Improving the robustness of Local_INN: Invertible Neural Networks

Jason Xie
University of Pennsylvania
Philadelphia, US
jchunx@seas.upenn.edu

Rohit Bhikule
University of Pennsylvania
Philadelphia, US
rbhikule@seas.upenn.edu

Jimmy Zhang
University of Pennsylvania
Philadelphia, US
zmz@seas.upenn.edu

Patrick Paglia
University of Pennsylvania
Philadelphia, US
ppags@seas.upenn.edu

***Abstract*—For the ESE 6150 final project, our objective was implementing INN onto the F1 Tenth car and having it work reliably. The robustness of INN was evaluated during a laps around Levine Lobby, which was published to data that we viewed on Foxglove. We looked for things like the localization of our vehicle. Localization is a crucial aspect of autonomous navigation, and Invertible Neural Networks (INNs) have proven effective in addressing ambiguous inverse problems. However, implementing Zuirui's INN framework revealed a vulnerability concerning latency issues, leading to localization failures at higher speeds, particularly in long corridors. To improve the framework's performance, we propose a two-fold strategy. First, we suggest incorporating accelerometer data alongside speedometer data to enhance motion prediction accuracy during high-speed travel. Second, we recommend refining the framework's handling of extended lateral LiDAR readings through adjustments in the correlation mechanism or the integration of additional sensors. These enhancements aim to optimize the INN framework, enabling more accurate and reliable robot localization, even under challenging conditions or at higher speeds. Improved localization performance contributes to the development of safer and more efficient autonomous navigation systems.**

***Index Terms*—INN(Invertible Neural Network), UKF, Sensor fusion, SLAM**

I. Introduction

Robot localization involves the task of determining the position and orientation (pose) of a robot by utilizing a map and sensor measurements, such as data obtained from LiDAR scans. It is crucial for a robot to accurately perceive and interact with its physical surroundings. However, accurately establishing a one-to-one correspondence between sensor measurements and robot poses is challenging due to the potential similarities between measurements obtained from different locations.

Robot localization can be categorized as an inverse problem, where the objective is to determine the underlying causal factors based on a set of observations. In the case of LiDAR-based localization, the robot's pose in the environment is the causal factor that produces specific scan measurements. Furthermore, if a map is available, it becomes relatively straightforward to simulate LiDAR scans from any given pose on the map.

Invertible neural networks, such as normalizing flows, have proven effective in solving inverse problems across various domains. These networks learn a bijective mapping between source and target distributions through a series of invertible transformations. By utilizing a latent space, they capture the inherent ambiguous information present in the training data. In our specific case of robot localization, we employ pose-scan data pairs to train such a bijective mapping. The forward path of the network maps from poses to scans, while the reverse path maps from scans to poses. To ensure compatibility between input and output dimensions, we employ a Variational Autoencoder (VAE) to reduce the dimensionality of lidar scans, and Positional Encoding techniques to augment the dimensionality of poses. Conditional inputs, such as zones calculated from the robot's previous pose within the map, are used to reduce the ambiguity associated with the inverse problem. These conditional inputs are incorporated into the invertible neural network architecture to improve the accuracy and robustness of the localization process.

