Performance Evaluation of Machine Learning Models for Mobile Robot Odometry Error Prediction Using Gazebo

Anitha Mary X[1], Shashi Kant Gupta[2]

[1] Post Doctoral Fellow, Lincoln University College, Petaling Jaya, Selangor Darul Ehsan-47301, Malaysia ; [2] Computer Science and Engineering, Lincoln University College, Petaling Jaya, Selangor Darul Ehsan-47301, Malaysia

Email ID anithajohnson2003@gmail.com; shashigupta@lincoln.edu.my

Abstract

Precise localisation is a basic requirement for autonomous mobile robots operating in complex indoor environments. The wheel encoder based odometry is the most widely used dead-reckoning technique because of its simplicity, cheap implementation cost and great computing efficiency. However, the readings from the wheel encoder are very sensitive to the accumulated errors due to physical wheel slippage, structural misalignment, uneven surface terrains and deterministic mechanical flaws. Such unchecked concerns will cause localisation drift and may degrade the stability of the route planning and obstacle avoidance behaviours. This paper systematically evaluates the performance of three state-of-the-art gradient boosting frameworks, namely XGBoost, LightGBM and CatBoost, for the dynamic prediction and correction of cumulative localisation errors.

Keywords: Autonomous Robotics, Odometry Error Prediction, Gradient Boosting, XGBoost, LightGBM, CatBoost, Gazebo Simulation.

Introduction

The operational foundation of autonomous mobile robots in missions like warehouse distribution, industrial monitoring, infrastructure inspection, and public transit logistics is reliable and accurate localisation [1]. Wheel encoder based odometry is the most popular navigation component among many dead-reckoning methods due to its low structural cost, easy software integration and self-contained computing efficiency. Notwithstanding these practical benefits the encoder metrics are, by nature, subject to unbounded mistakes. Such deficiencies are generally categorised as systematic errors[2] (e.g. changes of structural wheel diameters, axle misalignment) and non-systematic mistakes (e.g. irregular wheel slipping on debris, micro-skidding, floor height variations). Since dead-reckoning is cumulative , these discrepancies accumulate over time . This leads to a significant amount of drift that can compromise the safety of autonomous navigation[3]. To overcome these limitations, recent robotics research is using data-driven machine learning methods to explicitly model complex non-systematic disturbances from tabular sensor streams. Gradient Boosted Decision Tree (GBDT) frameworks are one of the most effective approaches for processing structured tabular logs due to the speed, resistance to noise and ability to match complicated non-linear connections[4].

Related work

Dead-reckoning error configurations are classically mitigated through kinematic calibration procedures, especially those based on the UMBmark benchmark test that deals with systematic differences in wheel diameter and wheelbase. These geometric procedures are useful for predictable mechanical wear but do not help with the random slip conditions that happen in real-time in dynamic unmapped terrains [3].The research has therefore moved toward state estimation filters based on Multi-Sensor Data Fusion (MSDF). Current frameworks combine wheel encoder measurements with sensory measurements from IMU, LiDAR and vision systems using Extended Kalman Filters (EKF) and simultaneous localisation and mapping (SLAM) solvers [1, 6]. while these techniques improve the tracking performance, they require a large computer capacity and are based on complicated mathematical models which have difficulty in handling very nonlinear slippage dynamics.In more recent years, data driven regression methods have been proposed to overcome these constraints of kinematic models [2]. The emergence of tree boosting algorithms provides a rich new toolbox for correcting the state of robots. The strong robustness to data noise, inherent regularisation features and outstanding running efficiency of these state-of-the-art Gradient Boosted Decision Tree (GBDT) algorithms have set a new performance bar for structured data [4].

Key Contributions

This study makes three main contributions:

1. Generation of a multi-trajectory high-fidelity robotic odometry dataset in Gazebo, with a wide range of motion profiles and systematic drifts occurrences [5].
2. A formal study of the tradeoffs between accuracy and computing cost of the three popular GBDT models[8].
3. Finding a suitable software based corrective mechanism to reduce the tracking errors without adding more physical sensor packages [10].

