, no prior work has applied it to inertial positioning. In fact, IMU data differs fundamentally from language and vision data due to its continuous, dynamic, and noisy nature. Even small changes in motion patterns or devices can create significant domain gaps between the training and testing phases, making IMU generalization particularly challenging.

In this paper, we try to answer a key question: *How to achieve the foundation model for IMU state estimation in robotics?* We define an ideal IMU foundation model should be: i) generalizable positioning across robotic platforms, ii) positive knowledge transfer to new platforms, and iii) fast adaptation for new knowledge. We introduce TartanIMU, an IMU foundation model built on a three-stage pretrain-adapt-test framework: pre-training on a large dataset, fine-tuning on unseen data, and online adaptation for real-world testing (as shown in Figure 1). The main contributions are:

- **IMU Foundation Model:** We introduce the first IMU foundation model, featuring a shared backbone that enables scalable, cross-platform motion estimation, as illustrated in Figure 1. TartanIMU is trained on a diverse

dataset of over 100 hours of data collected from various robotic platforms, including ground vehicles, drones, legged robots, and human motion. As a result, it achieves a notable accuracy improvement of **36%** across all platforms, consistently outperforming expert-tuned models.

- **Efficient Fine-tuning for Unseen Motion:** Leveraging our pre-trained IMU foundation model, it can quickly adapt to new setups with minimal data. To achieve this, we integrated LoRA [21] into our architecture, requiring only **1.1M** parameters to adapt unseen scenes. This approach not only mitigates the problem of catastrophic forgetting but also allows flexible insertion of adapters tailored to different robotic platforms.

- **Online Adaptation:** We propose an online adaptation method that dynamically selects motion patterns for fast training. Unlike conventional learning approaches where the pre-trained model remains static during inference, our method eliminates the boundary between the training and testing phases —*we learn as we operate*. This approach progressively enhances performance and achieves rapid real-time adaptation.

We hope that Tartan IMU represents a step towards such general-purpose IMU foundation models that can serve as a foundation for diverse mobile robotic applications.

2. Related Work

Traditional model-based inertial odometry (IO) methods, such as those using kinematic motion models [19], estimate relative motion through double integration. However, these methods are prone to drift over time due to sensor noise, necessitating external corrections such as visual input [28]. Recent data-driven approaches [12, 29, 46] aim to address this issue by modeling and compensating for noise while quantifying uncertainty in IMU measurements. Nevertheless, error accumulation during double integration remains a significant challenge.

Deep learning has emerged as a promising alternative, driving the development of learning-based IO methods. Early approaches utilized CNNs [11, 24, 50], LSTMs [12, 17, 18, 33, 40, 41], and Transformers [1, 31, 39, 42, 43] to estimate motion. Hybrid methods, such as TLIO [25] and IDOL [35], combined learning-based techniques with kinematic models to enhance robustness and accuracy. Furthermore, learning-based approaches have been applied to tasks like motion classification and human gait estimation [4, 38], with RIO [9] introducing rotation-equivalent augmentation to improve robustness.

Despite these advancements, most learning-based IO methods are tailored for pedestrian navigation, limiting their ability to generalize to out-of-distribution scenarios [7]. Recent research has extended their application to diverse platforms, including legged robots [7, 8], wheeled robots [5, 6, 15], and racing drones [14]. For planar-motion