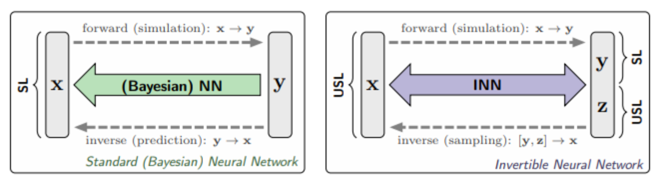

Fig. 1. Graph comparing standard neural networks and Invertible Networks

INNs have the capability of being used to solve an inverse problem, which has been utilized in our case to localize the F1Tenth car. INN can be a superior, drop-in replacement to particle filtering due to its high localization frequency. A comprehensive assessment approach for INN will enable us to conduct a comparative analysis with its rival technologies, thereby facilitating the identification of areas that require enhancement. Through this evaluation, we can refine the local INN implementation, ensuring its performance aligns with those of alternative models, as INN offers a more cost-effective and time-efficient solution compared to other localization techniques. We find Zirui's INN framework to show promising results in resolving ambiguous inverse problems. However, upon deeper investigation, we found that this issue occurs more frequently when the vehicle navigates through long corridors. In such scenarios, the lateral LiDAR readings become extended, causing the framework's correlation to fail and consequently compromising the localization accuracy. To address this shortcoming and improve the localization performance, we propose a two-fold strategy. First, we suggest incorporating not only the speedometer data but also accelerometer data into the framework. By combining these two sources of information, the system can more accurately account for the vehicle's motion and better predict its position, particularly when it is traveling at higher speeds. Second, we recommend refining the framework's ability to handle extended lateral LiDAR readings, which could involve adjusting the correlation mechanism or incorporating additional sensors to provide a more comprehensive understanding of the environment. By implementing these enhancements, we believe that Zuirui's INN framework can be further optimized, resulting in more accurate and reliable robot localization even under challenging conditions or at higher speeds. This improved localization performance will ultimately contribute to the development of safer and more efficient autonomous navigation systems. This paper will cover the our work in implementing Invertible Neural Networks (INNs) to create a compatible ROS package that can be used for F1 tenth racing.

II. Motivation

The motivation for this research paper stems from the critical role of robot localization in autonomous navigation and the increasing adoption of Invertible Neural Networks (INNs) for addressing localization challenges. Zirui's INN framework has shown promise in resolving ambiguous inverse problems with low latency, making it an attractive choice for localization tasks. However, during implementation on a test robotic vehicle, a vulnerability was identified: the framework's localization estimation fails as the vehicle's speed increases, leading to potential navigation errors and safety concerns.

In recent years, Invertible Neural Networks (INNs) have emerged as a promising alternative to particle filtering for localization tasks. One key advantage of INNs is their ability to operate at high localization frequencies, making them a superior choice for applications that require real-time and continuous pose estimation.

To gain a comprehensive understanding of the capabilities of INNs, a rigorous assessment approach is necessary. This approach would involve conducting a thorough comparative analysis of INNs against other rival technologies commonly used for localization. By comparing the performance, accuracy, and efficiency of INNs with alternative models, we can identify areas where INNs excel and areas that may require further enhancement.

Through this evaluation, we aim to refine the implementation of local INNs, ensuring that its performance aligns with or surpasses that of alternative models. By refining the INN framework, we can unlock its full potential as a cost-effective and time-efficient solution for localization tasks. This is of significant importance as it enables the adoption of INNs in a wide range of applications, benefiting industries such as autonomous vehicles, robotics, and navigation systems. This is where we gained motivation to work on Invertible neural networks as a localization tool.

III. Implementation

We use the Unscented Kalman Filter to estimate position, velocity, acceleration, yaw, IMU bias. See Figure 2 on the psuedocode for this process.

Kalman Filter	Unscented Kalman Filter
	$\mathcal{Y} = f(\chi)$
$\bar{x} = Fx$	$\bar{x} = \sum w^m \mathcal{Y}$
$\bar{P} = FPF^T + Q$	$\bar{P} = \sum w^c (\mathcal{Y} - \bar{x})(\mathcal{Y} - \bar{x})^T + Q$
	$\mathcal{Z} = h(\mathcal{Y})$
	$\mu_z = \sum w^m \mathcal{Z}$
$y = z - Hx$	$y = z - \mu_z$
$S = H\bar{P}H^T + R$	$P_z = \sum w^c (\mathcal{Z} - \mu_z)(\mathcal{Z} - \mu_z)^T + R$
$K = \bar{P}H^T S^{-1}$	$K = \left[\sum w^c (\mathcal{Y} - \bar{x})(\mathcal{Z} - \mu_z)^T\right] P_z^{-1}$
$x = \bar{x} + Ky$	$x = \bar{x} + Ky$
$P = (I - KH)\bar{P}$	$P = \bar{P} - KP_zK^T$

10.6 Van der Merwe's Scaled Sigma Point Algorithm

Fig. 2. psuedo code for unscented kalman filter

The filter propagates future states using wheel odometry data and nonlinear transformations. Two Kalman separate updates are used: one using pose from Local INN, and one using IMU accelerometer data. In our implementation for better prediction for the Local INN, we incorporate the IMU accelerometer data along with the Local INN updates to give a better estimate of the state using an Unscented kalman filter. Below is the flow model of the way we design our model.

