Department of Creative Informatics
Graduate School of Information Science and Technology
THE UNIVERSITY OF TOKYO

Master's Thesis

Cooperative Localization for Connected Autonomous Vehicles Utilizing LiDAR-Equipped Infrastructure

自動運転のための LiDAR 搭載の路側機を用いた
協調型自己位置推定

Yuze Jiang
姜 宇澤

Supervisor: Professor Hiroshi Esaki

January 2024

Abstract

Advancements in LiDAR technology have led to more cost-effective production while simultaneously improving precision and resolution. As a result, LiDAR has become integral to vehicle localization, achieving centimeter-level accuracy through techniques like Normal Distributions Transform (NDT) and other advanced 3D registration algorithms. Nonetheless, these approaches are reliant on high-definition 3D point cloud maps, the creation of which involves significant expenditure. When such maps are unavailable or lack sufficient features for 3D registration algorithms, localization accuracy diminishes, posing a risk to road safety. To address this, we proposed to use LiDAR-equipped roadside unit and Vehicle-to-Infrastructure (V2I) communication to accurately estimate the connected autonomous vehicle's position and help the vehicle when its self-localization is not accurate enough. Our simulation results indicate that this method outperforms traditional NDT scan matching-based approaches in terms of localization accuracy.

概要

世界中にて自動運転の研究開発や社会実装が進められている．その中で，自動運転における自己位置推定において LiDAR 技術の役割は重要である．また，LiDAR 技術の進歩は，精度と解像度の向上と同時に，より費用対効果の高い生産につながっている．その結果，LiDAR は車両の自己位置推定に不可欠なものとなり，NDT スキャンマッチングやその他の高度な 3D 位置推定アルゴリズムなどの技術により，センチメートルレベルの精度を達成することができている．しかしながら，これらのアプローチは高精細な 3D 点群マップに依存しており，その作成には多額の費用がかかるという問題がある．このような地図が利用できなかったり，3D 位置推定アルゴリズムをするための周囲に十分な特徴点がなかったりした場合，自己位置推定の精度が低下してしまい，自動運転が正しく動作しなく交通の安全性の問題を引き起こす可能性がある．この問題に対処するため，LiDAR を搭載した路側機と V2I（Vehicle-to-Infrastructure）通信を利用し，協調型自動運転の位置を正確に推定し，自己位置推定が十分に正確でない場合に車両を支援する手法を提案する．シミュレーションの結果，提案手法は従来の NDT スキャンマッチングベースよりも自己位置推定の精度の点で上回ることが示された．

Contents

Contents v

List of Figures vii

List of Tables ix

List of Acronyms ix

Chapter 1 Introduction 1
1.1 Background . . . 1
1.2 Proposal and Contributions . . . 5
1.3 Organization . . . 5

Chapter 2 Related Work 6
2.1 LiDAR-Based Vehicle Self-Localization Techniques . . . 6
2.2 Vehicle Detection and Tracking via Roadside Infrastructure . . . 8
2.3 Cooperative Localization Methods . . . 10

Chapter 3 Analysis and Improvement of Roadside LiDAR Perception 14
3.1 Understanding the Accuracy of Roadside LiDAR Perception via the Heat Map . . . 14
3.2 Pilot Experiment Results . . . 17

Chapter 4 Proposed V2I-Based Cooperative Localization System 22
4.1 Prior Information on Vehicle Size via V2I Communication . . . 23
4.2 Real-Time Information on Vehicle Yaw Angle via V2I Communication 25
4.3 Real-Time Information on Vehicle Reference Position via V2I Communication . . . 25

Chapter 5 Simulation Experiment of Roadside LiDAR-Based Localization 26
5.1 Implementation of Roadside Position Estimation Module . . . 26
5.2 Experiment Setup . . . 27
5.3 Evaluation and Analysis . . . 29

Chapter 6 Conclusion 38
6.1 Summary . . . 38
6.2 Future Work . . . 38

Publications 40

References 41

List of Figures

1.1 6 Levels of Autonomous Driving by SAE [5]. 2
1.2 A Simplified Workflow of a Typical Autonomous Vehicle 2
1.3 An Example of How C-ITS Helps to Eliminate Blind Spots [7]. 3
1.4 Sensor Setup of a Typical Autonomous Vehicle [11] 4

2.1 HD Point Cloud Map for Nishishinjuku District, Tokyo, Japan 6
2.2 3-Dimensional Registration Algorithm [21]. 8
2.3 Background Filtering Result from A Roadside LiDAR. 9
2.4 The DBSCAN (Density-Based Spatial Clustering of Applications with Noise) Algorithm [29]. 10
2.5 A Multilateration Surveillance System Used for Detecting the Position of an Airplane. 11
2.6 The Structure of the Multi-Sensor Multi-Vehicle (MSMV) Cooperative Localization System [44]. 11
2.7 Proposed Radio Frequency-Based V2I Cooperative Localization Scheme by [54]. 12

3.1 Pilot Experiment Flow Chart 15
3.2 The Setup of the Pilot Experiment to Generate Heat Map 16
3.3 Error of the Center Point in Distance (m) 18
3.4 Number of Points Perceived by the Roadside LiDAR 18
3.5 Error of Yaw Angle (°) 19
3.6 A Case of Rear-View Mirror Causing Increased Yaw Angle Error. 20
3.7 Error of the Bounding Box Size (m^2) 20

4.1 Semantic Diagram of the Proposed System 23
4.2 Error of the Center Point in Distance (m) 24
4.3 A Case of Partial Vehicle Perception and Size Correction Algorithm In-Action 24

5.1 The Data Flow of the Simulation Experiment 29
5.2 Existing Study on Evaluation of Localization Error in the Nishishinkjuku Map [67]. 30
5.3 Configuration of the Scenario 1. 30
5.4 Localization Error Comparison in Scenario 1 31
5.5 Localization Error for VLP-16 with Heuristic Angle Search 32
5.6 Box Plot of MAE for Different Localization Methods in Scenario 1 . . . 32
5.7 Configuration of the Scenario 2. 33
5.8 Localization Error Comparison for Scenario 2 33
5.9 Localization Error Comparison for Scenario 2 for Ten Trials 34

5.10 The MAE of NDT-Based Localization and the Proposed Method for Scenario 2. 34
5.11 Configuration of the Tunnel Scenario. 35
5.12 Localization Error Comparison for Scenario 3 35
5.13 Time Consumption Breakdown for VLS-128 36

List of Tables

3.1 Specification of Hardware and Software 15
3.2 Pilot Experiment Parameters 17

5.1 Subscription List of `listener` 26
5.2 Publication List of `listener` 27
5.3 Simulation Experiment Parameters and Specifications 28
5.4 Technical Specifications of LiDAR Models 28
5.5 The MAE of NDT-Based Localization and the Proposed Method with Two Different LiDAR Models for Scenario 1. 31

List of Acronyms

ADAS Advanced Driver-Assistance System

ABS Anti-lock Braking Systems

AI Artificial Intelligence

AD Autonomous Driving

AV Autonomous Vehicle

CAM Cooperative Awareness Message

CAV Connected Autonomous Vehicles

C-ITS Cooperative Intelligent Transport Systems

CPM Collective Perception Messages

CPS Collective Perception Service

EKF Extended Kalman Filter

DBSCAN Density-Based Spatial Clustering of Applications with Noise

DSRC Dedicated Short-Range Communications

HD High Definition

ETSI European Telecommunications Standard Institute

GNSS Global Navigation Satellite System

GPS Global Positioning System

NDT Normal Distributions Transform

ICP Iterative Closest Point

LiDAR Light Detection and Ranging

OBU On-Board Unit

RSU Road Side Units

ITS Intelligent Transportation Systems

ITS-S ITS Station

SAE Society of Automotive Engineers

V2I Vehicle-to-Infrastructure

V2V Vehicle-to-Vehicle

V2X Vehicle-to-Everything

C-V2X Cellular-V2X

P2P Peer-to-Peer

NLOS Non-Line-of-Sight

FoV Field of View

MAE Mean Absolute Error

Chapter 1

Introduction

1.1 Background

Vehicles have changed the lives of human beings by hugely extending the living range in the past 100 years [1]. Driving has quickly become the most used transportation method on this planet. Researchers, architects, engineers, and designers have put tremendous effort into improving the functionalities and safety of human-driven vehicles [2]. Safety features such as seat belts, airbags, and anti-lock braking systems (ABS) became standard decades ago. Sophisticated systems like advanced driver-assistance systems (ADAS) also prevail in modern vehicles. It is calculated that most of the newly manufactured vehicles are equipped with ADAS [3]. On the other hand, autonomous driving (AD) has been rapidly transcending over the past decade due to the advancement of artificial intelligence (AI) technologies and sensors. With an outlook for the future connected autonomous vehicle (CAV) deployment, we can achieve more safety with less congestion on the road. For example, the Korean Ministry of Land, Infrastructure and Transport predicted that travel speed would increase 30% in the downtown city and incidents will decrease by 46% in the future [4], which is because CAVs can acquire information from onboard sensors and exchange information with other intelligent transport system stations (ITS-S). Therefore, a CAV can access more information than a human-driven vehicle. The problems and challenges have significantly changed from human-driven vehicles to autonomous-driving vehicles.