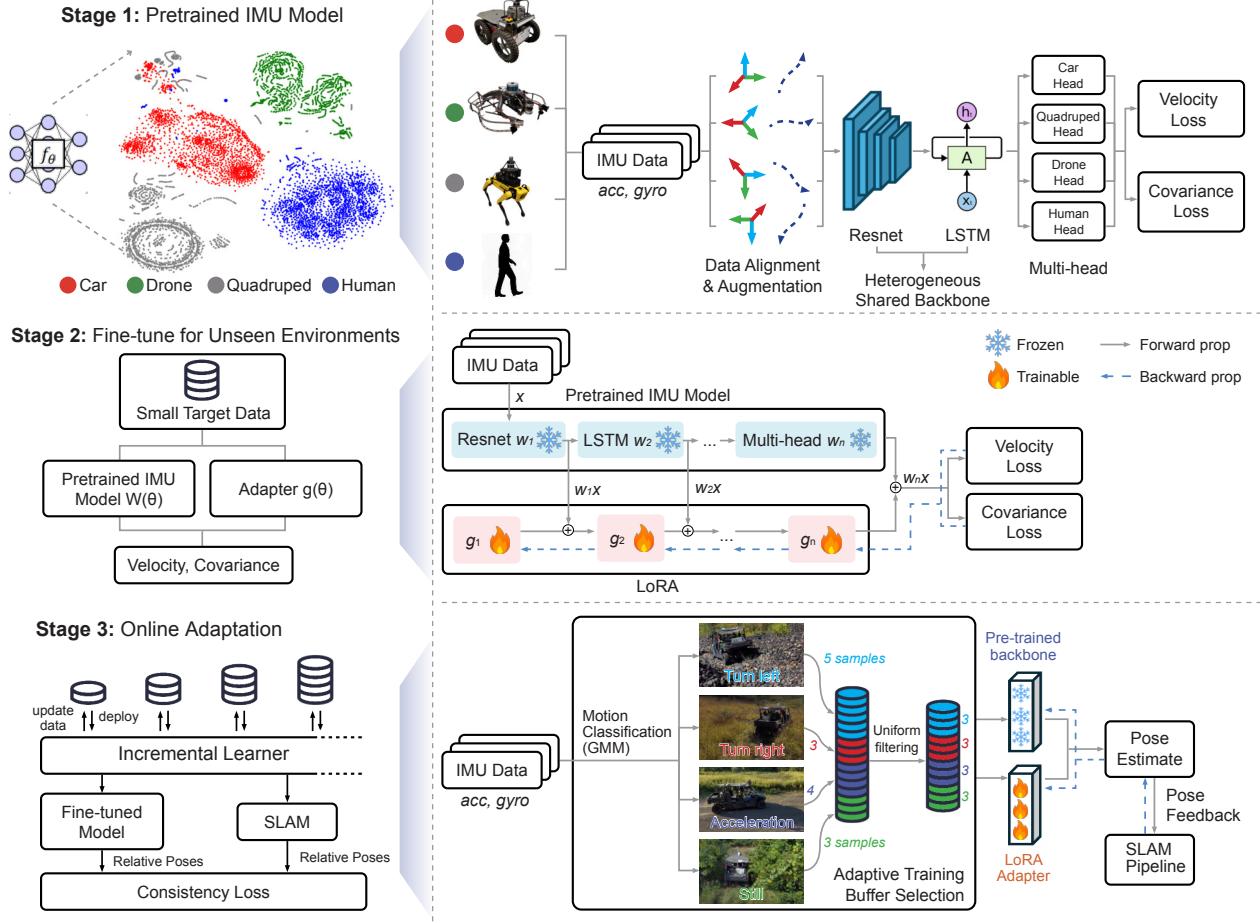


Fig. 2. Three Learning Stages of Tartan IMU a) Pretrained IMU Model: features a shared backbone to capture generalizable IMU knowledge. b) Efficient Fine-Tuning: utilizes an adapter to enable positive transfer for new tasks. c) Online Adaptation: employs an adaptive memory buffer to support on-the-fly model updates during deployment.

systems, such as ground robots and pedestrians, zero-velocity updates (ZUPTs) are often integrated with motion models [37], feature extractors [44], or uncertainty quantification frameworks [6] to mitigate drift. To improve the generalization of learning inertial odometry, EqNIO [22] introduces canonical displacement priors in the network design.

While these efforts have expanded the scope of learning-based IO, there is currently no IMU odometry model that achieves generalization across diverse platforms, adapts seamlessly to unseen environments, and serves as a foundation for diverse IMU-based motion estimation tasks. Challenges related to generalizability, and adaptability remain significant barriers to the broader adoption of learning-based IMU odometry across a wide range of applications. Further details are provided in the supplementary materials.

3. Method

3.1. Pipeline Overview

The framework of the proposed Tartan IMU is illustrated in Figure 2. Tartan IMU comprises three key components:

pre-trained IMU model trained from large-scale datasets, an adapter network *fine-tuning for unseen motion*, and *online test-time adaptation* to further boost the accuracy and robustness of odometry during deployment.

3.2. Pretrained IMU Model

To achieve broad generalization and few-shot learning capabilities, our Tartan IMU model is trained on a diverse array of datasets. The model architecture includes four main components: *data alignment and augmentation*, *a heterogeneous shared IMU backbone*, *a multi-head for different robots*, and *a loss function* shown as Figure 2 stage 1.

Data Alignment The datasets we use involve various IMU sensors with differing axis orientations. To ensure consistency, we standardize the coordinate system to X-forward, Y-left, and Z-up in the IMU body frame [30]. We also normalize the sampling frequency to 200 Hz across all datasets for consistency, balancing temporal resolution with computational efficiency. This setup offers two key benefits: (i) sufficient temporal resolution to accurately capture motion

dynamics; (ii) a consistent input structure that reduces variability from different sensor sampling rates, supporting uniformity in training and inference.

Data Augmentation To enhance model generalization, we apply rotation-equivariance [9] in the network. This method assumes that if the IMU axis undergoes a specific rotation, the predicted and ground truth trajectories should transform correspondingly. This alignment and augmentation process is critical in improving the model’s overall performance.

Heterogeneous Shared Backbone After data alignment and augmentation, we introduce our **foundation IMU encoder** which employed a ResNet-style architecture [20] to extract latent features from diverse IMU inputs. Given the continuous and regular nature of robotic motion, which often follows specific patterns over time, we utilized an LSTM network to capture these temporal dynamics. The LSTM outputs are then passed to a decoder, which regresses velocity and covariance shown in Figure 2, stage 1.

In our network’s data flow, the earlier ResNet layers function as a heterogeneous^{*} shared backbone, mapping diverse IMU data from different robots into a unified high-dimensional representation space. This shared representation allows IMU data from new platforms to require only minimal data and training to adapt to the shared “knowledge.” As a result, this representation space becomes generalizable across diverse robots, capturing broader, world-level knowledge. Our aim is to pre-train foundation models that can map raw IMU signals from various robots into a shared latent space. To showcase that our shared backbone effectively captures motion-specific representations across different robots, we use t-SNE to visualize the learned latent space (see Figure 2, Stage 1). The visualization reveals clear clusters corresponding to four robot types—vehicle (red), drones (green), legged robots (gray), and human (blue)—highlighting the model’s ability to learn a unified representation while preserving the distinctions between different motion patterns.

Multi-head Following the heterogeneous shared backbone, the final component of our network is a robot-decoupled, multi-head design that maps the shared latent representation to the velocity estimation space. During training, different regression heads are assigned to different robot types, benefiting from joint training within the shared backbone, as illustrated in Figure 2, stage 1. This multi-head setup provides two key advantages: (i) it enables the model to learn diverse motion patterns in parallel, enhancing adaptability across different robots during deployment; and (ii) it stabilizes the training process by avoiding competing learning objectives (e.g., drone motion and car motion are significantly different). We adopt two fully connected layers as decoders to regress the velocity of the window and the cor-

^{*}Heterogeneity refers to the varying motion patterns across different robot platforms and different IMU devices.

responding covariance. The mathematical formulation of the network is as follows:

$$\begin{aligned} (\hat{v}, \hat{u}) &= f\left(\left({}^B \mathbf{a}_{n-N}, {}^B \boldsymbol{\omega}_{n-N}\right), \dots, \left({}^B \mathbf{a}_n, {}^B \boldsymbol{\omega}_n\right), \mathbf{h}_{n-N}\right) \\ {}^B \mathbf{a}_n &= {}_W^B \mathbf{R}_n ({}^B \mathbf{R}_n (\mathbf{a} - \mathbf{b}_a) - {}^W \mathbf{g}) \\ {}^B \boldsymbol{\omega}_n &= \boldsymbol{\omega} - \mathbf{b}_g \end{aligned} \quad (1)$$

Here, $f(\cdot)$ represents the neural network function that processes inputs from the IMU sensor. \mathbf{a} denotes acceleration, $\boldsymbol{\omega}$ is angular velocity from the IMU, and \mathbf{h}_{n-N} refers to the hidden state produced by the LSTM at the previous time step. For each IMU measurement, we remove the gravity vector ${}^W \mathbf{g} = [0, 0, 9.8]$ mapped into the IMU body frame. At each time step, the network predicts the current motion based on the hidden state \mathbf{h}_{n-N} and a local window of N samples of acceleration and angular velocity in the body frame \mathbf{B} . The output consists of the estimated relative velocity \hat{v} and associated uncertainties \hat{u} . We also assume gyro bias \mathbf{b}_g and acceleration bias \mathbf{b}_a are known.

