FormerReckoning: Physics Inspired Transformer for Accurate Inertial Navigation

Jiaqi Li*
Chenyu Zhao*
Yuzhu Mao
li-jq22@mails.tsinghua.edu.cn
zhaocy22@mails.tsinghua.edu.cn
myz20@tsinghua.org.cn
Shenzhen International Graduate School, Tsinghua University
China

Wenbo Ding†
ding.wenbo@sz.tsinghua.edu.cn
Shenzhen International Graduate School, Tsinghua University
Pengcheng Laboratory
RISC-V International Open Source Laboratory
Shanghai AI Laboratory
China

Xinlei Chen
chen.xinlei@sz.tsinghua.edu.cn
Shenzhen International Graduate School, Tsinghua University
Pengcheng Laboratory
RISC-V International Open Source Laboratory
China

Xiaoyang Qu
Jianzong Wang
quxiaoy@gmail.com
jzwang@188.com
Ping An Technology (Shenzhen) Co., Ltd.
China

Abstract

Although modern localization methods have achieved remarkable accuracy with various sensors, there are still some circumstances where only proprioceptive sensing works (Inertial Navigation). However, localization and navigation using only IMU sensors (costing less than $1000) still face significant challenges such as low accuracy and large cumulative errors when using traditional filter methods. Furthermore, AI-based approaches, while promising, often yield unpredictable and unreliable outputs. This paper proposes **FormerReckoning**, an inertial localization estimation framework for wheeled robotics that incorporates physical prompts into a Transformer framework to enhance translation estimation accuracy. Our tests show that FormerReckoning not only reduces mean translation errors to 0.72% but also surpasses all baseline models in performance, demonstrating its potential to provide reliable and precise localization in a cost-effective manner.

CCS Concepts

• **Networks** → **Physical links**; • **Computer systems organization** → **Embedded and cyber-physical systems**; **Sensors and actuators**.

Keywords

Transformer Framework, Inertial Navigation

ACM Reference Format:
Jiaqi Li, Chenyu Zhao, Yuzhu Mao, Wenbo Ding, Xinlei Chen, Xiaoyang Qu, and Jianzong Wang. 2024. FormerReckoning: Physics Inspired Transformer for Accurate Inertial Navigation. In *International Workshop on Physics Embedded AI Solutions in Mobile Computing (PICASSO 24), November 18–22, 2024, Washington D.C., DC, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3636534.3694736

*Both authors contributed equally to this research.
†Wenbo Ding is the corresponding author.

PICASSO 24, November 18–22, 2024, Washington D.C., DC, USA

ACM ISBN 979-8-4007-0489-5/24/11
https://doi.org/10.1145/3636534.3694736

1 Introduction

Within the realm of embodied intelligence, mobile intelligent systems encompass a range of entities, including unmanned aerial vehicles (UAVs) [1–3], autonomous ground vehicles [4], robotic arms [5], and service robots [6]. These emerging entities leverage AI algorithms to effectively interact with humans, requiring precise motion and control. Accurately estimating the agent's motion and localization is crucial for achieving precise interactions and is considered a vital piece of information for perceiving the external environment [7, 8]. Put simply, intelligent agents need to be aware of their spatial position within the environment.

While modern localization methods such as vision [9, 10], LiDAR [11], GNSS [12], and UWB [13] can provide highly accurate localization, they heavily rely on external environmental factors or infrastructure, making them vulnerable to failure in certain emergency scenarios. For instance, when detectable visual features are absent or there are RF disconnections, IMU-based proprioceptive sensing, also known as **Dead-Reckoning**, can offer more accurate localization for the system [14].

The traditional approach to implementing inertial navigation involves utilizing control theory methods such as the Extended Kalman Filter (EKF) [15] and the Unscented Kalman Filter(UKF) [16] for filtering and noise reduction, which comply with physical laws [17],

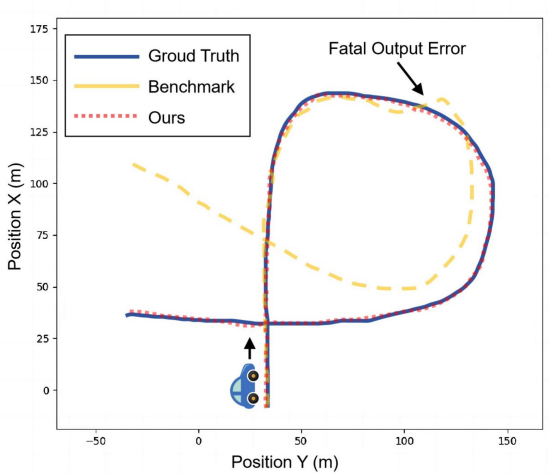

Figure 1: Without physical knowledge (Benchmark: CTIN), its outputs are susceptible to unreliable neural network inferences, while physical knowledge boosts our localization accuracy.

as is shown in Figure 2(b). This approach offers the advantage of lower computational requirements and real-time on-board operation. However, due to biases, drifts, and degradation of IMU [18–20], its data provides low accuracy and large accumulated errors, typically used with other modalities, such as vision and Lidar. Solely depending on IMU, it often fails to achieve reliable navigation due to the accumulation of errors [21, 22].