1.1.1 Autonomous Driving and Cooperative Intelligent Transport Systems

As Figure 1.1 shows, the Society of Automotive Engineers (SAE) has defined six levels of autonomous driving. These levels describe how much a vehicle can take over driving tasks and the level of human intervention required. Each level represents a significant step in technological advancement, with higher levels reducing the need for human input and control. As we are approaching higher level AD, it has become clear that standalone autonomous driving has limitations we cannot control – occlusions and blind spots. AVs rely on the sensors from their onboard unit (OBU) to sense the local environment and detect obstacles. The placement of sensors from OBU has to be, to some extent, confined within the vehicle. Thus, we cannot always get all the knowledge of the environment from these sensors. A collaboration of intelligent vehicles and infrastructure is needed to push the limit of autonomous driving further.

Cooperative Intelligent Transport Systems (C-ITS) represent a cutting-edge approach in modern transportation, focusing on the collaborative aspect of vehicular communication and data exchange. Participants of C-ITS are supposed to collaborate with other ITS-S, including connected vehicles, infrastructure, and pedestrians, via wireless com-

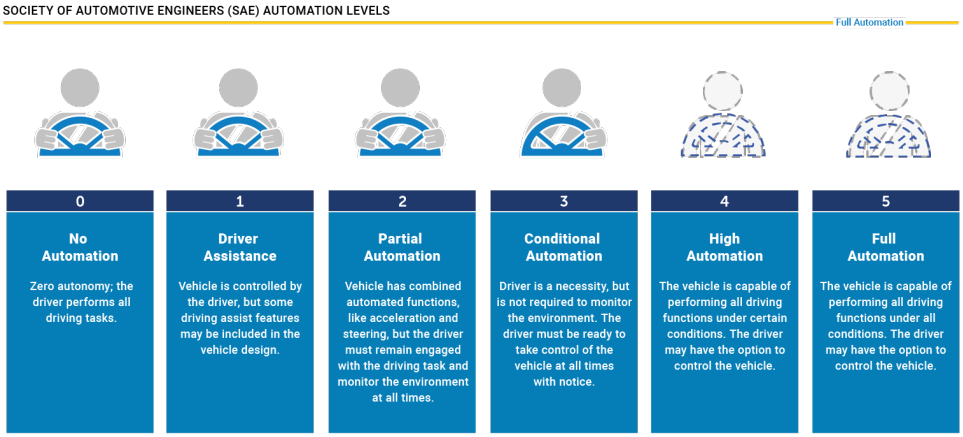


Fig. 1.1: 6 Levels of Autonomous Driving by SAE [5].

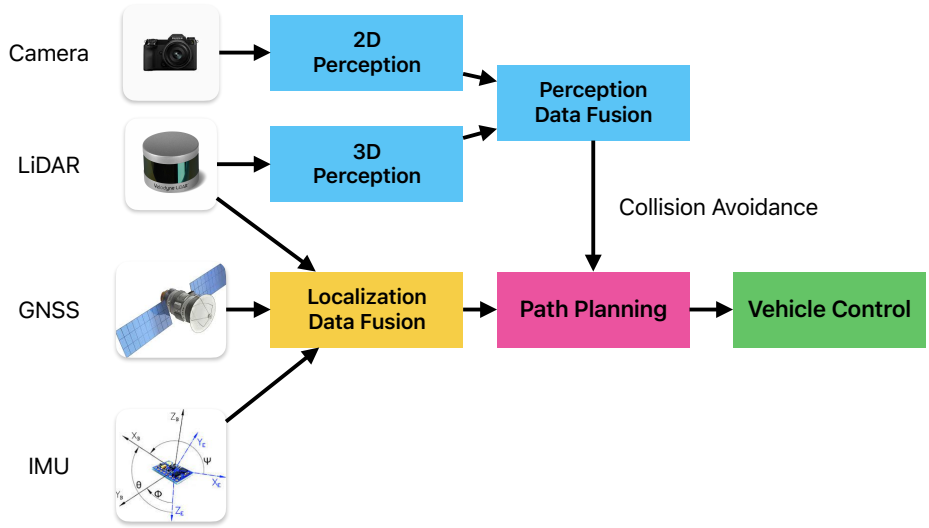


Fig. 1.2: A Simplified Workflow of a Typical Autonomous Vehicle

munication. By sharing information, C-ITS can assist in proactive traffic management, accident prevention, and optimized route planning. This collaborative approach is pivotal in advancing autonomous driving technologies and plays a significant role in shaping the future of smart cities [6]. With the help of C-ITS, we essentially fuse the sensor from multiple ITS-S, as Figure 1.3 shows an example of how such data fusion can eliminate blind spots.

Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communications form the cornerstone of C-ITS. These components are essential in enabling AVs to interact with each other and the surrounding infrastructure. In this setup, AVs are typically outfitted with OBUs. These OBUs are sophisticated devices capable of both transmitting and receiving traffic messages. The V2V communication process involves exchanging information between OBUs installed in different connected vehicles. This exchange is essential for coordinating movements and ensuring safety among vehicles on the road. On the other

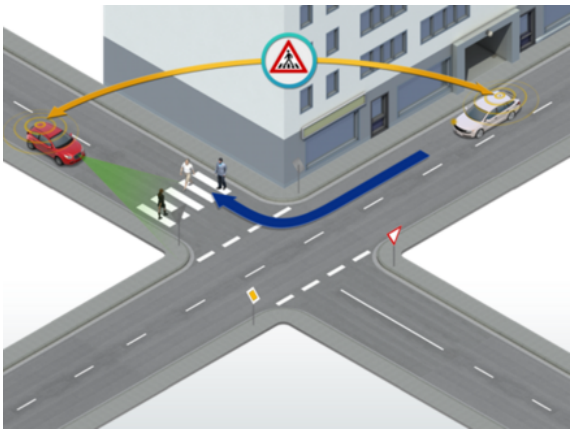


Fig. 1.3: An Example of How C-ITS Helps to Eliminate Blind Spots [7].
This is an example of how cooperative perception can help eliminate blind spots. In the figure, the white vehicle cannot observe the existence of the pedestrians on the zebra crossing via its on-board sensors, thus it cannot make decisions when it wants to turn right. However, the red vehicle observes the pedestrians, then it send this information to the white vehicle via V2V communication to help it to make better autonomous driving decision.

hand, V2I communication occurs between an OBU in a vehicle and a base station that's part of the connected roadside infrastructure. This interaction lets vehicles receive critical information about road conditions, traffic signals, and other infrastructure elements, facilitating smoother and safer travel. These technologies contribute to the formation of Vehicular Ad-hoc Networks (VANETs). This network is characterized by its decentralized nature, where each node can operate as both a receiver and a transmitter of information. This setup enhances the overall efficiency and safety of the transportation system by enabling real-time data sharing and collaborative decision-making among vehicles and infrastructure components.

