2021 The 21st International Conference on Control, Automation and Systems (ICCAS 2021)
Ramada Plaza Hotel, Jeju, Korea, Oct. 12∼15, 2021

Deep Learning based-State Estimation for Holonomic Mobile Robots Using Intrinsic Sensors

Dinh Van Nam[1] and Kim Gon-Woo[2*]

[1]Department of Control and Robot Engineering, Chungbuk National University,
Cheongju, 28644, South Korea, (quangnam.auto.tech@gmail.com, gwkim@cbnu.ac.kr)
[2]Department of Intelligent Systems and Robotics, Chungbuk National University,
Cheongju, 28644, South Korea, (quangnam.auto.tech@gmail.com, gwkim@cbnu.ac.kr) * Corresponding author

Abstract: State estimation is a fundamental component of the navigation system of autonomous mobile robots. Generally, the robot setup is equipped with intrinsic and extrinsic sensors. The state estimators have relied almost on intrinsic sensors such as wheel encoders and inertial measurement units in textureless and structureless environments. This paper will analyze and propose the learning state estimation frameworks for the dead-reckoning of autonomous holonomic vehicles based only on intrinsic sensors. First, we review and categories the intrinsic-only estimation problem. Second, we describe the problem formulation using learning-based techniques. Next, the learning inertial-only estimation is presented with several strategies using the deep learning technique. The initial experiment results are analyzed and deployed using a holonomic mobile robot in real-world environments.

Keywords: deep learning based-state estimation, inertial navigation system, inertial-only estimation, multi-sensor fusion.

1. INTRODUCTION

To deal with the self-navigation problem [1], intelligent vehicles such as self-driving cars [2], automated guided vehicles [3] needs to know their position in the environment. In general, the autonomous mobile robot using proprioceptive and exteroceptive sensors to fuse the sensor's information to provide reliable states. The Global Positioning System (GPS) is the most crucial sensor for long-term navigation for outdoor applications [1] [4]. However, the GPS-based estimator usually jumps with significant error in GPS-denied environments such as under-bridges, tunnels, or high buildings [1] Furthermore, GPS is disputed for indoor applications such as industrial factories, underwater [1].

For accurate estimation, the robot is equipped with extrinsic sensors such as the Light Detection and Ranging (LiDARs), Radars, and cameras that can perceive its surrounding environments [1] [4]. The robot is also implemented with intrinsic sensors such as the wheel encoders and the Inertial Measurement Unit (IMU) sensors. Nevertheless, the LiDARs and radar sensors cannot operate well in structureless environments like repetition walls in long corridors. Moreover, in textureless environments, visual sensors that are usually sensitive to light illumination and light condition fail to observe visual features [1] [4]. In contrast, intrinsic sensors can provide good accuracy in short-term navigation because they are immune to their surroundings [5]. Still, the wheel encoders or inertial sensors are drifted over time and cannot be corrected by using themself. Generally, the wheel encoders and inertial sensors provide higher frequency about 10 to 20 times than LiDAR and visual sensors [6] [5]. Moreover, inertial sensors can operate in 3D environments and give better results than wheel encoders in rotation drift scenarios [6] [5]. Therefore, we need technologies to handle the state estimation for mobile robots in extrinsic-denied environments.

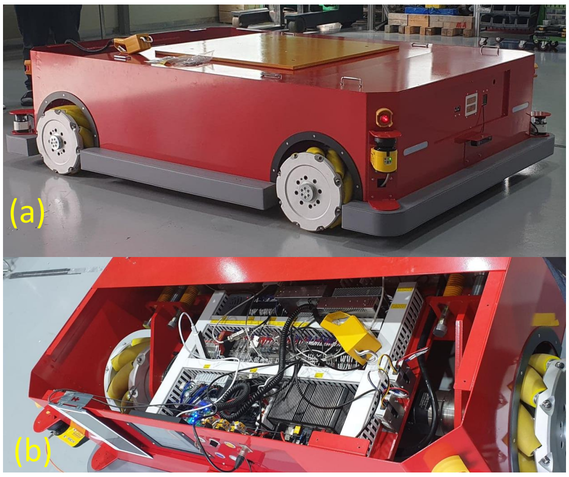

Fig. 1. The four wheels holonomic mobile robot platform in our experiments. (a) presents the overall mechanical structure; (b) describes the electronic hardware.