To address the issue of cumulative errors, some AI-based methods have been proposed to adjust certain parameters in the Kalman Filter, such as system and measurement noise covariance. Brossard et.al [21] initially incorporated two pseudo-variables predicted by a CNN into an Invariant EKF (IEKF). However, these pseudo-variables are only valid assuming that lateral and vertical velocities are zero. Alternatively, some approaches directly replace Kalman Filter with end-to-end learning [14, 23]. By leveraging learned noise distributions, these methods have successfully improved localization performance. Nevertheless, as is shown in Figure 1 and 2(a), the heavy reliance on purely black-box methods makes the results leading to unexpected outputs [1, 24–26].

The **primary objective** of this paper is to calibrate the localization result in a Dead-Reckoning system considering cumulative errors resulting from drifts and degradation, while ensuring reasonable and consistent outputs. To tackle this issue, the first challenge **(C1)** lies in the hard-to-model noisy and degraded IMU sensor data. The second challenge **(C2)** involves maintaining the localization output compliant with the kino dynamics and physical reality using neural networks.

To address these concerns, this paper introduces a novel Dead-Reckoning framework, called FormerReckoning, to accurately estimate the motion and localization of smart ground vehicles. First, a carefully designed Transformer [27] model is used to better model the noisy and degraded data with its non-linear modeling, spatial-temporal, and feature extraction ability. Second, a Kalman Filter is embedded into the framework to provide physical constraints, as is shown in Figure 2(c). The main contributions of this paper can be summarized as follows:

- A new FormerReckoning framework incorporating spatial-temporal constraints for the noisy and degraded IMU data.
- A Transformer model integrated with a Kalman Filter to calibrate the estimation of localization using IMU sensor data, with the advantage of enhancing localization accuracy while maintaining physical law consistency.
- Verification of superior calibration results on datasets compared to other methods.

The rest of this paper is organized as follows: Section 2 introduces problem definitions. Section 3 describes the framework design. Section 4 represents experimental results and evaluation. In the last, Section 5 concludes this paper.

2 Problem Definitions

In this section, we aim to provide a comprehensive background definition of IMU modeling. Firstly, we present the dynamic formulation of an IMU, which encompasses the fundamental principles governing its behavior. Subsequently, we delve into the definition of the physical prompts, tailored specifically to account for the unique motion characteristics exhibited by car-like vehicles.

2.1 IMU-based Dynamics

The IMU can provide the measurement of angular velocity $\boldsymbol{\omega}_\mathrm{I} \in \mathbb{R}^3$ and acceleration rate $\mathbf{a}_\mathrm{I} \in \mathbb{R}^3$ of the agent's motion with noise and bias as follows,

$$\begin{aligned} \boldsymbol{\omega}_\mathrm{I} &= \boldsymbol{\omega} + \mathbf{n}_\mathrm{I}^{\boldsymbol{\omega}}, \\ \mathbf{a}_\mathrm{I} &= \mathbf{a} + \mathbf{n}_\mathrm{I}^{\mathbf{a}}, \end{aligned} \tag{1}$$

where $\boldsymbol{\omega}$ and $\mathbf{a}$ are actual motion information, and $\mathbf{n}_\mathrm{I}^{\boldsymbol{\omega}}$ and $\mathbf{n}_\mathrm{I}^{\mathbf{a}}$ are the sums of noise and bias of IMU angular velocity and acceleration rate that need to be eliminated. Here, we assume that the IMU is rigidly fixed on agents and their frames are aligned. So we use IMU motions to represent agent motion. The kinematic model at time step $n+1$ can be described as

$$\begin{aligned} \mathbf{R}^{n+1} &= \mathbf{R}^n exp((\boldsymbol{\omega}^n dt)_\times), \\ \mathbf{v}^{n+1} &= \mathbf{v}^n + (\mathbf{R}^n \mathbf{a}^n + \mathbf{g}) dt, \\ \mathbf{p}^{n+1} &= \mathbf{p}^n + \mathbf{v}^n dt, \end{aligned} \tag{2}$$

where dt denotes the interval of two sampling instants, $\mathbf{g}$ is the acceleration of gravity, $\mathbf{R}^n \in SO(3)$ (Special Orthogonal Group [28]) is the 3×3 rotation matrix of the IMU orientation, i.e. that maps the IMU frame to the world frame. $\mathbf{v}^n \in \mathbb{R}^3$ and $\mathbf{p}^n \in \mathbb{R}^3$ represent IMU velocity and position in the world frame. The symbol $(\cdot)_\times$ represents the skew-symmetric matrix associated with the cross product.