1.1.2 Localization in Autonomous Driving

A typical autonomous driving workflow contains four important stages: localization, perception, path planning, and control, as depicted in the Figure 1.2. In robotics, localization is a process to determine the relative position of a mobile robot inside its environment, which is often an indoor situation [8]. In the context of autonomous driving, localization is finding where a vehicle is in a map. Localization is the foundation cornerstone of a autonomous driving system, as all the AD modules requires the precise location of the AV in real time. Vehicle localization can be divided into two categories: global navigation satellite system (GNSS)-based localization and vision-based localization. Global positioning system (GPS) is a popular GNSS solution, it provides up to meter-level positioning accuracy when using GPS standalone under best situation [9]. GPS has been widely used in the present vehicles for global path planning [10], in which vehicles only need to differentiate which road they are on, so that the required localization accuracy is relatively low. On the other hand, autonomous vehicles (AV) need to identify the current lane on a road as the bare minimum, which requires at least around 30 centimeters accuracy in localization [9], GNSS-based localization often falls short for this task.

Vision-based localization systems hugely differ based on the sensors they use. The RGB camera is the most essential 2D image sensor, and it has an increasingly powerful ability in object detection tasks with the help of neural networks [12]. However, the usage of

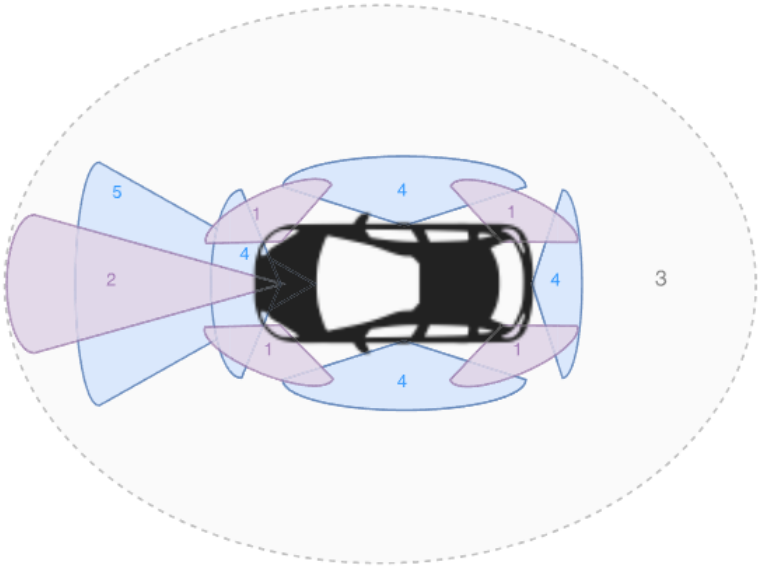


Fig. 1.4: Sensor Setup of a Typical Autonomous Vehicle [11]
The purple areas are covered by radars, cameras cover the blue areas, and the gray area is covered by LiDAR. While radars and cameras are usually used in near-object detection and cruise control, respectively, LiDARs are used to sense the environment in a broader range.

cameras in localization is limited due to the fact that they lose 3D depth information from the environment. Even though researchers have developed localization methods that use 2D cameras [13], such methods also need a prior 3D high-definition (HD) map in order to perform localization. LiDAR, one of the widely-used 3D image sensors, refers to Light Detection and Ranging. It uses light travel time to measure the distance between the sensor and the non-transparent obstacles.

$$l = \frac{1}{2} c\delta$$

Here, l is the distance we want to measure, c is the time of light, and δ is the round-trip delay from when the laser is emitted until the LiDAR receives the reflected laser. Rotor-based mechanical LiDARs, which contain multiple laser beams that are able to measure the distance, are the most used LiDAR type in the automotive industry [14]. The set of laser beams rotates inside the LiDAR sensor at a fixed frequency, and once a 360° rotation is finished, a 3D scan of the environment is generated, called point cloud data. A 3D point cloud registration algorithm is performed to match the current scan to the prior HD map to get the sensor's location. Thanks to the incomparable precision of the LiDAR sensor, the LiDAR-based localization can achieve centimeter-level accuracy under favorable situations [15]. As depicted in Figure 1.4, a typical AV combines cameras and LiDARs to make use of the strength of each type of sensor.

Even though we can achieve highly accurate localization results from vehicle LiDAR, we cannot only depend on it. Firstly, we need a prior HD map in order to perform map-based localization. The HD map costs a lot to produce, calibrate, and maintain. Such limitations make it not profitable to make and maintain HD maps at all locations [16]. Secondly, even if we have an HD map, the localization accuracy is also affected by the features in the local map [17]. For instance, when the vehicle is on the highway without buildings and other structures besides the road, registration algorithms may not work well, resulting in a degraded localization performance.

1.2 Proposal and Contributions

Localization is a crucial part of AD. However, its accuracy is always affected by environmental factors such as deficiency in features or multi-path effects in wireless communication. We want to tackle this problem by introducing stationary roadside infrastructure into the localization process to stabilize the localization performance at places where localization degradation is prone to happen. We thus proposed a roadside LiDAR-based, V2I communication-enabled cooperative localization system for CAVs. The infrastructure will acquire real-time information about CAV by facilitating standard messages in the VANET communication. After sensing the CAV by the roadside LiDAR, the infrastructure will use the real-time information to refine and calculate an accurate localization result for the CAV and send it to the vehicle. The vehicle then fuses the roadside localization result with the other onboard sensors to improve its localization accuracy in the area covered by supported infrastructure.

This thesis presents several contributions to the field of autonomous vehicles. Firstly, it delves into the novel use of roadside LiDAR systems for object detection, aiming to improve the localization capabilities of connected autonomous vehicles. Secondly, the research introduces an innovative method designed to increase the accuracy of determining a CAV's central point. This method leverages V2I communications, emphasizing the synergy between vehicles and roadside infrastructure to refine localization processes. Last but not least, the effectiveness of the proposed approach is validated through simulation experiments. These experiments reveal that our approach notably outperforms traditional Normal Distributions Transform (NDT)-based self-localization methods in specific scenarios. This finding underscores the potential of the proposed method in advancing the precision and reliability of autonomous vehicle localization.

1.3 Organization

The remainder of this thesis is organized as follows: Chapter 2 reviews existing literature on vehicle self-localization, cooperative localization strategies, and other techniques that incorporate roadside units and LiDAR, like roadside vehicle detection and tracking. Chapter 3 introduced the pilot experiment design and the discussion of the pilot experiment results. Chapter 4 detailed the design of the proposed cooperative localization system with V2I communication. In Chapter 5, we first discussed some details of the implementation of the roadside localization module, and then we detailed our simulation-based experiment and analyzed its result. Concluding remarks and future research directions are presented in Chapter 6.

Chapter 2

Related Work

In this chapter, we will introduce some of the state-of-the-art LiDAR-based self-localization and cooperative localization techniques and discuss their strengths and weaknesses.

LiDAR-based localization methods can be divided into two main categories: Simultaneous Localization And Mapping (SLAM) [18] and map-based localization methods. The main difference between them is that SLAM does need prior knowledge about the environment. In contrast, map-based methods require a static 3D map of the environment, called High Definition Maps (HD Maps). Figure 2.1 shows such an HD map for the Nishishinjuku district, Tokyo, Japan. Although SLAM can handle dynamic environments better than map-based methods, it suffers from cumulative error. Thus, in general, map-based localization methods have better accuracy than SLAM.

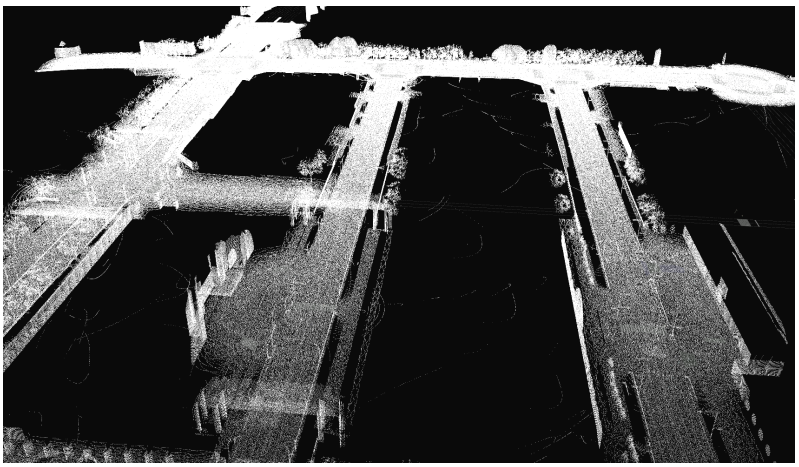


Fig. 2.1: HD Point Cloud Map for Nishishinjuku District, Tokyo, Japan
Credit: Tier VI

2.1 LiDAR-Based Vehicle Self-Localization Techniques

For map-based localization, the AV first acquires the HD map for the vicinity. Then, the AV uses the onboard LiDAR sensor to sense the surrounding environment. Finally, the AV localizes itself by transforming the perceived point cloud data to match the prior map using 3D registration algorithms. Figure 2.2 is an example of how two point clouds merged together via a 3D registration algorithm. Two popular registration algorithms are often used for vehicle localization tasks: Iterative Closest Point (ICP) [19] and Normal Distribution Transform (NDT) [20].