Loss Function Our network employs relative motion loss functions for each robot head, ensuring high accuracy in local measurements, as shown in Figure 2 Stage 1. The relative loss function helps the network capture motion dynamics within a single local window, improving the smoothness of the predicted trajectory. To optimize this relative loss, we use the Mean Squared Error (MSE), defined as follows:

$$L_{RL}^{MSE}(\mathbf{v}, \hat{\mathbf{v}}) = \frac{1}{n} \sum_{i=1}^n \left(\sum_{j=m}^{m+M} (v_{j \rightarrow j+1} - \hat{v}_{j \rightarrow j+1}) \right)^2 \quad (2)$$

In contrast to existing works [25][13], we formulate the loss in the body frame, similar to [30] rather than in global coordinates, which allows the model to learn more generalized features instead of simply memorizing fixed global coordinates from training trajectories. Here, $\hat{v}_{j \rightarrow j+1}$ denotes the 3D body velocity output of the network for the j -th window in the body frame, while $v_{j \rightarrow j+1}$ represents the corresponding ground truth in the same frame. The variable n indicates the batch size during training, m the starting window of the sequence, and M the total number of LSTM windows. In practice, we use a 10-window LSTM, where each window spans 1 second of IMU data at 200 Hz, trained with a 1 Hz supervision signal.

Covariance: The covariance estimation follows the methodology outlined in [25]. The network outputs a three-dimensional vector, where each element represents the logarithm of a diagonal element of the covariance matrix Σ .

$$L^{\text{NLL}} = \frac{1}{2} (\mathbf{v} - \hat{\mathbf{v}})^T \Sigma^{-1} (\mathbf{v} - \hat{\mathbf{v}}) + \frac{1}{2} \ln |\Sigma| \quad (3)$$

3.3. Fine-tuning for Unseen Environments

While our pretrained IMU model is designed to be generalizable across various platforms, we cannot guarantee

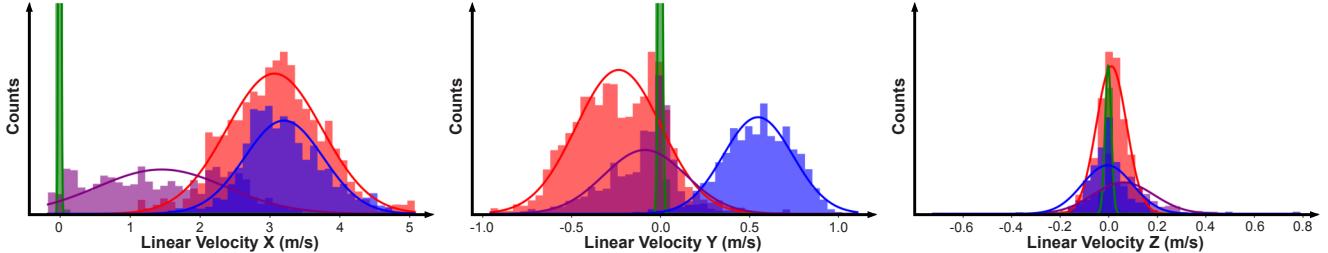


Fig. 3. Distribution of motion patterns classified by GMMs. The histograms with overlaid Gaussian fits show clusters based on linear and angular velocities. The GMM identifies motion types: right turns (red) with negative linear velocity Y, left turns (blue) with positive linear velocity Y, stationary states (green) near zero velocities, and forward motion (purple) with positive linear velocities.

”zero-shot” capability for every robot platform or motion pattern. Additionally, because training efficiency and real-time performance are critical for most robotic state estimation tasks, fine-tuning every parameter of the pretrained model becomes impractical. To address this, we propose adopting a Low-Rank Adapter (LoRA)[21], which freezes the entire pretrained IMU model and only updates a small, zero-initialized adapter network with 5 million parameters (see Figure 2 Stage 2). To update each pretrained weight matrix $W_0 \in \mathbb{R}^{d \times k}$ in Tartan IMU, we apply a low-rank decomposition, representing $W_0 + \Delta W = W_0 + BA$, where $B \in \mathbb{R}^{d \times r}$ and $A \in \mathbb{R}^{r \times k}$, with $r \ll \min(d, k)$. During adaptation, W_0 remains fixed, while A and B are trainable. Both W_0 and $\Delta W = BA$ process the input, and their outputs are summed. This yields:

$$h = W_0x + \Delta Wx = W_0x + BAx \quad (4)$$

Following LoRA [21], A is initialized by a random Gaussian distribution and B at zero, making ΔW start as zero. We then scale ΔWx by $\frac{\alpha}{r}$ for controlled adaptation, where α is a constant in r . This approach preserves the original knowledge from the pretrained model while progressively incorporating new information during adaption.

3.4. Online Adaptation

While LoRA effectively addresses the domain gap between training and testing, it requires pre-collected datasets, which may not be feasible in real-world operations due to the need for rapid adaptation when domain shifts occur. To overcome this limitation, we propose a novel online learning pipeline that enables our model to adapt to new robotic platforms **on-the-fly**. We introduce a ”learning within SLAM” approach, where SLAM acts as a teacher model, providing relative poses as supervision, while our IMU model functions as a student, continuously adapting to reduce the domain gap (see Figure 2 stage 3). Our approach has to overcome two major challenges associated with online adaptation: *i) Limited computation time, ii) Restricted data availability*

Adaptive Training Buffer Selection To address challenges *i) and ii)*, we found that the key lies in having a *diverse and concise dataset* training buffer. We introduce a

dynamic training buffer selection mechanism that preserves diverse samples by focusing on sample similarity. This approach dynamically selects distinct training samples from the input distribution, minimizing overlap and ensuring a balanced dataset without overburdening computational resources. For motion sample similarity comparison, we employ Gaussian Mixture Models (GMMs) [32] to cluster IMU data by distinct motion patterns, as shown in Figure 3. By maintaining a uniform sample distribution across these clusters (see Figure 2, stage 3), the buffer reduces memory load, accelerates online adaptation, and enhances accuracy by learning from diverse motion representations.