2.2 Problem Modelling

Based on the IMU-based Dynamics, in the rest of the paper, we tackle the following problem: **IMU Dead-Reckoning Problem.** Given an initial known configuration $(\mathbf{R}^0, \mathbf{v}^0, \mathbf{p}^0)$, perform in real-time IMU dead-reckoning, i.e. estimate the IMU and car variables at time step $n+1$

$$\mathbf{x}^{n+1} := \left[\mathbf{v}^{n+1}, \mathbf{p}^{n+1}\right]. \tag{3}$$

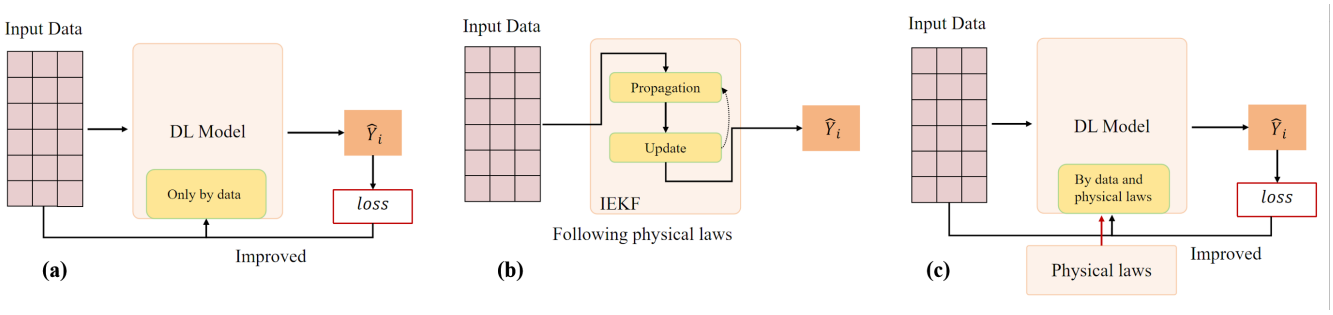

Figure 2: The difference of three frameworks on Dead-Reckoning. (a) A typical deep learning model whose performance has a close correlation with data. (b) An overview of an IEKF for denoising and estimating, constrained by physical laws. (c) Our framework: physics-inspired deep learning, combining the advantages of pure physics-based and pure data-driven methods.

Note that the estimation use the inertial measurements $\boldsymbol{\omega}_{\mathrm{I}}^{n}$ and $\mathbf{a}_{\mathrm{I}}^{n}$.

3 Methodology

3.1 System Overview

The proposed physics-inspired Transformer is shown in Figure 3, which incorporates physical quantities into the Transformer framework as prompts to achieve more accurate translation estimation. Our main insight is that the Transformer [29, 30] architecture with carefully designed self-attention mechanism and training framework incorporating physical prompts into the Transformer's inputs and the inputted Kalman filter result serves as the constraint of the Transformer model learning to stabilize the outputs.

3.2 FormerReckoning Design

Physical Prompts. i) The first prompt comprises the calibrated values of acceleration and angular velocity by the Kalman Filter of acceleration $\mathbf{a}_{\mathrm{K}}^{n+1}$ and angular velocity $\boldsymbol{\omega}_{\mathrm{K}}^{n+1}$ at the predicted time $n+1$. In our scenario, when predicting $\mathbf{x}^{n+1}$, we have already acquired information from the current time step, such as $\mathbf{a}_{\mathrm{I}}^{n+1}$and $\boldsymbol{\omega}_{\mathrm{I}}^{n+1}$, albeit with potential errors due to measurement inaccuracies. In traditional Transformer models, the use of Masked Multi-Head Attention obscures this portion of information, so we use these two extra pieces of physical information as input prompts. For the first physical information, we employ the Kalman filtering algorithm to obtain better estimates of the quantities at the predicted time. The values $\mathbf{a}_{\mathrm{K}}^{n+1}$ and $\boldsymbol{\omega}_{\mathrm{K}}^{n+1}$ serve as inputs for the temporal decoder. **ii)** The second prompt is the covariance $\mathbf{N}^{n+1}$ that we set for the velocity values in the lateral ($\leftrightarrow$) and vertical ($\updownarrow$) direction of the IMU. We calculate $\mathbf{N}^{n+1}$ as the physics prompt considering that in the real system, the covariance of the velocity in the lateral and vertical direction is dynamically changing. For instance, when turning, the covariance in the lateral direction is much greater than that in the straight line. In the wheeled agent frame, the lateral and vertical velocities are roughly zero, so we generate $\mathbf{N}^{n+1}$ according to:

$$\mathbf{v}^{n+1} = \begin{bmatrix} v_{\leftrightarrow}^{n+1} \\ v_{\updownarrow}^{n+1} \end{bmatrix} + \begin{bmatrix} n_{\leftrightarrow}^{n+1} \\ n_{\updownarrow}^{n+1} \end{bmatrix}, \tag{4}$$

where $\mathbf{v}^{n+1}$ is calculated followed by Equation 2 and the noises $\mathbf{n}^{n+1} = [n_{\leftrightarrow}^{n+1}, n_{\updownarrow}^{n+1}]^{\top}$ are assumed centered and Gaussian with covariance matrix $\mathbf{N}^{n+1} \in \mathbb{R}^{2\times 2}$. $v_{\leftrightarrow}^{n+1} \approx 0$ and $v_{\updownarrow}^{n+1} \approx 0$ are the velocities in the lateral and vertical direction, respectively. Thus, the elements of $\mathbf{N}^{n+1}$ are the pseudo-variables that we consider as part of the input of the spatial decoder.

Data Streaming. To exploit temporal characteristics of IMU samples, a sliding window with size $m+2$ is used to prepare datasets at timestamp $n+1$, denoted by $\mathbf{x}_{0:m+1}^{n+1} = \left[x^{n-m}, \ldots, x^{n+1}\right]$. That is, we adopt this rolling mechanism to build the ground truth of velocities and positions: $\mathbf{v}_{0:m+1}^{n+1}$ and $\mathbf{p}_{0:m+1}^{n+1}$. Therefore, we can predict velocities and positions from an input window of IMU measurements, as shown in Equation 5.

$$[\hat{\mathbf{v}}^{n+1}, \hat{\mathbf{p}}^{n+1}]_{0:m+1} = \mathcal{F}_{\theta}([\hat{\mathbf{v}}^{n}, \hat{\mathbf{p}}^{n}]_{0:m}, [\mathbf{a}_{\mathrm{K}}^{n+1}, \boldsymbol{\omega}_{\mathrm{K}}^{n+1}]_{0:m+1}, \mathbf{N}^{n+1}). \tag{5}$$

Embedding. To prepare the IMU samples for input into the encoder and decoder, it is necessary to compute feature representations. We employ two types of embeddings: Spatial Embedding and Temporal Embedding. Spatial Embedding involves utilizing a 1D convolutional neural network [31] to learn spatial representations from the IMU samples. Temporal Embedding leverages a 1-layer LSTM [32] model to exploit the temporal information present in the IMU samples. The LSTM layer learns to capture the dependencies and patterns over time.

Temporal Encoder. The encoder maps an input sequence of $[\mathbf{a}_{\mathrm{K}}^{n+1}, \boldsymbol{\omega}_{\mathrm{K}}^{n+1}]_{0:m+1}$ to a sequence of continuous representations. To capture spatial knowledge of IMU samples at each timestamp, the Temporal Encoder consists of multi-head attention blocks and a feedforward network.

Spatial Decoder. Noting the physical relationship among $\mathbf{N}$, $\mathbf{v}$, and $\mathbf{p}$ described in Equation 2, to fully capture the contextual information among neighboring keys, we employ a self-attention mechanism within the local region, as shown on the right of Figure 3. Specifically, we apply a 3×3 group convolution [33] over all the neighboring keys to extract local contextual representations for each key.

Velocity and Position. Finally, two MLP-based branch heads regress velocity $\hat{\mathbf{v}}_{0:m+1}^{n+1}$ and the position $\hat{\mathbf{p}}_{m+1}^{n+1}$. We utilized multi-task learning [34] to model the output as two regression tasks containing a time window, ensuring high performance and consistent output. Inspired by [14], we derive a multi-task loss function by maximizing the Gaussian likelihood with uncertainty. First, we define our likelihood as a Gaussian with mean given by the model output as $p_u\left(y \mid \mathcal{F}_{\theta}(x)\right) = \mathcal{N}\left(\mathcal{F}_{\theta}(x), \delta^2\right)$, where δ is an observation noise. Next, we derive the model's minimization objective as a Negative Log-Likelihood (NLL) of two model outputs $\mathbf{v}$ (velocity)