Iterative Closest Point (ICP)

ICP tries to minimize the summed distance between points from the HD map and the perceived point cloud from vehicle LiDAR. Assume the point set of the HD map is M, and the perceived point cloud is L. In each iteration, the algorithm finds the best rotation $\mathbf{r}$ and translation $\mathbf{t}$ to L to minimize the sum of the squared distance:

$$\underset{\mathbf{r},\mathbf{t}}{\operatorname{argmin}} \sum_{i=1}^{|M|} \sum_{j=1}^{|L|} w_{i,j} ||\mathbf{m}_i - (\mathbf{r} \cdot \mathbf{n}_j + \mathbf{t})||^2$$

Here $w_{i,j}$ is 1 if M_i is the nearest point to L_i; 0 otherwise.

Unlike the ICP algorithm, which works directly with point clouds, NDT represents the environment using a set of overlapping normal distributions.

Normal Distribution Transform (NDT)

First, the 3D environment is divided into voxels. Within each voxel, a mean $\mathbf{c}$ and a covariance matrix Σ are calculated based on the points inside the voxel.

$$\mathbf{c} = \frac{1}{|P|} \sum_{i=1}^{|P|} P_i$$

$$\Sigma = \frac{1}{|P|} \sum_{i=1}^{|P|} (P_i - \mathbf{c})(P_i - \mathbf{c})^T$$

In the equation above, P is the set of points in a voxel. After down sampling the input scan by filtering it into F, we find the optimal transformation by minimizing the following cost function:

$$\underset{\mathbf{r},\mathbf{t}}{\operatorname{argmin}} \sum_{i=1}^{|F|} \exp\left(-1/2((\mathbf{r}F_i + \mathbf{t}) - \mathbf{c}_i)^T \Sigma_i^{-1} ((\mathbf{r}F_i + \mathbf{t}) - \mathbf{c}_i)\right)$$

The optimization algorithm is usually done by utilizing Newton's method.

Though NDT and ICP are both widely used algorithms for point cloud registration and vehicle localization, NDT is generally superior to ICP in terms of accuracy and robustness [22]. A manual initial guess is needed for the algorithm as a bootstrap when using NDT to perform vehicle localization at the starting phase. Subsequent NDT scan matching's initial guess will be given by the vehicle motion model [23].

In [24], the researchers performed a real-world experiment showcasing the performance of NDT matching-based localization. The vehicle's onboard sensors contain GNSS, IMU, and LiDAR. The results show that in a GNSS-challenging environment, NDT scan matching can improve the 2D localization accuracy by up to 70%. The researchers also noted that the initial guess is pivotal for the NDT scan matching-based localization. In a study by Kan *et al.* [25], the efficacy of map-based NDT scan matching was assessed under conditions where LiDAR scans were partially occluded. The results revealed that localization remained stable until the occlusion rate exceeded 25% and only faltered when the rate surpassed 80%, underscoring the resilience of NDT scan matching.

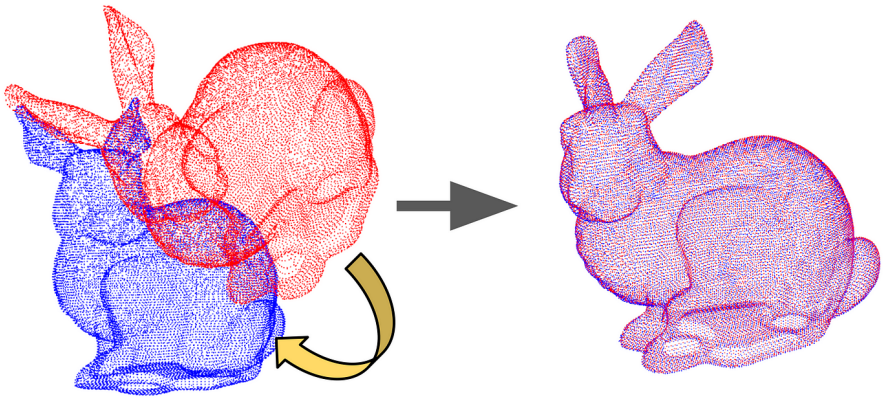


Fig. 2.2: 3-Dimensional Registration Algorithm [21].

Despite its strengths, the success of NDT-based localization is contingent upon the map's granularity [22] and the availability of distinctive features [17]. Consequently, the effectiveness of NDT-based localization varies based on the map's detailedness or the environment's complexity. For example, in a highway or a rural environment, the NDT scan matching-based localization may perform poorly because the features in the map are repeated or not prominent enough [26].

2.2 Vehicle Detection and Tracking via Roadside Infrastructure

Infrastructure plays a vital role in the VANET. As stipulated by the ESTI (European Telecommunications Standards Institute) standard for Collective Perception Service (CPS) [27], infrastructure has the role of obtaining information from mounted stationary sensors and broadcasting the information via Collective Perception Message (CPM) to other connected agents nearby. Therefore, the infrastructure is responsible for tracking and monitoring the objects (including vehicles and pedestrians) on the road. Many studies focus on vehicle tracking from the roadside infrastructure, some of which use roadside LiDAR.

Vehicle tracking via roadside LiDAR usually goes through the following stages: background filtering, clustering, classification, linking, and tracking objects in multiple frames. Compared to vehicle LiDAR, roadside LiDAR can perform background filtering with relatively high precision with low recall due to the stationary nature of roadside LiDAR. Figure 2.3 depicts a result of background filtering for a LiDAR frame captured from a roadside LiDAR. A novel method, 3D-DSF [28], was proposed for roadside LiDAR background filtering tasks. This method uses the statistical history data of the roadside LiDAR to determine whether a point is a background point or not. After background filtering, vehicle and pedestrian points remain in the scan. The next step is to cluster the remaining points into objects. This step is usually done by DBSCAN [29], which is a density-based algorithm to identify cluster spatial points with the ability to mark outliers in a noisy data set. Figure 2.4 is a diagram showing how DBSCAN works. In the object classification process, each object is labeled with its object type (in this case, most of the objects are vehicles and pedestrians). Many methods were proposed for this task; they can be divided into machine learning (SVM)-based [30, 31, 32, 33], and deep learning-based methods [34, 35, 36]. Finally, in the tracking stage, the same object from different LiDAR scan frames is linked to track the object so that we can acquire a trajectory of the vehicle