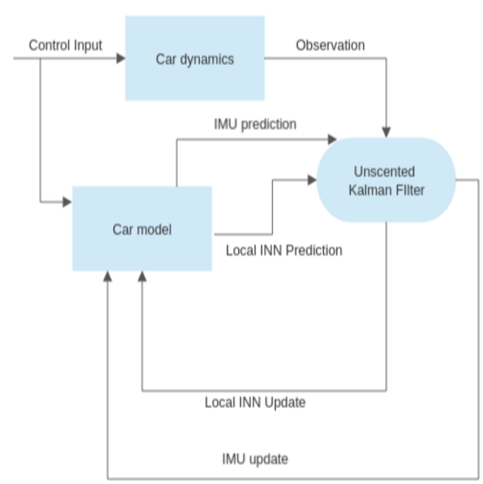

Fig. 3. Sensor fusion model

To enhance the localization performance, we incoporate the UKF to refine the pose estimates based on the fusion of laser scan data and IMU updates. When an IMU update is available, the pose is adjusted using the IMU estimate, which helps compensate for biases and errors in the IMU measurements. Furthermore, when both laser scan and IMU observations are available sequentially, the framework utilizes a sequential update mechanism to optimize the pose estimation. This sequential fusion of observations enables a more comprehensive and accurate estimation of the car's pose.

A. Equations

We define q as follows:

$$q = \begin{bmatrix} x \\ y \end{bmatrix}_{2\times 1}$$

To propagate the dynamics, we define the dynamic model as follows:

$$X_k = \begin{bmatrix} q \\ \dot{q} \\ \ddot{q} \\ \theta \end{bmatrix}_{7\times 1}$$

Here, X_k is the state vector at particular timestep k.
The control inputs for the model are

$$u = \begin{bmatrix} v \\ \omega \end{bmatrix}_{2\times 1}$$

The state vector dynamic model is defined as,

$$X_{k+1} = AX_k + Bu$$

where,

$$A = \begin{bmatrix} \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 2} & \frac{\Delta t^2}{2}\,\mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times 2} & \Delta t\,\mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times 2} & \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{1\times 2} & \mathbf{0}_{1\times 2} & \mathbf{0}_{1\times 2} & 1 \end{bmatrix}_{7\times 7}$$

$$B = \begin{bmatrix} \Delta t\cos\theta & 0 \\ \Delta t\sin\theta & 0 \\ \mathbf{0}_{2\times 1} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{2\times 1} & \mathbf{0}_{2\times 1} \\ 0 & \Delta t \end{bmatrix}_{7\times 2}$$

We also include the bias of the accelerometer from the IMU in our dynamic propagation model. We augment the state vectors as follows.

$$X_k = \begin{bmatrix} q \\ \dot{q} \\ \ddot{q} \\ \theta \\ b_x \\ b_y \end{bmatrix}_{9\times 1}$$

With the augmented state vectors, the new dynamic model becomes,

$$X_{k+1} = \begin{bmatrix} \mathbf{A}_{7\times 7} & \mathbf{0}_{7\times 2} \\ \mathbf{0}_{2\times 7} & \mathbf{I}_{2\times 2} \end{bmatrix} X_k + \begin{bmatrix} \mathbf{B}_{7\times 2} \\ \mathbf{0}_{2\times 2} \end{bmatrix} u$$

To update the state vector, we have the observation model for the IMU as follows:

$$y = \begin{bmatrix} a_x \\ a_y \end{bmatrix}$$

$$y = \begin{bmatrix} \mathbf{0}_{2\times 7} & \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} & \mathbf{I}_{2\times 2} \end{bmatrix}_{2\times 9} X$$

To update the state vector, we have the observation model for the Local INN as follows:

$$y = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$$

$$y = \begin{bmatrix} \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 5} & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{1\times 6} & 1 & \mathbf{0}_{2\times 2} \end{bmatrix}_{3\times 9} X$$

We use the above equations to propagate our model.