4. Experiments

Training Data: We train Tartan IMU on a large-scale, heterogeneous dataset encompassing over 100 hours of real-world IMU data from eight distinct robotic platforms with diverse dynamics. Sourced from prior works such as SubT-MRS [49], IDOL [35], Blackbird [3], and the UZH dataset [16], this dataset includes wheeled robots, drones, legged robots, and human-held devices, providing raw IMU data and ground-truth trajectories. To enhance the model’s generalization capability, we applied *data alignment* and *data augmentation* described in section 3.2.

Experiment Goals: Our approach is designed to achieve *strong generalization, efficient fine-tuning, rapid adaptation, and resistance to forgetting* for IMU foundation model. To validate these, we examine four key questions:

Q1: Does our model generalize across platforms and outperform platform-specific models?

Q2: Can it adapt to new environments through fine-tuning with minimal data?

Q3: Is it capable of real-time adaptation to a new platform with high accuracy?

Q4: Can it mitigate catastrophic forgetting problems?

4.1. Generalization Evaluation

To address **Q1**, we focus on two key aspects for training a more generalized model: (1) *heterogeneous shared backbone*; (2) *large-scale, high-quality, diverse data*.

To illustrate the impact of model architecture, we evaluated various IMU models on data from four motion plat-

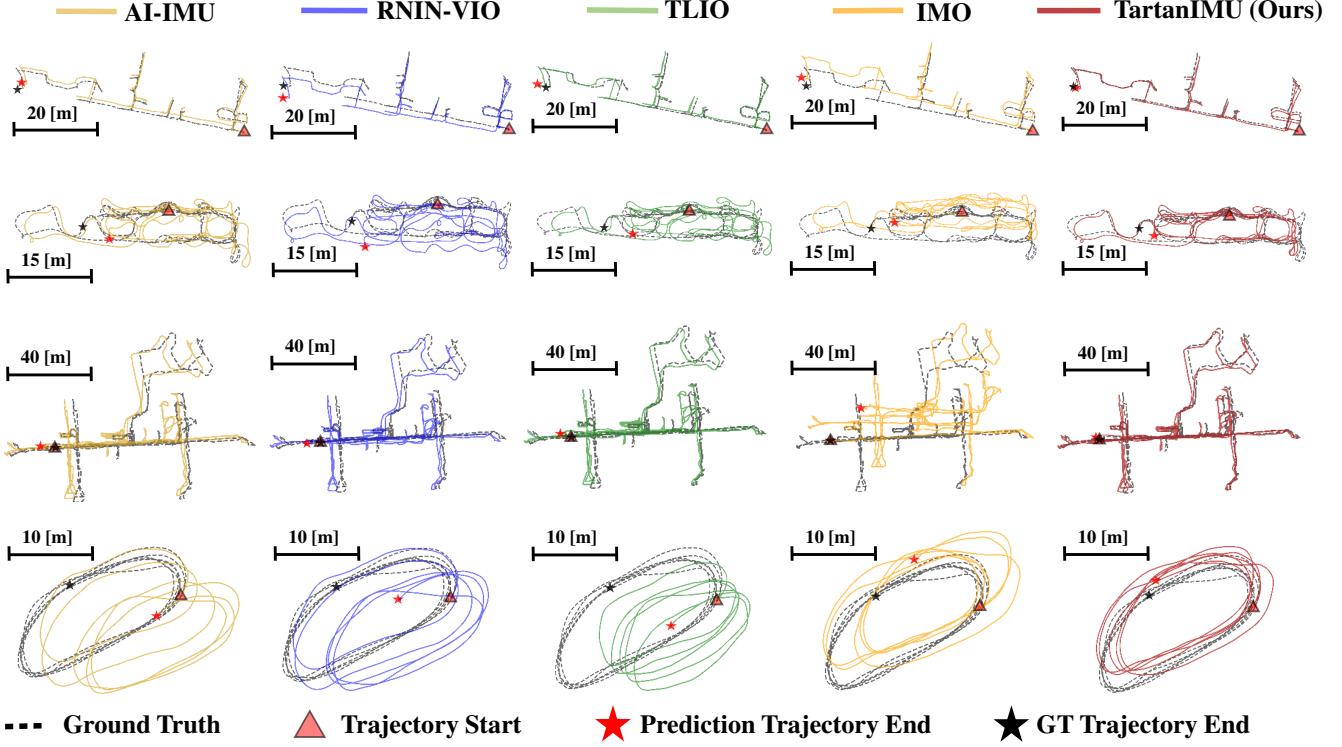


Fig. 4. Qualitative analysis of our proposed TartanIMU compared to other specialized IMU models. Our TartanIMU outperforms different specialized IMU models across various robotic platforms, including wheeled (first row), human (second row), quadruped (third row), and drone (fourth row) systems. Our method provides the best pose estimation results shown in red trajectory.

Robot Platform	AI-IMU [6]		RNIN-VIO [13]		TLIO [25]		IMO [14]		TartanIMU		Improvement	
	ATE ↓	T-RTE ↓	ATE ↓	T-RTE ↓	ATE ↓	T-RTE ↓	ATE ↓	T-RTE ↓	ATE ↓	T-RTE ↓	ATE	T-RTE
Wheeled (Car)[49]	7.68	3.33	7.82	5.06	8.12	3.73	8.12	3.73	6.17	2.52	↑ 19.63 %	↑ 24.36 %
Handled (Human)[35]	8.26	4.89	7.62	5.61	6.96	4.82	10.19	5.67	4.32	1.95	↑ 47.69%	↑ 60.05%
Legged (Dog)[49]	3.23	1.60	3.10	1.58	3.61	1.73	3.35	1.64	1.46	0.79	↑ 54.83%	↑ 50.63%
Aerial (Drone)[2]	4.14	1.45	4.32	1.51	3.93	1.40	3.72	1.34	3.32	1.04	↑ 19.81%	↑ 28.97%

Table 1. Quantitative comparison of pose estimation between the proposed TartanIMU and existing IMU methods [6, 13, 14, 25]. Our TartanIMU outperforms different specialized IMU models, achieving a **35.5%** on ATE and **41.0%** on T-RTE across various robotic platforms including wheeled, handled, legged and aerial systems.