In this paper, we present the learning strategies to deal with the robot localization in the outage environments of the exteroceptive sensors. The comprehensive platform of a holonomic mobile robot is shown in Figure 1.

2. RELATED WORKS

The navigation system of autonomous mobile robots operates multi-sensor fusion consists of LiDARs, radars, cameras, odometers and IMUs [1] [7] [4]. Recently, Simultaneous Localization And Mapping (SLAM) [1] has been an advantage technique for state estimation based on the multi-sensor fusion of camera, LiDAR, and inertial sensor. SLAM has been developed for real-time estima-

978-89-93215-21-2/21/$31.00 ©ICROS

tion systems for two decades[11]. The Bayes filtering inference is leveraged to maximum the posterior state probability given the initial state and sensor measurements to fuse multiple sensor information [16].

The fusion technique can be categorized into filtering-based and optimization-based [7] [4]. Extended Kalman Filter (EKF), unscented KF (UKF), particle filtering are popular methods with two-phase prediction and correction. The most high-performance visual-inertial navigation system was based on multi-state constraints EKF, which applied the tightly coupled approach for real-time application [8] [9]. The optimization-based method handles the state and sensor information for a long time, providing better accuracy than filtering-based [7][4]. VIN-mono [23] is an excellent open-source visual-inertial state estimation system using factor graph optimization (FGO). LIO-SAM [10] is an accurate estimator using the tightly coupled LiDAR inertial. LVI-SAM [11] has recently been introduced for the tightly coupled of 3D LiDAR, camera, and IMU via smoothing localization.

Artificial intelligence (AI) based- approach can be considered the third class of multi-sensor fusion. AI is adopted to learn the uncertainty parameters, which are then joined to the traditional fusion techniques [6] [24] [25] [22]. AI can also be applied immediately to determine the prediction state joined to the fusion techniques [12] [15]. Moreover, AI can only use the end-to-end approach to estimate the robot state without MAP inference [13] [27] [14]. To achieve the best performance for state estimation based on the intrinsic sensor -only, we follow the AI-based technique.

3. METHODOLOGY

3.1 IMU and wheel encoders model

In general, a modern IMU sensor can directly provide 9 DoF outputs [4][9][7] which are 3D angular velocity vector $\tilde{\boldsymbol{\omega}}_b(t)$, 3D acceleration vector $\tilde{\mathbf{a}}^b(t)$ with respect to body coordinate b and 3D orientation of IMU to w coordinate, as follows [7] [4],

$$\begin{array}{l}\tilde{\boldsymbol{\omega}}_b(t) = \boldsymbol{\omega}_b(t) + \mathbf{b}_g(t) + \boldsymbol{\eta}_g(t) \\ \tilde{\mathbf{a}}_b(t) = {}^b\mathbf{R}_w(\mathbf{a}_w(t) + \mathbf{g}_b(t)) + \mathbf{b}_a(t) + \boldsymbol{\eta}_a(t)\end{array}, \quad (1)$$

where $\mathbf{b}_g(t)$, $\mathbf{b}_a(t)$ are quasi-constant biases, $\boldsymbol{\eta}_g \triangleq \mathrm{N}(\mathbf{0}, \sigma_g^2)$, $\boldsymbol{\eta}_a \triangleq \mathrm{N}(\mathbf{0}, \sigma_a^2)$ are zero-mean Gaussian noises, ${}^b\mathbf{R}_w \in SO(3)$ is the rotation transforming from frame w to b [20]. Herein, the biases are given following the random walk process [7][8][20].