Method, Experiments and Results

The whole simulation process was done in an ordered step by step pipeline using the Robot Operating System (ROS) framework coupled to the Gazebo simulation engine [5] to generate a complete dataset including systematic and non-systematic error profiles.The physical environment was built in the Gazebo simulation environment which emulates a flat indoor testing floor surrounded by boundary walls and obstacle blocks to mimic common real world navigation obstacles as shown in Figure 1.

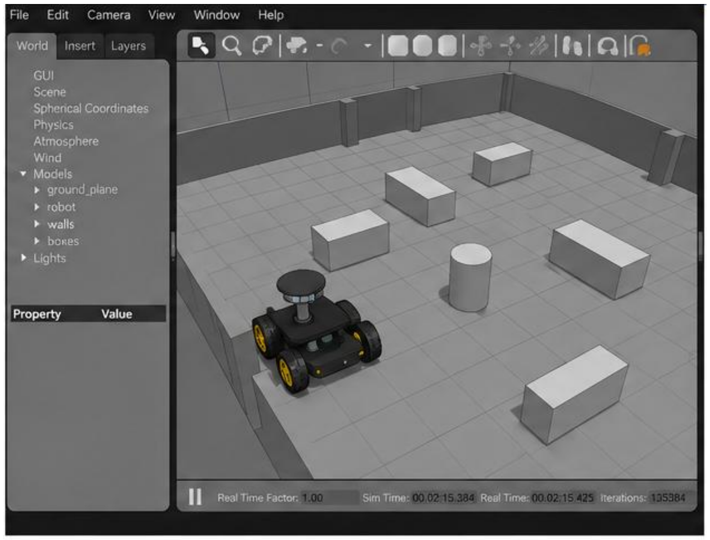

Figure 1 Gazebo Simulation

The mobile robot was run through three different reference trajectories in an effort to record a wide range of motion dynamics: Straight Line , Square Path and circular path. The data collection procedure collected a total of 10,000 spatial samples along the three trajectories The final dataset was split into 70% for training (~7,000 samples), 15% for validation (1,500 samples) and 15% for testing (1,500 samples) to assess the generalisation performance. Each data is represented as a 10-dimensional feature vector:

XGBoost (Extreme Gradient Boosting)

XGBoost constructs trees sequentially, optimising a regularised objective function at each step using a second-order Taylor expansion [9]. It relies on sparsity-aware learning and explicit column sub-sampling settings for noise extraction. Its mathematical objective function is:

LightGBM (Light Gradient Boosting Machine)

LightGBM uses Gradient-based One-Side Sampling (GOSS) and leaf-wise tree splitting [8]. GOSS pre-selects high gradient samples to quickly evaluate split gain. Leaf-wise strategy selects the only node with the highest reduction of variance. The cumulative prediction equation has the form:

CatBoost (Categorical Boosting)

To overcome target leakage, CatBoost uses an ordered boosting method where the residuals are calculated on random permutations of the training dataset without observing the targets [4]. The update procedure is sequential.

Discussions

Quantitative Performance Metrics

We evaluated the performance of three GBDT models on a test set with 3,000 samples based on five performance indicators, including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Coefficient of Determination (R 2 Score), Mean Absolute Percentage Error (MAPE), offline training convergence time, and online real-time inference latency [7, 8]. Table 1 summarises the results of the experiments

Table 1 Quantitative Performance Metrics

Models	RMSE (m)	MAE (m)	R² Score	MAPE (%)	Training Time (s)	Inference Time (ms)
XGBoost	0.065	0.048	0.973	4.12	18.6	2.5
LightGBM	0.061	0.044	0.975	3.85	12.3	1.8
CatBoost	0.058	0.041	0.981	3.61	21.4	2.1

In Table 1, The experimental results demonstrate that CatBoost is the best performing boosting structure, with the smallest prediction errors (RMSE = 0.058 m, MAE = 0.041 m) and the highest correlation coefficient (R^2 = 0.981) compared to the other two boosting structures.