IV. Experiments

Initially, our pose estimation relied solely on the observations generated by the Local INN network using laser scans. However, we encountered a significant challenge during experimentation: the estimated pose exhibited erratic behavior, particularly on straight sections of the map. This issue arose because the Levine map corridor featured similar visual features along these straight patches, leading the localization to identify two different locations with similar lidar scans. The consequence was a pose estimate that appeared to jump erratically across the map.

To address this problematic jumping behavior, we turned to the Extended Kalman Filter (EKF), which had already been implemented by Zirui. Unfortunately, even with the EKF in place, the accuracy of the localization remained suboptimal. This subpar performance could be attributed to the highly non-linear nature of the observations, which posed a challenge for the EKF's linearization assumptions.

In response, we sought to improve the pose tracking accuracy by implementing a more suitable alternative: the Unscented Kalman Filter (UKF). The UKF allows for non-linear modeling of the observations, making it better suited to handle the complex relationships between the IMU measurements, Local INN observations, and the actual pose. By incorporating both the IMU data and the Local INN observations into the UKF-based pose tracking model, we aimed to refine the localization process and achieve more accurate and reliable pose estimates.

By employing the UKF and integrating information from both the IMU and the Local INN, we anticipated a significant improvement in the accuracy and robustness of the pose estimation. This approach accounted for the non-linear nature of the observations and leveraged the complementary information from the different sensor modalities, thereby mitigating the challenges encountered with the previous methods.

We demonstrate the improved robustness of the localization by running Local INN at Levine for multiple laps without losing tracking in the following video: Video.

We believe that we overcame the initial issue of INN breaking at locations with no distinguishable features based on our final run of INN in the Levine track.

The proper implementation of Unscented Kalman Filter allowed our estimates to be a lot more accurate for localization and updated quickly as our car moved through the halls of Levine!

V. Results

As seen in Figure 4, our team faced difficulties with certain spots of the Levine track. Based on the previous discussion in, we believe that we faced these difficulties for a couple different reasons: Certain portions of Levine track are not distinguishable from the other due to the similar scans along the corridor. In addition, the UKF was also not active during the update stage.

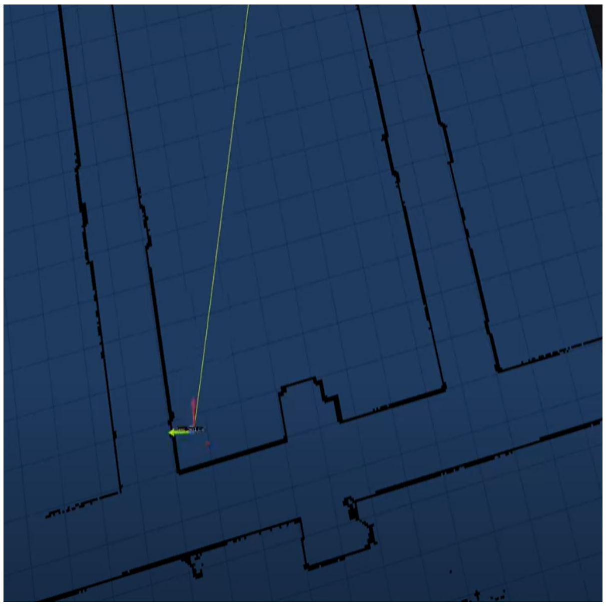

Fig. 4. Figure of our vehicle being out of bounds

The Local INN, although capable of generating plausible poses based on laser scans, exhibited inconsistency in its outputs. Figure 2 illustrates the range of plausible outputs obtained from the Local INN. It became evident that the existing Extended Kalman Filter (EKF) solution was inadequate for racing scenarios, as it lost localization at the Levine map after just one lap.

Figure 5 provides visual evidence of the issue: the poses estimated by the Local INN along the straight corridor were spread out due to the similarity of laser scans. Multiple poses were observed along the straight patch, leading to unreliable localization results. To address this challenge, we integrated observations from the Inertial Measurement Unit (IMU) into the pose estimation process.