Following the kinetic model of holonomic robot, we can compute the instance robot velocity as [3],

$$\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = \frac{\pi\rho}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ -1/d & 1/d & -1/d & 1/d \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix}, \quad (2)$$

where $\mathbf{v}_b(t), \boldsymbol{\omega}_{bw}(t) = {}^3\vec{\mathbf{e}}_2(v_x, v_y, \omega)^{\mathrm{T}}$ is the current velocity of the robot on SE(3), ${}^3\vec{\mathbf{e}}_2$ indicates the transformation from SE(2) to SE(3) [29], ω_i is the angular velocity of wheel i, ρ is the radius of the wheel, and d is the mechanical size of the robot [3].

3.2 Problem formulation

Let $\mathbf{x}$ is a sequence state, $\mathbf{z}$ is the observation, and $\mathbf{u}$ is the control inputs of the robot. The problem is that we need to determine robot states $\mathbf{x}_t$ given $\mathbf{z}_{1:t}$ and $\mathbf{u}_{1:t}$ aligned with observation model uncertainty. We denote distribution $\psi(\mathbf{x}_t) = p(\mathbf{x}_t|\mathbf{u}_{1:t}, \mathbf{z}_{1:t})$ is the belief state of the robot at time t. Using the Bayes filtering method [16], we can find the optimal solution to the problem. The prediction step uses the robot motion model $p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_{t-1})$ as follows,

$$\psi_{pre}(\mathbf{x}_t) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_t)\psi(\mathbf{x}_{t-1})d\mathbf{x}_{t-1} \quad (3)$$

Then the observation is used to update the prediction state as,

$$\psi(\mathbf{x}_t) = \eta p(\mathbf{z}_t|\mathbf{x}_t)\psi_{pre}(\mathbf{x}_t) \quad (4)$$

We assume that all the uncertainties are represented as Gaussian distribution. The most successful stories of Bayes filters are filtering-based methods such as EKF, UKF, PF [16] and factor graph optimization (FGO) [28]. Noted that, AI-based method can learn the motion model (3) and measurement model (4) as the end-to-end learning process. Moreover, we can employ AI to learn the uncertainty that will implement in Bayes filters. We study the learning-based EKF and FGO based-state estimation.

Here, an AI technique is used to model the learning problem as,

$$\mathbf{y} = g_{\mathrm{AI}}(\boldsymbol{\Psi}, \boldsymbol{\theta}) \quad (5)$$

where $g(.)$ is the AI solution such as CNN, LSTM, etc.; $\boldsymbol{\theta}$ is the trainable weights of the AI solution; $\mathbf{y}$ is the Bayes filtering parameter; $\boldsymbol{\Psi}$ is the input of the AI technique such as the sensors data, previous states.

To handle intrinsic sensors data, we employ a Neural Network (NN) as input of a sliding window of W inertial measurements and wheel odometers as,

$$\chi_{k+1,1} = \psi_1(\alpha\{\boldsymbol{\omega}_i, \mathbf{a}_i, \mathbf{v}_i^b, \boldsymbol{\omega}_i^b\}_{i=k-W+1:k}^W) \quad (6)$$

$$\chi_{k+1,2} = \psi_2(\beta\{\boldsymbol{\omega}_i, \mathbf{a}_i, \mathbf{v}_i^b, \boldsymbol{\omega}_i^b\}_{i=k-W+1:k}^W) \quad (7)$$

where ψ_1 and ψ_2 are a neural network and LSTM, respectively; α is the stack function that converts all the input information into a vector; β denotes the sequence input into an LSTM; Outputs $\chi_{(.)}$ is the Bayes filter parameters, motion model, or measurement model using NN or LSTM.

4. EXPERIMENT EVALUATION

We evaluate a sensor setup consists of two stereo cameras, 04 2D LiDARs, 04 wheel encoders, and an IMU, as shown in Fig. 1. The data is recorded from IMU and wheel encoder with 06 Sections. We use multi-sensor fusion technique [18] to build the ground truth and learning the observation model from its estimation. A sample sensor data in section 3 is shown in Figure 2.