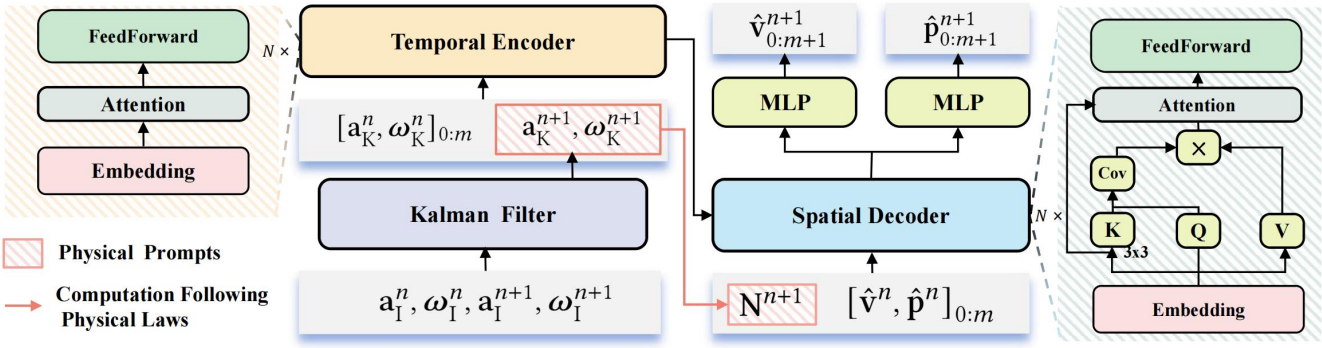

Figure 3: FormerReckoning framework overview.

and $\mathbf{p}$ (position):

$$
\begin{aligned}
&\mathcal{L}\left(\mathcal{F}_\theta, \delta_{\mathbf{v}}, \delta_{\mathbf{p}}\right) \\
&= -\left(\log\left(\mathcal{N}\left(\mathbf{v}_{0:m+1}^{n+1}; \mathcal{F}_\theta(x), \delta_{\mathbf{v}}^2\right)\right) + \log\left(\mathcal{N}\left(\mathbf{p}_{0:m+1}^{n+1}; \mathcal{F}_\theta(x), \delta_{\mathbf{p}}^2\right)\right)\right) \\
&\propto \underbrace{\frac{1}{2\delta_{\mathbf{v}}^2}\left\|\mathbf{v}_{0:m+1}^{n+1} - \hat{\mathbf{v}}_{0:m+1}^{n+1}\right\|^2 + \log \delta_{\mathbf{v}}}_{\text{Velocity}} + \underbrace{\frac{1}{2\delta_{\mathbf{p}}^2}\left\|\mathbf{p}_{0:m+1}^{n+1} - \hat{\mathbf{p}}_{0:m+1}^{n+1}\right\|^2 + \log \delta_{\mathbf{p}}}_{\text{Position}} \\
&= \frac{1}{2\delta_{\mathbf{v}}^2}\mathcal{L}_{\mathbf{v}} + \frac{1}{2\delta_{\mathbf{p}}^2}\mathcal{L}_{\mathbf{p}} + \log \delta_{\mathbf{v}}\delta_{\mathbf{p}}.
\end{aligned}
\tag{6}
$$

The mean square error (MSE) loss functions are denoted by $\mathcal{L}_{\mathbf{v}}$ and $\mathcal{L}_{\mathbf{p}}$ respectively.

4 Evaluation

This section aims to assess the performance of FormerReckoning and compare it with three alternative methods. In addition to the comparative analysis, we also conducted an ablation experiment to gain deeper insights into the contribution of the physical prompts efficiency to the overall performance of FormerReckoning.

4.1 Experiments Settings

For the training and testing of FormerReckoning in this paper, we choose PyTorch with version 2.0.1 and Python 3.9.16 as the framework to implement all algorithms. To be consistent with the baselines' experimental settings, we conduct training and testing on a server with NVIDIA RTX A6000 GPU, Intel Xeon(R) Gold 6242R CPU @ 3.10GHz × 80, and 880GB RAM.

4.2 Evaluation Metrics and Baselines

To assess the system's performance, we consider three metrics that capture the quality of localization estimation:

- **Relative Translation Error (E_t):** measures the average relative translation incremental error for different sub-sequences of distance, expressed as a percentage of the total driving distance. It quantifies the accuracy of translation estimation.
- **Relative Rotation Error (E_r):** calculates the average relative rotation incremental error for sub-sequences of distance, represented in degrees per meter. It evaluates the accuracy of rotation estimation.
- **Root Mean Squared Error (*RMSE*):** represents the average translation error, evaluating the absolute accuracy of the translation.

For comparative analysis, we benchmark FormerReckoning against three other methods:

- **KF:** The KF method adopts a fundamental dead-reckoning approach by integrating the acceleration rates and angular velocities obtained from the Kalman Filter. This technique leverages the principles of state estimation to predict the position and orientation.
- **AI-IMU** [21]**:** AI-IMU presents an original calibration method for IMU navigation. This approach combines CNN with the Invariant Extended Kalman Filter, resulting in improved dead-reckoning accuracy compared to traditional methods.
- **CTIN** [14]**:** CTIN introduces a novel approach for recovering 2-dimensional velocity from IMU measurements. This method employs a contextual Attention-based model, which utilizes a Transformer model to consider both temporal and spatial features.