Robot Platform	Tartan IMU					
	ATE ↓		T-RTE ↓			
	Single	All	Improve	Single	All	Improve
Wheeled[49]	7.51	6.17	↑22%	3.17	2.52	↑26%
Handled[35]	6.64	4.32	↑54%	4.24	1.95	↑117%
Legged[49]	3.64	1.46	↑150%	1.94	0.79	↑146%
Aerial[2]	3.84	3.32	↑16%	1.36	1.04	↑30%
Average	5.41	3.82	↑42%	2.68	1.57	↑71%

Table 2. Accuracy comparison of our Tartan IMU pre-trained model trained with data from single-robot vs. all-robots. These results confirm that training in a wide variety of robot data greatly enhances the model’s performance: up to **42%** average improvement on ATE and **71%** average improvement on T-RTE

forms: aerial, wheeled, legged, and handheld. For a fair comparison, all models—AI-IMU [6], RNIN-VIO [13], TLIO [25], and IMO [14]—were trained on the same large

dataset and tested on identical unseen data.

As illustrated in Fig. 4, our IMU pre-trained model, designed with a **heterogeneous shared backbone** network, demonstrated substantial *few-shot* learning capabilities, allowing it to generalize well across different robotic platforms. Without any additional fine-tuning, it was tested on four distinct systems—aerial, wheeled, legged, and handheld—and consistently outperformed models specifically optimized for each of these applications. This model provided more accurate and reliable odometry estimates across a variety of motion patterns. Notably, it achieved an average improvement of **35.5%** in Absolute Trajectory Error (ATE) and **41.0%** in Time-Relative Trajectory Error (T-RTE) over the second-best performing model. These results underscore the effectiveness of the *heterogeneous shared backbone* architecture in achieving robust, platform-

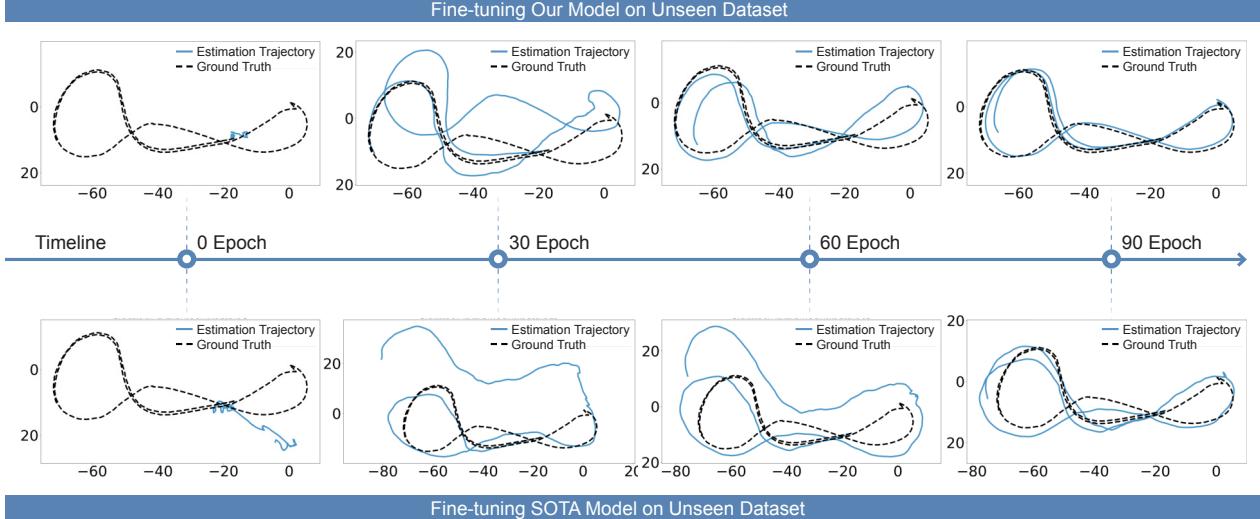


Fig. 5. Comparison of Broader Generalization Results of Our IMU Pre-trained Model with a SOTA Model on Unseen Dataset We fine-tuned both the IMU pre-trained model and the SOTA model [13] from the source domain (SubT dataset [49]) to the target domain (Tartandrive dataset [34]). The quantitative comparison show that our model adapts quicker, requiring 33% fewer iterations compared to the SOTA method.

agnostic generalization across diverse robotic motion types.

We also evaluated the impact of data diversity on model performance. Specifically, we compared the results of our IMU-pretrained model when trained on data from a single platform versus data from various robotic types. As shown in Table 2, models trained on diverse datasets demonstrated substantial improvements. For instance, in the "quadruped" sequence, the model trained on data from aerial, wheeled, legged, and handheld platforms significantly outperformed the model trained solely on legged robot data, with Absolute Trajectory Error (ATE) reduced by up to **150%** and Translational Relative Trajectory Error (T-RTE) improving by **146%**. These results confirm that training on a broad range of robotic data significantly enhances model performance, even when there are notable differences between the training and testing domains. This highlights the critical importance of large-scale, high-quality, and diverse datasets in improving the generalization capabilities of the IMU model (see supplementary video).

4.2. Fine-tuning Evaluation for Unseen Motion

Although our model was trained on a large dataset (see **Q1**), it still encounters generalization challenges when deployed on new robotic platforms with differing motion patterns. To address **Q2**, we employed LORA with minimal data to evaluate our model's adaptability to an unseen platform. Specifically, our model was trained exclusively on the SubT dataset [49] and tested on the unseen TartanDrive dataset [34].

The results, displayed in Figure 5, demonstrate the model's performance on a test sequence excluded from the training dataset. We compared our model against a state-of-the-art (SOTA) baseline model [13], trained on the same

SubT dataset under identical conditions. Over a 60-epoch fine-tuning period (Figure 5), our model achieved significantly more accurate odometry predictions, closely aligning with the ground truth (dashed line). In contrast, while the SOTA model showed some improvement, it retained considerable trajectory errors at the 60-epoch mark.