4.1 Learning LSTM and neural networks for measurement model

We will analyze the LSTM and deep neural network for end-to-end learning measurement model following [13] [6]. For LSTM, the input is a sequence of vector $\{\boldsymbol{\omega}_i, \mathbf{a}_i, \mathbf{v}_i^b, \boldsymbol{\omega}_i^b\}$ (6). The outputs are a 2-vector $[d; \theta]$ in a 10-steps sliding window, where $d = distance(se2(\mathbf{x}_{k+10} - \mathbf{x}_k))$, $\theta = abs(angle(se2(\mathbf{x}_{k+10} - \mathbf{x}_k)))$. We create an LSTM of two hidden layers as follows [13],
layers = [Input(9);
bilstmLayer(128,'sequence');
bilstmLayer(64,'last');
fullyConnectedLayer(32);
fullyConnectedLayer(2);
regressionLayer];
We use adam algorithm with 1024 epochs for training. After training, the accuracy is 8.73×10^{-3} MSE. Similarly, we design a deep neural network with two hidden networks (7) as,
imitateNetwork = [
featureInputLayer(99)
fullyConnectedLayer(128)
sigmoidLayer('relu')
fullyConnectedLayer(128)
sigmoidLayer('relu')
fullyConnectedLayer(2)
regressionLayer()
];
The accuracy of the deep neural network is 0.01 MSE. Although we increase the layers of LSTM and DNN, the results for the end-to-end measurement model did not get high accuracy for the holonomic mobile robot, as following IoNet [13] and AI-IMU [6]. For the end-to-end method, the results conclude that the LSTM model is better than DNN. However, when we increase the number of neural for deep learning to 2048 as, *imitateNetwork = [*
featureInputLayer(99)
fullyConnectedLayer(1024)
sigmoidLayer('relu')
fullyConnectedLayer(256)
sigmoidLayer('relu')
fullyConnectedLayer(2)
regressionLayer()
];
The results of DNN about 0.005 is more better than LSTM. Next, we will examine the learning uncertainty model for Bayes filers.

4.2 Learning uncertainty model for Bayes filers

The end-to-end solution for the holonomic robot is not working well. Therefore, the learning uncertainty is implemented for Bayes filtering. The input for a network is similar aforementioned. The outputs are $\Delta\mathbf{p} = [\Delta d; \Delta\theta]$, where Δd is the error of the distance $\Delta\mathbf{x}$ to ground-truth, $\Delta\theta$ is also the error rotation $\Delta\theta$ to ground-truth. Although the mean squared error (MSE) of LSTM is about 0.01, the actual response posses poor accuracy. The deep learning algorithm is trained on a desktop PC with an Intel 4-core i7-7700 processor CPU and GPU GTX 3070 Ti 8Gb. Next, we design a neural network with two layers (01 hidden layer with 30 neutrals) using Levenberg–Marquardt algorithm. The results are surprisingly good that provides a 0.005 MSE error for training. Flatten neural network is better than LSTM in the case of learning uncertainty. After training, we will implement the deep neural network into the Bayes filters technique, as shown earlier. Following the type of deep learning accuracy, we will select the flattened neural network for learning uncertainty covariance. In contrast, for the learning uncertainty method, the results conclude that the flattened neural network is better than LSTM, as shown in Fig. 3.

5. CONCLUSION

This paper studies deep neural network based- state estimation using intrinsic sensors for a holonomic mobile robot. We first provide valuable discussion and opportunities for not only intrinsic sensors fusion but also extrinsic sensors merging. We then show how to connect Bayes filtering to the neural network using multiple sensor data information. The initial experiment implemented the LSTM, deep neural networks, and multilayer perceptron (MLP) to analyze the results. In future work, we will implement the proposed system using specific state estimation techniques such as EFK, PF, FGO. We will also conduct more neural network architecture for evaluation experiments.