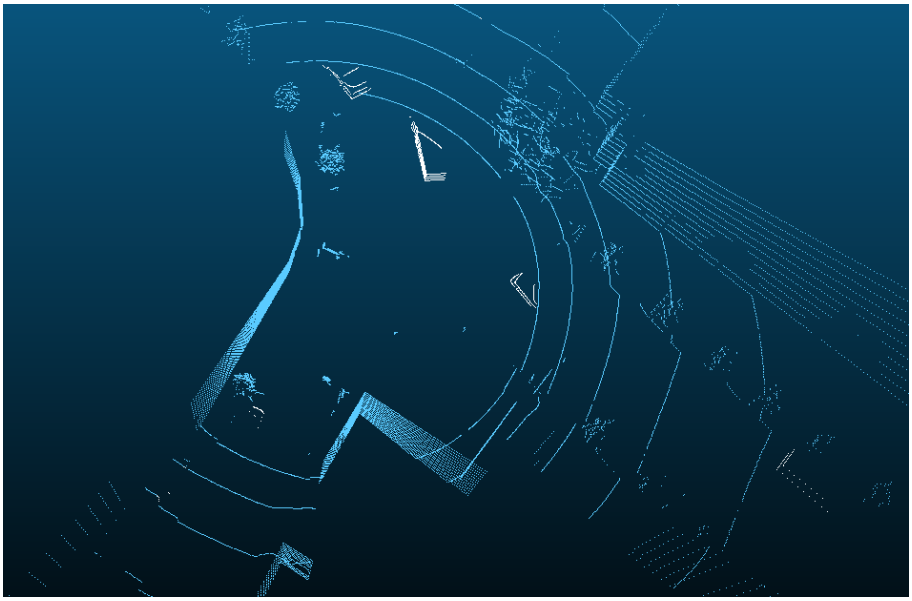


Fig. 2.3: Background Filtering Result from A Roadside LiDAR.
This LiDAR frame was recorded in AWSIM (a simulation software) using VLP-16. Blue points represent the background points, and the white points are the vehicle points. There are three vehicles in the frame, and these vehicles are shown in three white point clusters in this picture.

and pedestrians on the road based on the roadside LiDAR data. Further analysis could be done, for example, to estimate the speed of the vehicles or to detect misbehavior [37].

Zhao *et al.* [38] conducted a real-world experiment where roadside LiDAR is used to detect, track the trajectories, and estimate the speed of the vehicles and pedestrians at an intersection. In this study, the authors evaluated the accuracy of the estimated speed of the vehicles by comparing it with the speed measured from the onboard devices. The authors did not evaluate the accuracy of the tracked trajectories of the vehicles. Therefore, the trajectories cannot be used to enhance the connected vehicles' self-localization. Subsequently, Zhang *et al.* [39] improved the roadside LiDAR-based speed tracking performance by introducing centroid-based tracking. By using this method, the tracked trajectories become smoother. However, the position of the tracking point is relatively moving with respect to the vehicle's center point, rendering it unsuitable for localization purposes.

Many studies have focused on vehicle tracking and speed monitoring/validation from a roadside LiDAR. Despite their contributions, hardly any study intended to leverage the roadside LiDAR to enhance the connected vehicles' localization accuracy. One of the critical reasons for evaluating the localization accuracy of roadside LiDAR is that it is hard to acquire the ground truth of vehicle position in a real-world setting. Although one can use a GPS-based system to collect the ground truth, the accuracy and applicability are often limited due to the multipath effect and Non-Line-of-Sight (NLOS) problems [9]. Some efforts have been made, though, to mitigate this issue by evaluating cooperative localization accuracy. For instance, in a recent study [40], the authors used a set of 1/10 scale model autonomous vehicles to be able to get the advantage of using an Optitrack System [41] for ground truth collection. However, not only is the accuracy of this positioning system not evaluated in the paper, but the scaled model is also not accurate compared to an actual vehicle, rendering the results of this work unwarranted. To overcome these problems, we decided to use AWSIM as the simulator to evaluate our proposed method in this work. For one, AWSIM can provide the ground truth of the vehicle at a very high frequency;

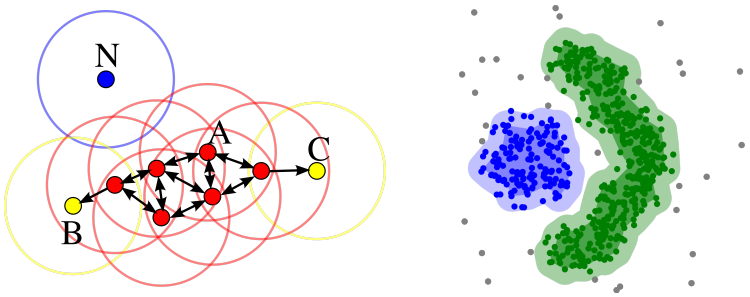


Fig. 2.4: The DBSCAN (Density-Based Spatial Clustering of Applications with Noise) Algorithm [29].
The algorithm works by identifying points in dense regions of a dataset as clusters, using two parameters: minimum number of points required to form a cluster and the maximum distance between two points for them to be considered part of the same cluster. It classifies points as core points if they have enough neighbors, border points near core points with fewer neighbors, or noise points isolated from dense areas.

the model of the testing vehicle is also identical to a real-world vehicle.

2.3 Cooperative Localization Methods

In the context of C-ITS, many tasks that were done standalone previously are now done collaboratively for better reliability and robustness owing to the development and advancement of the Internet of Things (IoT) and V2X communications. These technologies enable us to fuse the data from multiple sensors or even from different connected nodes in the VANET [42]. Localization, one of the most crucial autonomous driving tasks, is also being explored for cooperative methods. The key idea behind cooperative localization is to use sensors from other nodes in the network (other than the ego-vehicle) to help the ego-vehicle localize itself. Cooperative localization often involves multilateration. Multilateration is a technique that uses inter-vehicle distance data and individual vehicle self-localization outcomes to determine a subject vehicle's relative position, as shown in Figure 2.5. Depending on the type of helping nodes, cooperative localization can be generally divided into vehicle-to-vehicle-based methods and vehicle-to-infrastructure-based.

2.3.1 Vehicle-to-Vehicle-Based Cooperative Localization Methods

In [43], Fujii *et al.* proposed a multiple-vehicle system in which the vehicles measure and share the distances to other vehicles together with their current position. Two sensors are being used in this system: a GPS sensor for providing self-localization results as well as speed information and a ranging sensor to estimate inter-vehicle distances. The authors only specified a millimeter-wave radar sensor for the ranging sensor as an example. Upon receiving the shared information, a vehicle can improve its localization accuracy by over

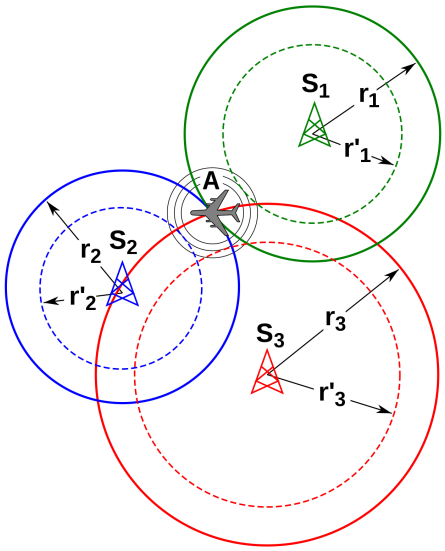


Fig. 2.5: A Multilateration Surveillance System Used for Detecting the Position of an Airplane.

50% compared to self-localization. Yang *et al.* developed a Multi-Sensor Multi-Vehicle framework [44] which enables data from advanced sensors, including LiDAR, to be fused via V2V communication, as depicted in Figure 2.6. The simulation-based results show that their method can improve the GPS-based localization accuracy, and the improvement of accuracy is proportional to the number of vehicles participating in this system. Only when the number of vehicles exceeds 60 can the mean squared error be reduced to 0.6 m. On the one hand, this number of connected vehicles sensing each other is unrealistic; conversely, it becomes indispensable to consider network heterogeneity and delay in such a crowded communication environment. Nevertheless, the efficiency of such V2V cooperative localization systems is fundamentally limited by each vehicle's self-localization accuracy. The effectiveness of the whole system deteriorates when all vehicles cannot perform self-localization accurately enough, for example, due to environmental factors.

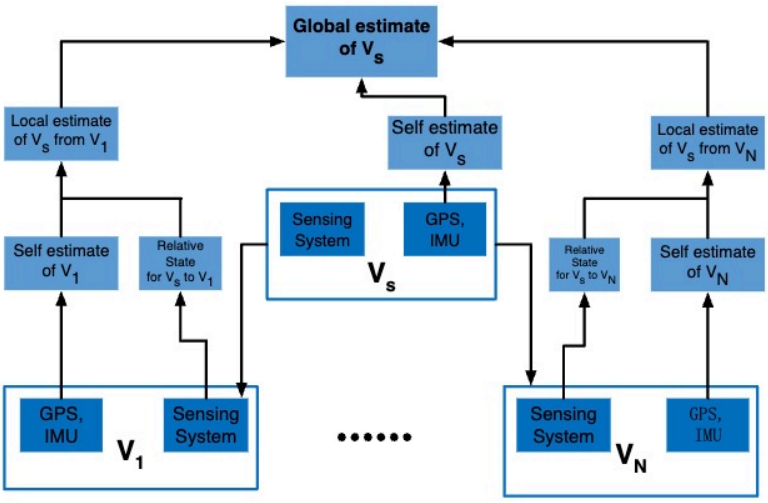


Fig. 2.6: The Structure of the Multi-Sensor Multi-Vehicle (MSMV) Cooperative Localization System [44].