By incorporating IMU data, our UKF-based model significantly improved the robustness of pose estimation. As shown in the results, the pose estimates obtained with the combined IMU and Local INN observations were no longer spread along the straight patch but rather concentrated in a more consistent and accurate manner. Notably, the UKF-based model successfully estimated the car's pose consistently over three laps, as demonstrated by its reliability in the racing environment.

This improvement in the robustness of the Local INN pose estimation, achieved through the incorporation of IMU observations within the UKF framework, represents a significant advancement. It addressed the inconsistency and spread of poses along the straight corridor, enhancing the reliability and accuracy of the pose estimation process.

Overall, our findings indicate that the UKF-based model, incorporating IMU observations, yielded substantial improvements in the robustness of pose estimation compared to the Local INN alone. The consistent pose estimation achieved over multiple laps in a racing environment signifies a noteworthy enhancement in the reliability of the localization system.

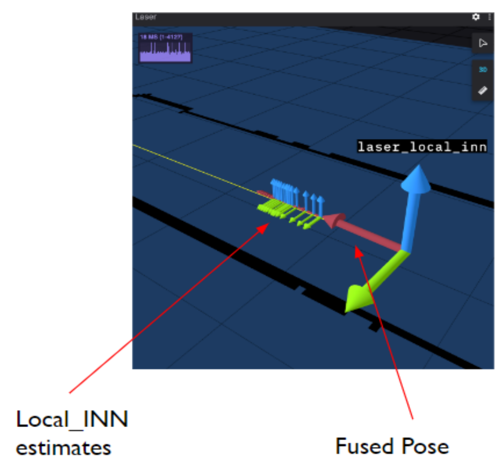

Fig. 5. Image of plausible poses output

VI. Conclusion

Ultimately, we were able to combine many of the topics we studied this semester in a practical and challenging setting. One of the most interesting parts of this final project was seeing the difference in behavior of the environment between simulation and hardware testing that we had been observing all semester in the context of our project. Something that worked well for our team right from the beginning was testing our initial assumptions early on on Levine track. From our lap we observed how different the car behaved on Foxglove when compared to real life.

VII. Future Work

Given additional time for testing and experimentation, we would have expanded our evaluation of INNs in several ways. Firstly, we would have sought to test the INN implementation on different tracks or environments, as the challenges we encountered in Levine Hall highlighted the difficulty in distinguishing certain parts of the map. By exploring different tracks, we could assess the generalizability and adaptability of the INN approach to various real-world scenarios.

Additionally, we would have explored the potential impact of altering the network architecture of the Local INN. This investigation would involve experimenting with different model architectures, such as varying the number of layers, adjusting the size of hidden units, or exploring alternative activation functions. By modifying the network architecture, we could observe the effects on pose estimation accuracy, consistency, and robustness. This analysis would provide insights into the sensitivity of the INN framework to architectural choices and guide future improvements.

Furthermore, to gain a comprehensive understanding of INN's performance, we would have conducted a comparative analysis with alternative techniques such as ORB SLAM 3. By comparing INNs with other state-of-the-art localization methods, we could assess their respective strengths, weaknesses, and applicability in different contexts. This comparative analysis would contribute to the broader field of localization research by providing insights into the relative performance and suitability of different techniques.

In summary, given more time for testing and exploration, we would have expanded the scope of our evaluation by testing INNs on different tracks, investigating variations in network architecture, and conducting comparative analyses with alternative localization techniques. These efforts would have provided a more comprehensive assessment of INNs and facilitated insights into their performance, limitations, and potential areas for further enhancement.

Acknowledgment

"ESE 6150 teaching team. thanks...". We would also like to thank Prof. Dr. Rahul Mangharam and Zirui for giving us the wonderful opportunity to work on Local INNs.

References

[1] Ardizzone, Lynton, et al. 'Analyzing Inverse Problems with Invertible Neural Networks'. CoRR, vol. abs/1808.04730, 2018, http://arxiv.org/abs/1808.04730.

[2] Zang, Zirui, et al. 'Local_INN: Implicit Map Representation and Localization with Invertible Neural Networks'. ArXiv [Cs.RO], 2022, http://arxiv.org/abs/2209.11925. arXiv.