ACKNOWLEDGEMENT

This research was financially supported in part by the Ministry of Trade, Industry and Energy(MOTIE) and Korea Institute for Advancement of Technology(KIAT) through the International Cooperative R&D program. (Project No.P0004631) and in part by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (IITP-2020-0-00211, Development of 5G based swarm autonomous logistics transport technology for smart post office)

REFERENCES

[1] C. Cadena et al., "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," in *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309-1332, Dec. 2016. doi: 10.1109/TRO.2016.2624754.

[2] Van, N.D et al. "A Hierarchical Control System for

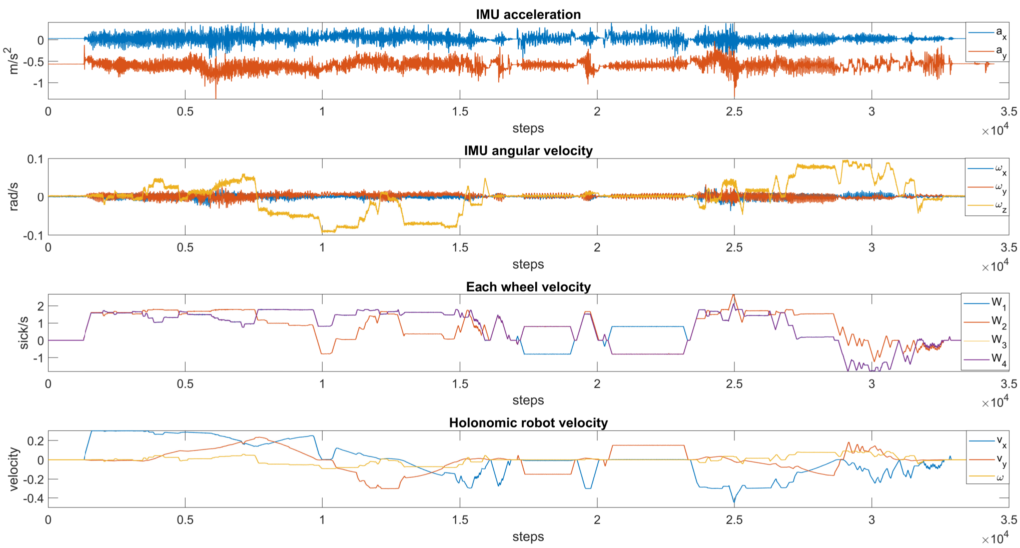

Fig. 2. A sample dataset was recorded in the real experiment.

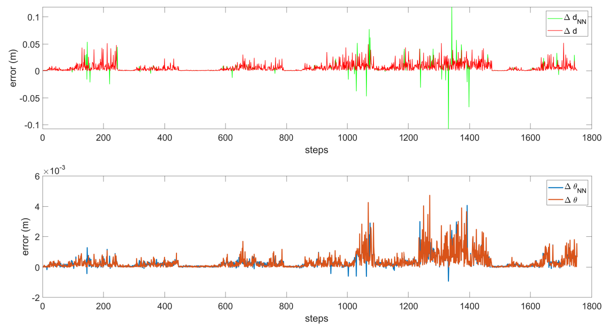

Fig. 3. The training uncertainty process employs the flattened neural network of about 0.005 MSE.

Autonomous Driving towards Urban Challenges, " *Appl. Sci.* 2020, 10, 3543.

[3] Qian, Jun, et al. "The design and development of an omni-directional mobile robot oriented to an intelligent manufacturing system," *Sensors* 17.9 (2017): 2073.

[4] N. V. Dinh and G. Kim, "Multi-sensor Fusion Towards VINS: A Concise Tutorial, Survey, Framework and Challenges," *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Busan, Korea (South), 2020, pp. 459-462.

[5] Chen, Changhao, et al. "Selective sensor fusion for neural visual-inertial odometry," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2019.

[6] Brossard, Martin, Axel Barrau, and Silvère Bonnabel. "AI-IMU dead-reckoning," *IEEE Transactions on Intelligent Vehicles* 5.4 (2020): 585-595.