Connected vehicles in the VANET essentially form a Peer-to-Peer (P2P) network. Cooperative perception is a great example showcasing the power of reducing blind spots of V2V communication under C-ITS. However, every node in a P2P network is both the sender and the receiver due to the lack of a centralized server. So, the cost of becoming a malicious node becomes lower than that in a traditional server-client structured network. Therefore, a P2P network needs to be attack-resistant. Many countermeasures were carried out against various types of attacks in P2P networks [45, 46, 47, 48]. Recently, some researchers started to focus on data verification for cooperative perception systems to enhance the authenticity of perception data shared among the V2V networks [49]. While V2V-based cooperative localization techniques advanced in recent years, the data authenticity issue remains vastly inextricable in this area.

2.3.2 Vehicle-to-Infrastructure-Based Cooperative Localization Methods

Stationary RSUs can be used as reference anchors in the V2I-based cooperative localization systems, significantly reducing the source of error. Another vital advantage of V2I systems is that we can control the placement of the RSUs at critical areas to improve localization accuracy as wished. In contrast, we cannot control the position of the vehicles. Existing literature mainly employs radio positioning techniques for V2I-based cooperative localization systems [50, 51], which not only demand precise time synchronization between the RSUs and the connected vehicles but are also susceptible to multipath effects [52]. Ou *et al.* proposed a system employing RSUs with directional antennas [53]. As depicted in Figure 2.7, once two or more RSUs detect a vehicle, the RSU measures the distance between it and the vehicle by using radio frequency-based distance measurement methods. Then, we draw two circles with the centers being the RSUs and the radius being the distance measured by the RSUs. The vehicle is on one of the intersections of the two circles. However, this approach struggles with complex maneuvers such as sharp turns and U-turns.

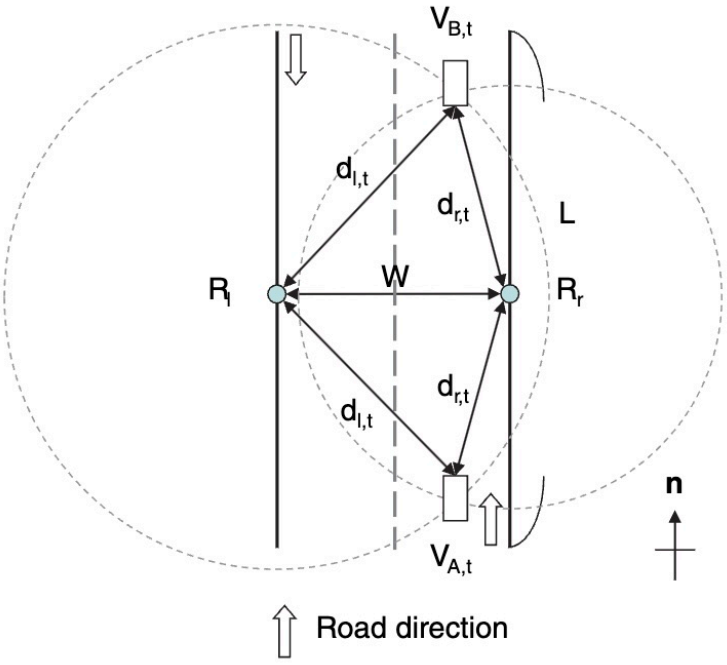


Fig. 2.7: Proposed Radio Frequency-Based V2I Cooperative Localization Scheme by [54].

The aforementioned radio frequency (RF)-based cooperative localization methods demand the deployment of dedicated hardware on the RSUs. Some research also explored the possibility of utilizing existing 4G/5G base stations for vehicle-to-infrastructure distance measurement to reduce the cost of the infrastructure. Liu *et al.* developed a collaborative positioning algorithm that merges received signal strength indication (RSSI) from base stations, GPS data, dead reckoning, and V2I communications [55]. While the algorithm outperforms GPS-only methods, its meter-level accuracy under realistic scenarios falls short for tasks like lane identification, rendering it inadequate for autonomous driving. Such RSSI-based methods inherently can only provide inadequate accuracy for cooperative localization [56]. To combine the strength of RF-based and 5G-based distance measurement methods, a novel method that records signal characteristics from multiple base stations into a database as fingerprints at different locations. Then, the received signal of a mobile client is collected and compared with recorded fingerprints to determine the location of the mobile client [57]. Fingerprinting techniques apply not only to signals from base stations but also to Wi-Fi Access Points [58]. Despite their contributions to reducing the expenditure of infrastructure deployment, their accuracy does not come close to the map-based vehicle self-localization.

Chapter 3

Analysis and Improvement of Roadside LiDAR Perception

Despite significant advancements in LiDAR technology and 3D registration algorithms enabling precise vehicle self-localization, its dependency on expensive 3D point cloud maps with identifiable features presents limitations in reliability under certain conditions. The challenge posed by maps with insufficient features is not easily resolvable. Meanwhile, the deployment of LiDAR-equipped roadside infrastructure emerges as a viable solution to these limitations. Our proposal, therefore, focuses on investigating the role of the infrastructure in enhancing localization accuracy. Part of this chapter has been published in a past conference paper [59].

3.1 Understanding the Accuracy of Roadside LiDAR Perception via the Heat Map

With a limited number of past studies that have exploited roadside LiDAR for vehicle localization enhancement, we first wanted to explore the roadside LiDAR's accuracy of vehicle perception and its ability for vehicle localization. From the point cloud data to the position of a vehicle, there is still one missing step – determining the representative point of the vehicle. As previously discussed in Section 2.2, although some methods of selecting the representative point for the vehicle point cloud were proposed, they are primarily for tracking and speed monitoring purposes. For example, the previous work used the center point of the point cloud or the nearest point to the LiDAR in the point cloud data as the tracking point. The motivation of such methods is for even and smooth tracking trajectories, which is both good for trajectory visualization and speed estimation. However, these methods failed to provide a stable reference point suitable for localization. Any point on the vehicle's silhouette will have a displacement of at least tens of centimeters, which is unacceptable for localization. In this study, we want to select the representative point for localization purposes. Therefore, we do not have to keep the trajectory smooth; instead, we want to find the point that is as close to the center of the vehicle as possible.

3.1.1 The L-Shape Fitting Algorithm

In this thesis, we used L-Shape Fitting Algorithm to estimate the bounding box based on the point cloud data of the vehicle, which was first proposed by Zhang *et al.* [60]. It is the state-of-the-art bounding box estimation algorithm for vehicle point clouds and is currently used as the default algorithm in Autoware [61]. Here, we assume that the 2D bounding box of every vehicle is always a rectangle. Given a yaw angle, we can always

draw a minimal rectangle that encloses all the vehicle points. The algorithm evaluates various placement angles and their closeness scores to find the best-fitting rectangle. The closeness score C is calculated by the following equation:

$$C = \sum_{i=1}^{n} \frac{1}{d_i}$$

Here, n is the total number of points, and d_i is the distance of the n-th point to the nearest edge of the estimated bounding box. To avoid the divide-by-zero problem, a threshold d_0 is set. Points with a distance lower than d_0 will directly contribute to the closeness score by a huge amount instead of calculating the reciprocal. After the closeness score for all yaw angles is calculated, the angle with the highest closeness score will be the yaw angle of the estimated bounding box.