These findings indicate that our IMU pre-trained model adapts to new domains faster and with **33%** fewer iterations compared to the SOTA method, making it highly effective for real-world applications across diverse robotic platforms and unfamiliar environments (see supplementary video).

4.3. Online Adaptation Evaluation

Q3 poses a difficult challenge, as it involves **continual learning** [23, 36], that requires updating the model effectively to balance prior knowledge with new experience.

In this experiment, our foundation model was initially trained on a UGV operating at a maximum speed of 5 m/s, while testing was conducted on an off-road vehicle reaching speeds up to 15 m/s—introducing a substantial domain shift. To address this, we implemented an adaptive training buffer selection strategy, detailed in Sec. 3.4. As shown in Figure 6, the model undergoes real-time adaptation from the "SubT UGV model" to the "Tartan drive car model." In the figure, the blue line represents the estimated trajectory from the Tartan IMU model, while the dashed line indicates the ground truth. Notably, the testing data is **incrementally added** to the training set, leading to an uneven data distribution. However, our approach dynamically selects meaningful training samples (stationary, forward, left turn, and right turn) to ensure a balanced and diverse memory buffer, which facilitates online adaptation in real-time. Figure 6 demonstrates that our pipeline can effectively adapt in real-

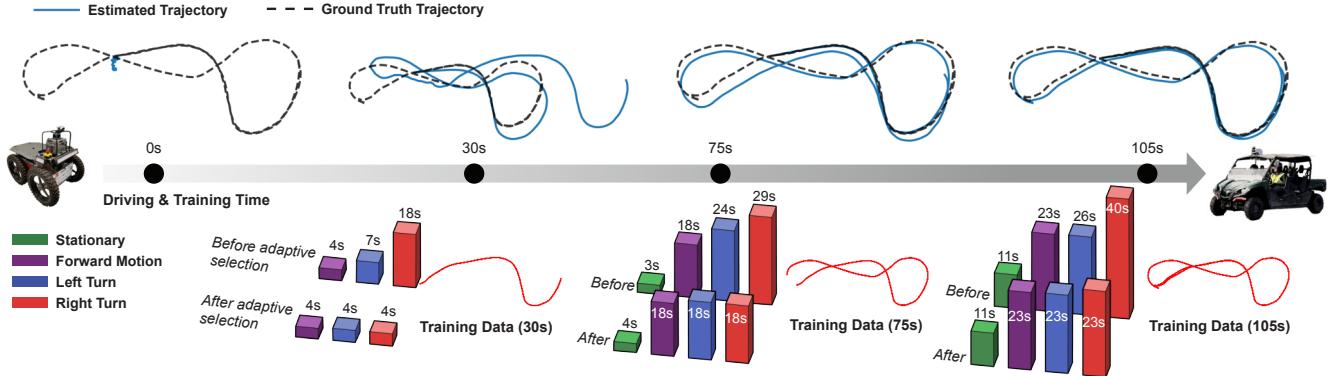


Fig. 6. Performance of online adaptation. The "SubT UGV model" adapts to the "Tartan drive model" within 105 seconds. Training data is **incrementally** added to the buffer, represented by the red trajectory at the bottom, and categorized into segments of stationary, forward motion, left turn, and right turn, each with varying time lengths. The height of each 3D bar indicates the duration of each segment in the buffer. After buffer selection, the data distribution becomes **uniform**, facilitating rapid adaptation.

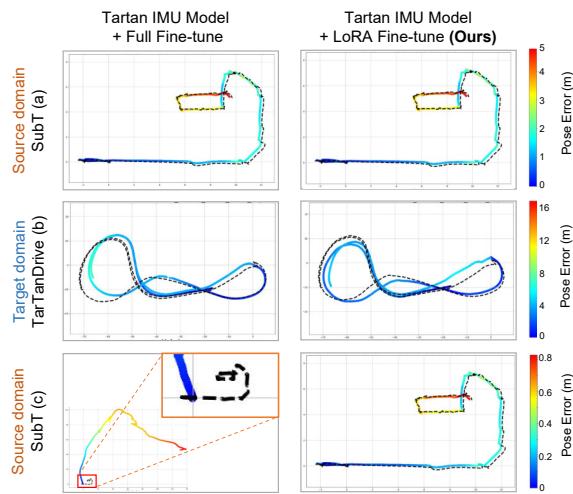


Fig. 7. Non-forgetting Characteristics of Our Model. We evaluate our Tartan IMU pretrained model’s generalization before and after fine-tuning from the SubT dataset [49] (source domain) to the TartanDrive dataset [34] (target domain). This demonstrates the non-forgetting capability of our LoRA-based fine-tuning adaptation, ensuring effective generalization across different scenarios.

time, completing the adaptation process in 105 seconds. We also incorporated online adaptation into the SLAM pipeline. Please see supplementary videos for this section.

4.4. Non-forgetting Characteristics Evaluation

To address **Q4**, we evaluate 2 models across source and target domains: a fully fine-tuned IMU model and an IMU model fine-tuned with a low-rank adapter (LoRA) [21].

In Figure 7(a), we present the odometry results from inference using our base Tartan IMU model without fine-tuning. Figure 7(b) shows model performance in an unseen environment (target domain) outside the pre-training dataset. Both models adapt to the new environment after fine-tuning. In Figure 7(c), we assess model performance on the source domain after domain adaptation. The

fully fine-tuned expert model performs worse than its original performance, demonstrating catastrophic forgetting. In contrast, the LoRA-based model maintains accurate estimations on the source domain, even after adapting to new knowledge. This demonstrates that LoRA-based approach effectively mitigates the forgetting problem.

5. Conclusion

This paper introduces Tartan IMU, a foundational model designed for generalizable IMU-based pose estimation across diverse robotic platforms. Tartan IMU improves accuracy by 36% compared to specialized models, leveraging over 100 hours of multi-platform data within its three-stage architecture. It utilizes LoRA fine-tuning to enable efficient domain adaptation with minimal additional parameters. Additionally, Tartan IMU supports online test-time adaptation, allowing for continuous performance improvement during deployment. The combination of these three stages makes Tartan IMU a versatile and robust solution for a wide range of robotic platforms.