[7] Huang G, "Visual-inertial navigation: A concise review, " In: *IEEE Int. Conf. Robot. Autom.(ICRA)*, 2019.

[8] Patrick Geneva, el al.,"OpenVINS: A Research Platform for Visual-Inertial Estimation," IROS 2019 Workshop on Visual-Inertial Navigation: Challenges and Applications.

[9] Nam, D.V.; Gon-Woo, "K. Robust Stereo Visual Inertial Navigation System Based on Multi-Stage Outlier Removal in Dynamic Environments," *Sensors* 2020, 20, 2922.

[10] Shan, Tixiao, et al. "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020.

[11] Shan, Tixiao, et al. "LVI-SAM: Tightly-coupled Lidar-Visual-Inertial Odometry via Smoothing and Mapping," *arXiv preprint* arXiv:2104.10831 (2021).

[12] Lee, Michelle A., et al. "Multimodal sensor fusion with differentiable filters.," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020.

[13] Chen, Changhao, et al. "Ionet: Learning to cure the curse of drift in inertial odometry," *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. No. 1. 2018.

[14] Zhao, Xiangrui, et al. "Learning to Compensate for the Drift and Error of Gyroscope in Vehicle Localization," *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2020.

[15] Yi, Brent, et al. "Differentiable Factor Graph Optimization for Learning Smoothers," *arXiv preprint* arXiv:2105.08257 (2021).

[16] Barfoot, Timothy D. State estimation for robotics. *Cambridge University Press*, 2017.

[17] Sola, Joan, et al. "A micro Lie theory for state estimation in robotics," *arXiv preprint* arXiv:1812.01537 (2018).

[18] Labbé, M, Michaud, F. "RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation," *J Field Robotics*. 2019; 35: 416–446. https://doi.org/10.1002/rob.21831

[19] Kailai Li, Meng Li and Uwe D. Hanebeck.

"Towards High-Performance Solid-State-LiDAR-Inertial Odometry and Mapping," *arXiv:2010.13150*. 2020.

[20] Forster C, et al., "On-manifold preintegration for real-time visual-inertial odometry," *IEEE Trans Robot* 33(1):1–21, DOI 10.1109/TRO.2016.2597321, 2017.

[21] Frank Dellaert and Michael Kaess, "Factor Graphs for Robot Perception," *Foundations and Trends in Robotics*, Vol. 6, No. 1-2, pp. 1–139, 2017

[22] Van Nam, Dinh, and Kim Gon-Woo. "Online Self-Calibration of Multiple 2D LiDARs using Line Features with Fuzzy Adaptive Covariance," *IEEE Sensors Journal*, vol. 21, no. 12, pp. 13714-13726, 15 June15, 2021, doi: 10.1109/JSEN.2021.3053260.

[23] Qin, Tong, et al. "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics* 34.4 (2018): 1004-1020.

[24] Liu, Wenxin, et al. "TLIO: Tight learned inertial odometry, " IEEE Robotics and Automation Letters 5.4 (2020): 5653-5660.

[25] Brossard, Martin, et al. "RINS-W: Robust inertial navigation system on wheels,"*2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019.

[26] Haarnoja, Tuomas, et al. "Backprop kf: Learning discriminative deterministic state estimators," *Advances in neural information processing systems*. 2016.

[27] Herath, Sachini, et al. "RoNIN: Robust Neural Inertial Navigation in the Wild: Benchmark, Evaluations, & New Methods," *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020.

[28] Dellaert, Frank, and Michael Kaess. "Factor graphs for robot perception," *Foundations and Trends® in Robotics* 6.1-2 (2017): 1-139.

[29] Geneva, Patrick, et al. "Versatile 3d multi-sensor fusion for lightweight 2d localization," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020.

[30] R. Jonschkowski, et al.,"Differentiable Particle Filters: End-to-End Learning with Algorithmic Priors," *in Proceedings of Robotics: Science and Systems (RSS)*, 2018.