3.1.2 Pilot Experiment Setup

Table 3.1: Specification of Hardware and Software

CPU	Ryzen 9 5950X
GPU	NVIDIA RTX 3090
RAM	128GB DDR4
Operating System	Ubuntu 20.04
ROS Version	ROS 2 Humble
AWSIM Version	v1.1.0

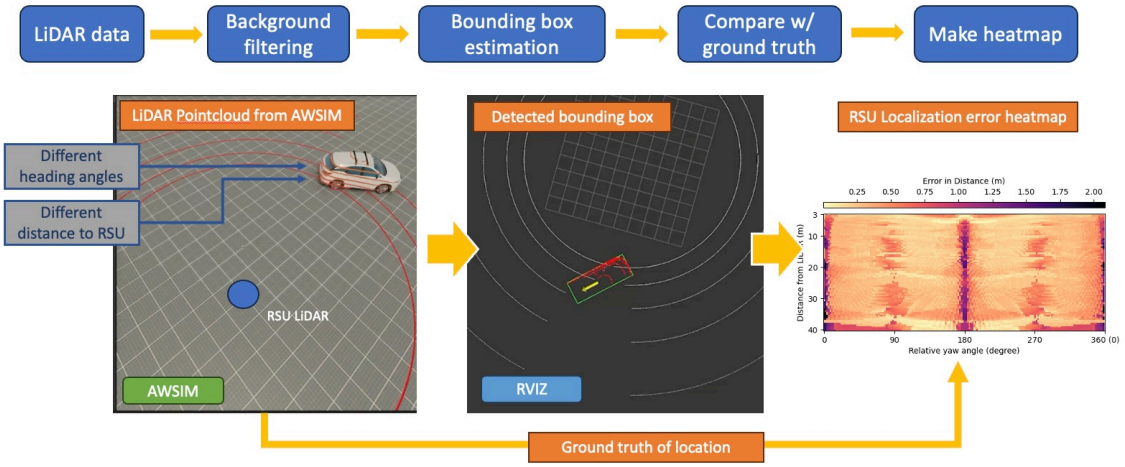


Fig. 3.1: Pilot Experiment Flow Chart

In the initial phase of our research, we carried out a preliminary study to investigate the capabilities of roadside LiDAR for vehicle localization under optimal conditions. We generated heat map visualizations based on the data we collected to show the accuracy of roadside LiDAR perception. As our primary simulation environment, we employed AWSIM [62], a Unity-based digital twin simulator tailored for autonomous driving research. AWSIM uses RobotecGPULidar (RGL) [63] as its LiDAR simulator, which is an efficient CUDA-powered LiDAR simulator that can simulate the LiDAR behavior quickly

and accurately. The host machine uses an AMD Ryzen 9 5950X CPU and NVIDIA RTX 3090 GPU. The detailed specification of the host machine can be found in Table 3.1

Flow chart 3.1 outlined the process of the experiment. We set up the experiment in an empty scene in the AWSIM Unity project, as illustrated in Figure 3.2, a LiDAR sensor was positioned at a static point in the air to function as a roadside LiDAR. We place the LiDAR at the origin point of the Unity world; therefore, no offset is needed to be applied when doing calculations. The LiDAR perceives the surrounding environment and publishes the perceived point cloud data to ROS [64] topics. Meanwhile, a test vehicle rotates around its center point at a constant rate. Upon completing a full 360° rotation, the vehicle moves 0.5 m away from the RSU. This set of movements allowed us to collect point clouds for various combinations of distances from the vehicle to roadside LiDAR and the vehicle's yaw angles.

After the vehicle is rotated and moved far away from the origin, the LiDAR starts to report its perception. Upon receiving the point cloud data from the LiDAR, we performed background filtering to acquire vehicle points. Since the experiment was done in an empty scene in AWSIM, the ground points are easy to identify. We filtered out zero-height points as background filtering from the collected point cloud data, leaving only vehicle points. Then, we use the L-shape fitting algorithm to estimate the position of the vehicle. Finally, we contrasted this with the ground truth to assess the localization error for each combination of distance and angle and used the data to make heat map visualizations. The detailed configuration is shown in Table 3.2. For more data collection and processing details, please refer to the implementation of the roadside positioning module detailed in Section 5.1.

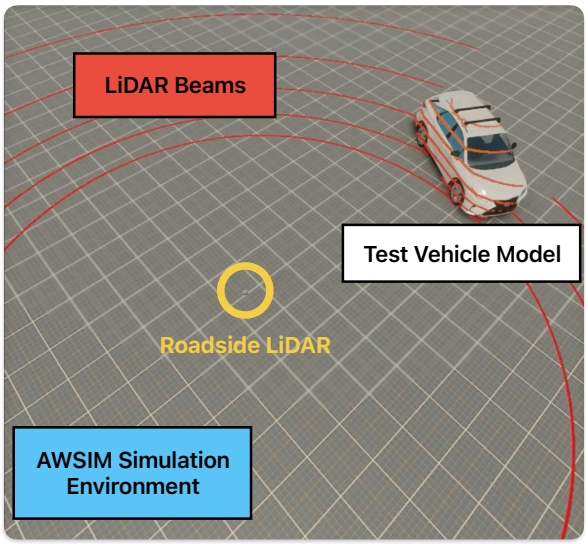


Fig. 3.2: The Setup of the Pilot Experiment to Generate Heat Map
The vehicle model rotates at a constant rate, during which the LiDAR reports the point cloud data it perceives at 10 Hz. We did background filtering to the acquired point cloud data and performed L-shape fitting to estimate the vehicle's position.

Table 3.2: Pilot Experiment Parameters

Test Vehicle Model	Lexus RX450h 2015
Vehicle Rotation Speed	20° per second
LiDAR Model	Velodyne VLP-16
LiDAR Height	2 m
LiDAR Reporting Frequency	10 Hz
Pitch and Roll Angle	0°

3.2 Pilot Experiment Results

After collecting the raw data, we used the center point of the bounding box as the vehicle's estimated location. We compared it with the ground truth to generate the error heat map for this pilot experiment. The yaw angle was estimated based on the orientation of the bounding box rectangle's longer edge. We only collected data for distances ranging from 3 m to 40 m between the vehicle and the roadside LiDAR since Starting the experiment from 0 m was meaningless due to the VLP-16 LiDAR's 30° vertical field of view (FoV) [11], which creates a detection gap directly underneath it. In practice, if the LiDAR is placed flat in the air, it cannot detect anything under it.

Post L-shape fitting, we assessed and visualized various errors and metrics: distance error for the bounding box center, bounding box size error, the number of points caused by the vehicle perceived by the roadside LiDAR, and yaw angle error at each given distance and angle. The following observations and deductions were made based on the Figures 3.3, 3.4, 3.7, and 3.5.

3.2.1 Accuracy Drop Due to Self Occlusion

We evaluate the localization accuracy as the distance between the center of the estimated bounding box and the ground truth of the center position of the vehicle. The error value is the absolute value of the error distance. We can easily observe from Figure 3.3 that the localization accuracy drops sharply when the vehicle is relatively parallel and perpendicular to the LiDAR. After inspecting some erroneous cases, we found that this is because the LiDAR beams cannot penetrate the vehicle. L-shape fitting works well when two edges of the vehicle are sensed. However, since the LiDAR beams cannot penetrate the vehicle, we can only get a partial point cloud of the vehicle at a time. When the vehicle's yaw angle is near 0° (or 360°), meaning the vehicle's front or back is almost directly pointing to the LiDAR, only half of the vehicle is sensed by the roadside LiDAR. Figure 4.3 shows an example of how self-occlusion can increase the error of the estimation position.