6. Limitations

While TartanIMU exhibits strong generalization across vehicles, drones, and legged robots, it still cannot support arbitrary robotic platforms. However, our experiments show that the car motion head generalizes well to TartanDrive and SubT vehicles. We believe our categories—car, humanoid, quadruped, and drone—encompass most robots. For unseen platforms, introducing a new motion head or leveraging a mixture of existing experts (MoE)[45] presents a promising future direction. At the same time, we plan to explore the integration of both real-world and simulated IMU data to develop a more generalizable model to further improve the generalization capability. Additionally, our GMM-based buffer selection may struggle with complex robot motion patterns. Developing a more robust, robot-agnostic adaptation strategy would be a next step.

Acknowledgments

We sincerely thank the anonymous reviewers for their valuable feedback on this work. This research was supported by funding from the U.S. Army Research Lab under grants W911NF-23-S-0001, W911NF-21-20152, W911NF-24-20125, and W911NF-17-S-0003.

References

- [1] Yasin Almalioglu, Mehmet Turan, Muhamad Risqi U Saputra, Pedro PB de Gusmão, Andrew Markham, and Niki Trigoni. Selfvio: Self-supervised deep monocular visual-inertial odometry and depth estimation. *Neural Networks*, 150:119–136, 2022. [2](#)
- [2] Amado Antonini, Winter Guerra, Varun Murali, Thomas Sayre-McCord, and Sertac Karaman. The blackbird uav dataset. *The International Journal of Robotics Research*, 0(0):0278364920908331, 0. [6](#)
- [3] Amado Antonini, Winter Guerra, Varun Murali, Thomas Sayre-McCord, and Sertac Karaman. The blackbird dataset: A large-scale dataset for uav perception in aggressive flight. In *International Symposium on Experimental Robotics*, pages 130–139. Springer, 2018. [5](#)
- [4] Omri Asraf, Firas Shama, and Itzik Klein. Pdrnet: A deep-learning pedestrian dead reckoning framework. *IEEE Sensors Journal*, 22(6):4932–4939, 2021. [2](#)
- [5] Martin Brossard, Axel Barrau, and Silvere Bonnabel. Rins-w: Robust inertial navigation system on wheels. *The IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019. [2](#)
- [6] Martin Brossard, Axel Barrau, and Silvère Bonnabel. Ai-imu dead-reckoning. *IEEE Transactions on Intelligent Vehicles*, 5(4):585–595, 2020. [2, 3, 6](#)
- [7] Russell Buchanan, Varun Agrawal, Marco Camurri, Frank Dellaert, and Maurice Fallon. Deep imu bias inference for robust visual-inertial odometry with factor graphs. *IEEE Robotics and Automation Letters*, 8(1):41–48, 2022. [2](#)
- [8] Russell Buchanan, Marco Camurri, Frank Dellaert, and Maurice Fallon. Learning inertial odometry for dynamic legged robot state estimation. In *Conference on Robot Learning*, pages 1575–1584. PMLR, 2022. [2](#)
- [9] Xiya Cao, Caifa Zhou, Dandan Zeng, and Yongliang Wang. Rio: Rotation-equivariance supervised learning of robust inertial odometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6614–6623, 2022. [2, 4](#)
- [10] Changhao Chen and Xianfei Pan. Deep learning for inertial positioning: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 2024. [2](#)
- [11] Changhao Chen, Xiaoxuan Lu, Andrew Markham, and Niki Trigoni. Ionet: Learning to cure the curse of drift in inertial odometry. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018. [2](#)
- [12] Changhao Chen, Yishu Miao, Chris Xiaoxuan Lu, Linhai Xie, Phil Blunsom, Andrew Markham, and Niki Trigoni. Motiontransformer: Transferring neural inertial tracking between domains. In *The Conference on Artificial Intelligence (AAAI)*, 2019. [2](#)
- [13] Danpeng Chen, Nan Wang, Runsen Xu, Weijian Xie, Hujun Bao, and Guofeng Zhang. Rnn-in-vio: Robust neural inertial navigation aided visual-inertial odometry in challenging scenes. In *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 275–283, 2021. [4, 6, 7](#)
- [14] Giovanni Cioffi, Leonard Bauersfeld, Elia Kaufmann, and Davide Scaramuzza. Learned inertial odometry for autonomous drone racing. 2023. [2, 6](#)
- [15] Santiago Cortés, Arno Solin, and Juho Kannala. Deep learning based speed estimation for constraining strapdown inertial navigation on smartphones. In *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2018. [2](#)
- [16] Jeffrey Delmerico, Titus Cieslewski, Henri Rebecq, Matthias Faessler, and Davide Scaramuzza. Are we ready for autonomous drone racing? the UZH-FPV drone racing dataset. In *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2019. [5](#)
- [17] Mahdi Abolfazli Esfahani, Han Wang, Keyu Wu, and Shenghai Yuan. Aboldeepio: A novel deep inertial odometry network for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 21(5):1941–1950, 2019. [2](#)
- [18] Tobias Feigl, Sebastian Kram, Philipp Woller, Ramiz H Sidiqui, Michael Philippsen, and Christopher Mutschler. A bidirectional lstm for estimating dynamic human velocities from a single imu. In *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8. IEEE, 2019. [2](#)
- [19] Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation. In *Robotics: Science and Systems XI*, 2015. [1, 2](#)
- [20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. [4](#)
- [21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. [2, 5, 8](#)
- [22] Royina Karegoudra Jayanth, Yinshuang Xu, Ziyun Wang, Evangelos Chatzipantazis, Daniel Gehrige, and Kostas Daniilidis. Eqnio: Subequivariant neural inertial odometry, 2024. [3](#)
- [23] Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, and Natalia Díaz-Rodríguez. Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges, 2019. [7](#)
- [24] Feng Liu, Hongyu Ge, Dan Tao, Ruipeng Gao, and Zhang Zhang. Smartphone-based pedestrian inertial tracking: Dataset, model, and deployment. *IEEE Transactions on Instrumentation and Measurement*, 2023. [2](#)
- [25] Wenxin Liu, David Caruso, Eddy Ilg, Jing Dong, Anastasios I Mourikis, Kostas Daniilidis, Vijay Kumar, and Jakob Engel. Tlio: Tight learned inertial odometry. *IEEE Robotics and Automation Letters*, 5(4):5653–5660, 2020. [2, 4, 6](#)
- [26] R OpenAI et al. Gpt-4 technical report. *ArXiv*, 2303:08774, 2023. [2](#)