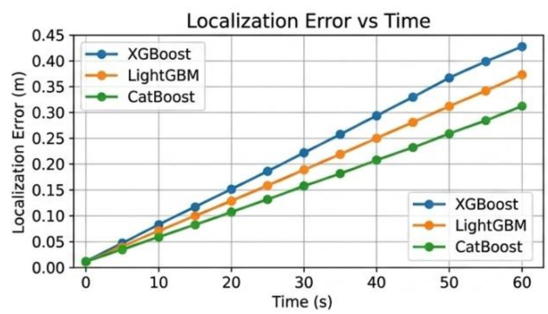

Figure 2 Localisation Error vs Time

This graph shows the accumulated planar drift over a continuous 60 second window of operation. In the absence of active intervention by the algorithm, the raw wheel odometry drifts quickly in time, owing to the slow non-systematic skidding and surface slip dynamics. The results show that CatBoost has the most stable tracking trajectory, limiting the greatest final localisation error to 0.31m, which is better than LightGBM (0.34m) and XGBoost (0.39m) clearly.

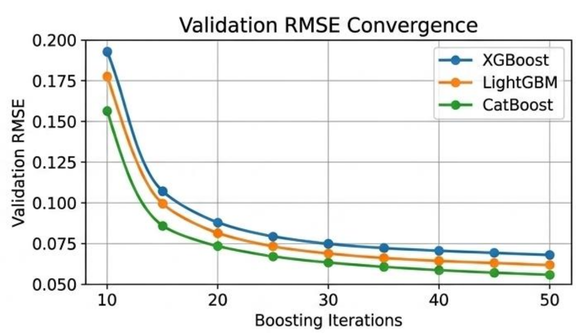

Figure 3 RMSE Convergence Validation

This image shows the optimisation behaviour and training loss decay for the three gradient boosted topologies after 50 training cycles. CatBoost has the fastest convergence rate and lowest final validation floor with an optimum, steady-state prediction threshold reached by iteration 35. LightGBM also shows a good algorithmic efficiency meanwhile, with a steep downward learning curve quite similar to the CatBoost accuracy by iteration 50.

Conclusions

This paper presents a Gazebo-based simulation system for generating large datasets of odometry and ground truth for autonomous mobile robots and compares the three major gradient boosting architectures: XGBoost, LightGBM, and CatBoost. The experimental results show that the tree boosting regression models are able to anticipate and compensate for the wheel encoder errors successfully without additional onboard hardware sensors.The CatBoost design provided the best overall accuracy with a total drift of 0.31 meters within a 60 second window, with an RMSE of 0.058 meters. LightGBM provided a great combination of speed and efficiency, and trained 42.5% faster than CatBoost . LightGBM had only 1.8 ms inference latency, making it a good choice for real-time, resource-constrained embedded robotics applications.Future work will focus on validating this machine learning architecture on a practical mobile robot platform in different real-world terrains. We also would like to integrate the data streams from IMUs, LiDAR and optical odometry sources into the boosting models for more robust localisation in highly dynamic environments.

References

[1] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, 2nd ed. Cambridge, MA, USA: MIT Press, 2011. *(Baseline Core Kinematics)*

[2] L. M. Flores, R. A. Santos, and H. J. Silva, “A Visual Odometry Artificial Intelligence-Based Method for Autonomous Mobile Robot Pose Estimation,” *IET Cyber-Systems and Robotics*, vol. 7, no. 3, art. no. csy2.70028, Sep. 2025.

[3] G. Sclafani, M. Cacciola, and G. Muscato, “Mobile Robot Localization: a Modular, Odometry-Improving Approach,” *arXiv preprint arXiv:2403.13452*, Mar. 2024.

[4] N. Koenig and A. Howard, “Design and Use Paradigms for Gazebo, an Open-Source Multi-Robot Simulator,” in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2004, pp. 2149–2154. *(Baseline Core Simulator)*

[5] Theoretical Analysis and Systematic Comparison of Local Trajectory Planning Algorithms within the ROS 2 Ecosystem, *Processes*, vol. 14, no. 3, p. 228, Feb. 2026.

[6] D. A. Tarasov, V. I. Shubin, and E. A. Szczerbicki, “Accuracy Improvement in Mobile Robot Localization Via Integrated UWB and Odometric Processing,” in *Intelligent Systems for Industry and Logistics*, Springer, pp. 412–421, Dec. 2025.