4.3 Performance Results

We evaluate FormerReckoning(FR), along with three baselines, on the KITTI [35] dataset to compare their localization performances. In the testing process, we utilize the Dead-Reckoning methods with raw IMU data, including angular velocity $\boldsymbol{\omega}_I$ and acceleration rate $\mathbf{a}_\mathbf{I}$, as input. The methods output the position and velocity of the vehicle along the trajectories.

In Table 1, we present the evaluation results of the methods on several sequences. The dead-reckoning method relying on KF exhibits large errors and produces inaccurate localization results. The results of AI-IMU show smaller errors while the vehicle is in motion, with the metrics being almost a dozen times lower than those of KF. Moreover, the contextual Attention-based method CTIN achieves smaller average errors. Notably, our method FR consistently achieves lower errors than the other three methods. These findings affirm that our method provides more accurate localization capabilities.

4.4 Ablation Study

To evaluate the effectiveness of the physical prompts, we compare the performance of four methods: FormerReckoning (FR) without the physical prompts, KF, AI-IMU, and CTIN.

Table 1: The relative translation errors E_t and relative rotation errors E_r which are tested on six KITTI datasets

2*Test Seq.	2*Distance	2*Duration	KF		AI-IMU		CTIN		Ours	
			$E_t(\%)$	$E_r(\%)$	$E_t(\%)$	$E_r(\%)$	$E_t(\%)$	$E_r(\%)$	$E_t(\%)$	$E_r(\%)$
Seq1	0.4	27	6.79	0.26	0.35	0.08	0.36	0.10	0.31	0.06
Seq2	1.2	110	13.44	0.32	0.97	0.20	0.91	0.23	0.77	0.17
Seq3	0.7	110	19.93	0.59	0.84	0.32	0.87	0.27	0.66	0.23
Seq4	3.2	407	15.59	0.71	1.48	0.32	1.31	0.36	1.11	0.29
Seq5	1.7	159	29.45	0.63	0.80	0.22	0.71	0.19	0.65	0.19
Seq6	0.9	120	11.49	0.43	0.98	0.23	0.93	0.17	0.80	0.16
Total score			16.12	0.49	1.10	0.23	1.02	0.22	**0.72**	**0.18**

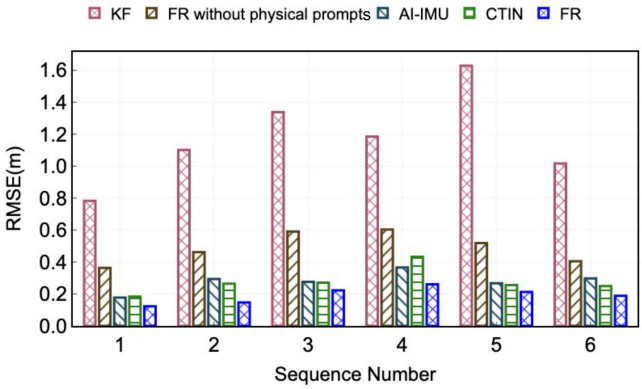

Figure 4: The *RMSE* with and without physical prompts on different KITTI dataset sequences. FormerReckoning always has the best accuracy, but without physical prompts, it degrades.

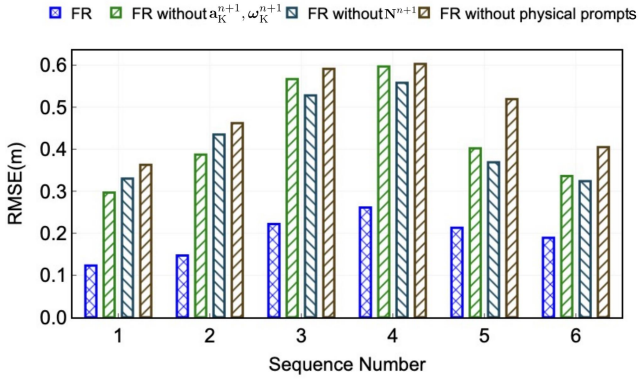

Figure 5: The *RMSE* with and without $a_K^{n+1}, \omega_K^{n+1}$ and N^{n+1} on different dataset sequences. The results show that both the outputs from the Kalman Filter and variance N^{n+1} contribute to the estimation improvement.

Table 2: Parameter Scale and Training Costs

Method	№ of Parameters (1×10^5)	GPU time(ms)
AI-IMU	0.84	13.26
CTIN	5.57	74.50
OURS	4.39	56.23

As is shown in Figure 4, FR without the physical prompts consistently exhibits lower *RMSE* compared to KF. This suggests that even without the physical prompts, FormerReckoning's underlying Transformer-like model shows promising performance in dead-reckoning estimation. However, its accuracy falls short when compared to AI-IMU. Without physical constraints, the model cannot stabilize its output, leading to suboptimal results. In contrast, the complete FR, incorporating physical prompts, achieves the lowest *RMSE* among all the methods. This result highlights the significant enhancement in dead-reckoning accuracy: by incorporating the physical constraints and prior knowledge into the model, the complete FR achieves superior performance in estimating the system's trajectory.