- [27] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 2
- [28] Tong Qin, Peiliang Li, and Shaojie Shen. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE transactions on robotics*, 34(4):1004–1020, 2018. 2
- [29] Yuheng Qiu, Chen Wang, Can Xu, Yutian Chen, Xunfei Zhou, Youjie Xia, and Sebastian Scherer. Airimu: Learning uncertainty propagation for inertial odometry. 2023. 2
- [30] Yuheng Qiu, Can Xu, Yutian Chen, Shibo Zhao, Junyi Geng, and Sebastian Scherer. Airio: Learning inertial odometry with enhanced imu feature observability, 2025. 3, 4
- [31] Bingbing Rao, Ehsan Kazemi, Yifan Ding, Devu M Shila, Frank M Tucker, and Liqiang Wang. Ctin: Robust contextual transformer network for inertial navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5413–5421, 2022. 2
- [32] Douglas A. Reynolds. Gaussian mixture models. In *Encyclopedia of Biometrics*, 2018. 5
- [33] João Paulo Silva do Monte Lima, Hideaki Uchiyama, and Rin-ichiro Taniguchi. End-to-end learning framework for imu-based 6-dof odometry. *Sensors*, 19(17):3777, 2019. 2
- [34] Matthew Sivaprakasam, Parv Maheshwari, Mateo Guaman Castro, Samuel Triest, Micah Nye, Steve Willits, Andrew Saba, Wenshan Wang, and Sebastian Scherer. Tartandrive 2.0: More modalities and better infrastructure to further self-supervised learning research in off-road driving tasks. *arXiv preprint arXiv:2402.01913*, 2024. 7, 8
- [35] Scott Sun, Dennis Melamed, and Kris Kitani. Idol: Inertial deep orientation-estimation and localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6128–6137, 2021. 2, 5, 6
- [36] Niko Sünderhauf, Oliver Brock, Walter Scheirer, Raia Hadsell, Dieter Fox, Jürgen Leitner, Ben Upcroft, Pieter Abbeel, Wolfram Burgard, Michael Milford, and Peter Corke. The limits and potentials of deep learning for robotics. *The International Journal of Robotics Research*, 37(4-5):405–420, 2018. 7
- [37] Hailiang Tang, Xiaoji Niu, Tisheng Zhang, You Li, and Jingnan Liu. Odonet: Untethered speed aiding for vehicle navigation without hardware wheeled odometer. *IEEE Sensors Journal*, 2022. 3
- [38] Xiaoqiang Teng, Pengfei Xu, Deke Guo, Yulan Guo, Runbo Hu, Hua Chai, and Didi Chuxing. Arldr: An accurate and robust pedestrian dead reckoning system for indoor localization on handheld smartphones. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10888–10893. IEEE, 2020. 2
- [39] Yiming Tu and Jin Xie. Undeeplio: Unsupervised deep lidar-inertial odometry. In *Asian Conference on Pattern Recognition*, pages 189–202. Springer, 2022. 2
- [40] Brandon Wagstaff and Jonathan Kelly. Lstm-based zero-velocity detection for robust inertial navigation. In *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8. IEEE, 2018. 2
- [41] Qu Wang, Haiyong Luo, Langlang Ye, Aidong Men, Fang Zhao, Yan Huang, and Changhai Ou. Pedestrian heading estimation based on spatial transformer networks and hierarchical lstm. *IEEE Access*, 7:162309–162322, 2019. 2
- [42] Yingying Wang, Hu Cheng, and Max Q-H Meng. A2dio: Attention-driven deep inertial odometry for pedestrian localization based on 6d imu. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 819–825. IEEE, 2022. 2
- [43] Peng Wei, Guoliang Hua, Weibo Huang, Fanyang Meng, and Hong Liu. Unsupervised monocular visual-inertial odometry network. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 2347–2354, 2021. 2
- [44] Xinguo Yu, Ben Liu, Xinyue Lan, Zhuoling Xiao, Shuisheng Lin, Bo Yan, and Liang Zhou. Azupt: Adaptive zero velocity update based on neural networks for pedestrian tracking. In *2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2019. 3
- [45] Seniha Esen Yuksel, Joseph N. Wilson, and Paul D. Gader. Twenty years of mixture of experts. *IEEE Transactions on Neural Networks and Learning Systems*, 23(8):1177–1193, 2012. 8
- [46] Ming Zhang, Mingming Zhang, Yiming Chen, and Mingyang Li. Imu data processing for inertial aided navigation: A recurrent neural network based approach. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3992–3998. IEEE, 2021. 2
- [47] Shibo Zhao, Peng Wang, Hengrui Zhang, Zheng Fang, and Sebastian Scherer. Tp-tio: A robust thermal-inertial odometry with deep thermalpoint. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4505–4512. IEEE, 2020. 1
- [48] Shibo Zhao, Hengrui Zhang, Peng Wang, Lucas Nogueira, and Sebastian Scherer. Super odometry: Imu-centric lidar-visual-inertial estimator for challenging environments. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8729–8736. IEEE, 2021. 1
- [49] Shibo Zhao, Yuanjun Gao, Tianhao Wu, Damanpreet Singh, Rushan Jiang, Haoxiang Sun, Mansi Sarawata, Yuheng Qiu, Warren Whittaker, Ian Higgins, et al. Subt-mrs dataset: Pushing slam towards all-weather environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22647–22657, 2024. 1, 5, 6, 7, 8
- [50] Baoding Zhou, Zhining Gu, Fuqiang Gu, Peng Wu, Chengjing Yang, Xu Liu, Linchao Li, Yan Li, and Qingquan Li. Deepvip: Deep learning-based vehicle indoor positioning using smartphones. *IEEE Transactions on Vehicular Technology*, 2022. 2