In Figure 5, the results illustrate the contributions of two parts of the physical prompts separately. It proves that the proposed Transformer with the aided $\mathbf{a}_K^{n+1}, \boldsymbol{\omega}_K^{n+1}$ and $\mathbf{N}^{n+1}$ boosts the performance of FR.

4.5 Computational Consumption

It is illustrated in Table 2 that, our FormerReckong framework achieves better computational efficiency with fewer parameter numbers than CTIN. Considering the localization performance, induced physical prompts provide more reliability with less computation effort.

5 Conclusion

This paper introduces FormerReckoning, a framework using a physics-inspired Transformer for accurate IMU-based estimation of agent translation and rotation, achieving a translation error of only 0.72%. Future work will adapt and expand FormerReckoning to enhance Dead-Reckoning across diverse autonomous systems in robotics, transportation, and surveillance.

6 Acknowledgements

This work was supported by the National Key R&D Program of China under Grant No.2022ZD0160504, by Tsinghua Shenzhen International Graduate School-Shenzhen Pengrui Young Faculty Program of Shenzhen Pengrui Foundation (No. SZPR2023005), by Shenzhen Ubiquitous Data Enabling Key Lab under Grant No. ZDSYS20220527171406015, by Guangdong Innovative and Entrepreneurial Research Team Program (2021ZT09L197).

References

[1] Chenyu Zhao, Haoyang Wang, Jiaqi Li, Fanhang Man, Shilong Mu, Wenbo Ding, Xiao-Ping Zhang, and Xinlei Chen. Smoothlander: A quadrotor landing control system with smooth trajectory guarantee based on reinforcement learning. In *Adjunct Proceedings of the 2023 ACM International Joint Conference on Pervasive and Ubiquitous Computing & the 2023 ACM International Symposium on Wearable Computing*, pages 682–687, 2023.

[2] Xuecheng Chen, Haoyang Wang, Zuxin Li, Wenbo Ding, Fan Dang, Chengye Wu, and Xinlei Chen. Deliversense: Efficient delivery drone scheduling for crowdsensing with deep reinforcement learning. In *Adjunct Proceedings of the 2022 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2022 ACM International Symposium on Wearable Computers*, pages 403–408, 2022.

[3] Xinlei Chen, Aveek Purohit, Carlos Ruiz Dominguez, Stefano Carpin, and Pei Zhang. Drunkwalk: Collaborative and adaptive planning for navigation of micro-aerial sensor swarms. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, SenSys '15, page 295–308, New York, NY, USA, 2015. Association for Computing Machinery.

[4] Xinlei Chen, Susu Xu, Xinyu Liu, Xiangxiang Xu, Hae Young Noh, Lin Zhang, and Pei Zhang. Adaptive hybrid model-enabled sensing system (hmss) for mobile fine-grained air pollution estimation. *IEEE Transactions on Mobile Computing*, 21(6):1927–1944, 2020.

[5] Shilong Mu, Shoujie Li, Hongfa Zhao, Zihan Wang, Xiao Xiao, Zenan Lin, Ziwu Song, Huaze Tang, Qinghao Xu, Dongkai Wang, et al. A platypus-inspired electro-mechanosensory finger for remote control and tactile sensing. *Nano Energy*, 116:108790, 2023.

[6] Zenan Lin, Kai Chong Lei, Shilong Mu, Ziwu Song, Yuan Dai, Wenbo Ding, and Xiao-Ping Zhang. Multimodal surface sensing based on hybrid flexible triboelectric and piezoresistive sensor. In *Adjunct Proceedings of the 2022 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2022 ACM International Symposium on Wearable Computers*, pages 421–426, 2022.

[7] Xinlei Chen, Aveek Purohit, Carlos Ruiz Dominguez, Stefano Carpin, and Pei Zhang. Drunkwalk: Collaborative and adaptive planning for navigation of micro-aerial sensor swarms. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 295–308, 2015.

[8] Zhuozhu Jian, Zejia Liu, Haoyu Shao, Xueqian Wang, Xinlei Chen, and Bin Liang. Path generation for wheeled robots autonomous navigation on vegetated terrain. *IEEE Robotics and Automation Letters*, 2023.

[9] Jingao Xu, Hao Cao, Zheng Yang, Longfei Shangguan, Jialin Zhang, Xiaowu He, and Yunhao Liu. {SwarmMap}: Scaling up real-time collaborative visual {SLAM} at the edge. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*, pages 977–993, 2022.

[10] Stephen Se, David G Lowe, and James J Little. Vision-based global localization and mapping for mobile robots. *IEEE Transactions on robotics*, 21(3):364–375, 2005.

[11] Zhuozhu Jian, Qixuan Li, Shengtao Zheng, Xueqian Wang, and Xinlei Chen. Lvcp: Lidar-vision tightly coupled collaborative real-time relative positioning. *arXiv preprint arXiv:2407.10782*, 2024.

[12] Xinlei Chen, Carlos Ruiz, Sihan Zeng, Liyao Gao, Aveek Purohit, Stefano Carpin, and Pei Zhang. H-drunkwalk: Collaborative and adaptive navigation for heterogeneous mav swarm. *ACM Transactions on Sensor Networks (TOSN)*, 16(2):1–27, 2020.

[13] Danping Su, Xianyao Wang, Sicong Liu, and Wenbo Ding. Four-dimensional indoor visible light positioning: A deep-learning-based perspective. *Journal of the Franklin Institute*, 360(6):4071–4090, 2023.

[14] 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*, volume 36, pages 5413–5421, 2022.

[15] Mingyang Li and Anastasios I Mourikis. High-precision, consistent ekf-based visual-inertial odometry. *The International Journal of Robotics Research*, 32(6):690–711, 2013.

[16] Song Qi and Han Jian-Da. An adaptive ukf algorithm for the state and parameter estimations of a mobile robot. *Acta Automatica Sinica*, 34(1):72–79, 2008.

[17] Jiahao Li, Huandong Wang, and Xinlei Chen. Physics-informed neural ode for post-disaster mobility recovery. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1587–1598, 2024.

[18] Rui Sun, Yuanxi Yang, Kai-Wei Chiang, Thanh-Trung Duong, Kuan-Ying Lin, and Guang-Je Tsai. Robust imu/gps/vo integration for vehicle navigation in gnss degraded urban areas. *IEEE Sensors Journal*, 20(17):10110–10122, 2020.

[19] Saurabh Godha, Gérard Lachapelle, and M Elizabeth Cannon. Integrated gps/ins system for pedestrian navigation in a signal degraded environment. In *Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2006)*, pages 2151–2164, 2006.

[20] Jong Tai Jang, Angel Santamaria-Navarro, Brett T Lopez, and Ali-akbar Aghamohammadi. Analysis of state estimation drift on a mav using px4 autopilot and mems imu during dead-reckoning. In *2020 IEEE Aerospace Conference*, pages 1–11. IEEE, 2020.

[21] Martin Brossard, Axel Barrau, and Silvère Bonnabel. Ai-imu dead-reckoning. *IEEE Transactions on Intelligent Vehicles*, 5(4):585–595, 2020.

[22] Haoyang Wang, Xuecheng Chen, Yuhan Cheng, Chenye Wu, Fan Dang, and Xinlei Chen. H-swarmloc: Efficient scheduling for localization of heterogeneous mav swarm with deep reinforcement learning. In *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*, pages 1148–1154, 2022.

[23] Martin Brossard, Silvere Bonnabel, and Axel Barrau. Denoising imu gyroscopes with deep learning for open-loop attitude estimation. *IEEE Robotics and Automation Letters*, 5(3):4796–4803, 2020.

[24] Guanya Shi, Xichen Shi, Michael O'Connell, Rose Yu, Kamyar Azizzadenesheli, Animashree Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural lander: Stable drone landing control using learned dynamics. In *2019 international conference on robotics and automation (icra)*, pages 9784–9790. IEEE, 2019.

[25] Hendry Ferreira Chame, Matheus Machado dos Santos, and Silvia Silva da Costa Botelho. Neural network for black-box fusion of underwater robot localization under unmodeled noise. *Robotics and Autonomous Systems*, 110:57–72, 2018.

[26] Ming Dai, Jianhong Hu, Jiedong Zhuang, and Enhui Zheng. A transformer-based feature segmentation and region alignment method for uav-view geo-localization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(7):4376–4389, 2021.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[28] Richard M Murray, Zexiang Li, and S Shankar Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[29] Yuxuan Liu, Haoyang Wang, Fanhang Man, Jingao Xu, Fan Dang, Yunhao Liu, Xiao-Ping Zhang, and Xinlei Chen. Mobiair: Unleashing sensor mobility for city-scale and fine-grained air-quality monitoring with airbert. In *Proceedings of the 22nd Annual International Conference on Mobile Systems, Applications and Services*, pages 223–236, 2024.

[30] Haoyang Wang, Yuxuan Liu, Chenyu Zhao, Jiayou He, Wenbo Ding, and Xinlei Chen. Califormer: Leveraging unlabeled measurements to calibrate sensors with self-supervised learning. In *Adjunct Proceedings of the 2023 ACM International Joint Conference on Pervasive and Ubiquitous Computing & the 2023 ACM International Symposium on Wearable Computing*, pages 743–748, 2023.

[31] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. Recent advances in convolutional neural networks. *Pattern recognition*, 77:354–377, 2018.

[32] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[33] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[34] Jonathan Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198, 2000.

[35] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.