3.2.2 Accuracy versus Number of Perceived Points

For each frame that LiDAR captured, we recorded how many points were caused by the vehicle, then visualized it in Figure 3.4. The number of vehicle points perceived by a roadside LiDAR depends on the distance between the vehicle and the LiDAR. Despite VLP-16, the LiDAR model we used in this experiment, captures about 30,000 points per frame, the maximum number of vehicle points perceived by the roadside LiDAR in a frame is only around 1,200, and the number drops extremely fast when the vehicle is getting far away from the LiDAR. Most of the time, the LiDAR can only perceive the

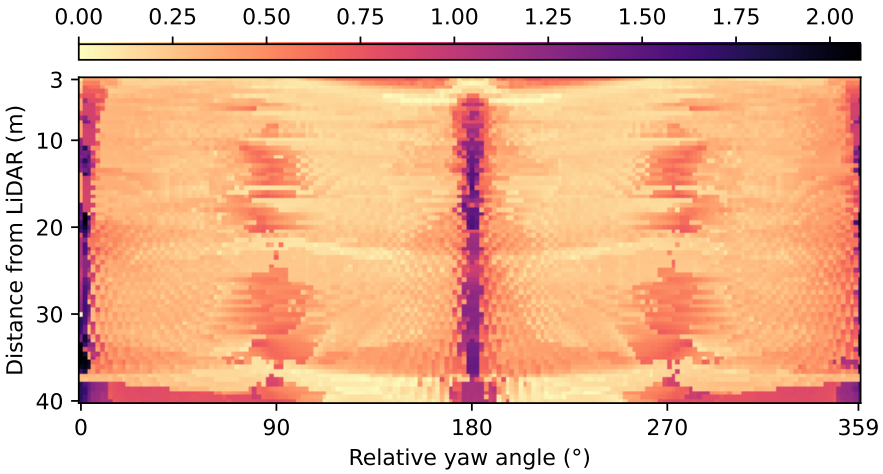


Fig. 3.3: Error of the Center Point in Distance (m)
When the vehicle is parallel and perpendicular to the roadside LiDAR, we can observe large errors. Besides, the typical localization error from 3 to 40 meters ranges from 15 cm to 50 cm. After 37 m, the accuracy starts to drop.

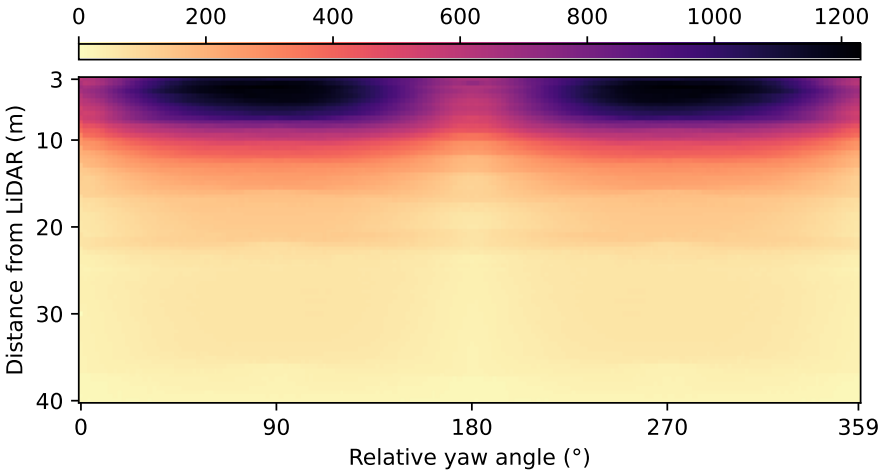


Fig. 3.4: Number of Points Perceived by the Roadside LiDAR
The number of perceived points by the roadside LiDAR decreases quickly as the distance grows due to a single vehicle not covering enough FoV of the LiDAR. Compared with Figure 3.3, we did not observe a direct relationship between the number of perceived points and the localization accuracy in terms of position error.

vehicle by a mere tens of points. The number of points perceived by the roadside LiDAR can be reverent and important in some tasks like object classification [38]. However, for localization, accuracy is not proportional to the number of points, as one might naively assume. This means that to acquire accurate localization results, the vehicle does not have to be very close to the roadside LiDAR to increase the number of perceived points, thus extending the effective range of the LiDAR.

3.2.3 Yaw Angle Estimation and Error Analysis

For vehicle localization, not only the accurate position is required, but the vehicle's yaw angle is also very important for the vehicle to determine its current state. In Figure 3.5, there are some black areas, which means the angle estimation is exactly off by 90° at the distance and yaw angle represented by the point in the figure. Since we solely estimate the yaw angle by the length of the edges of the bounding box, if the shorter edge in the estimated bounding box is the longer edge of the vehicle, we can observe the off-by-90° error.

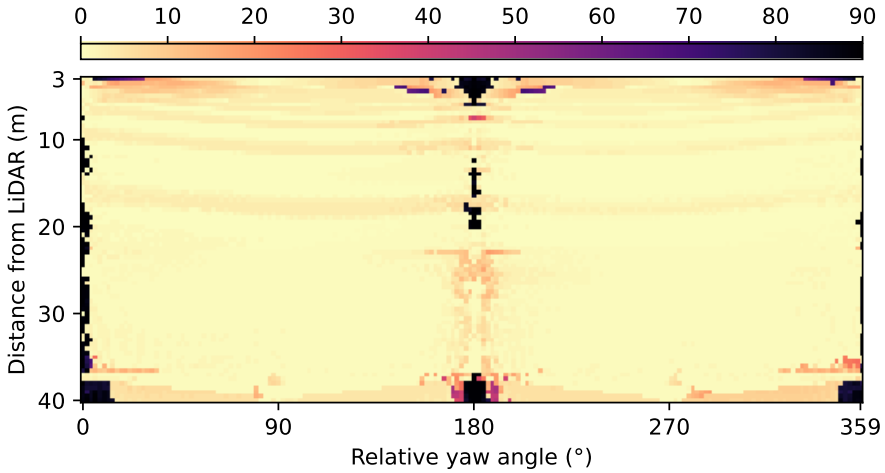


Fig. 3.5: Error of Yaw Angle (°)
Black areas indicate where off-by-90° issue happen. This issue will not only estimate a hugely incorrect yaw angle of the vehicle but also hurt the localization accuracy badly. Apart from that, the angle estimation is quite accurate, ranging from 0° to 5° for most of the situations.

Besides the black areas, the yaw angle error in other places is not very prominent, except we can observe some red ribbons in the figure. As suggested by such a structured inaccuracy pattern, we infer that there must be a systematical factor to cause this error. We therefore inspected an instance of LiDAR scan where this error happens, as shown in Figure 3.6. We can clearly observe that because one of the rear-view mirrors of the vehicle is sensed by the LiDAR, the L-Shape Fitting Algorithm has to enclose the mirror points into the bounding box. Hence, the bounding box's orientation deviates from the vehicle's actual yaw angle. Due to the layered nature of the LiDAR sensor, each layer of the LiDAR scan can only sense a specific height for a certain position. Therefore, the rear-view mirrors are not always visible by the LiDAR sensor. This is the reason why a ribbon-like inaccuracy pattern appears in the visualization.

3.2.4 Correlation with the Error in Bounding Box Size

We can observe that the pattern of localization inaccuracy closely aligns with the error pattern for bounding box size when we compare Figure 3.7 with Figure 3.3. This suggests that errors in bounding box dimensions correlate with errors in the estimated position of the vehicle's center. Because we only deal with 2D bounding boxes, the LiDAR sensing model can be assumed to be 2D. Thus, when a LiDAR senses a rectangle, only two edges can be sensed due to self-occlusion, and these two edges form the estimated bounding

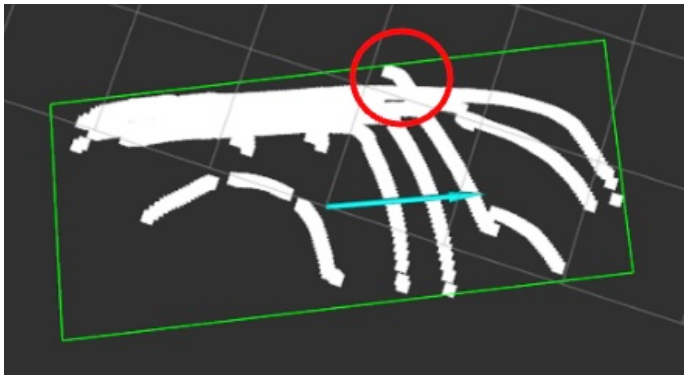


Fig. 3.6: A Case of Rear-View Mirror Causing Increased Yaw Angle Error.
The white points are perceived points by the roadside LiDAR. The blue arrow is the ground truth of the vehicle pose. The green rectangle is the estimated bounding box given by the L-Shape Fitting Algorithm. The red circle indicates the rear-view mirror of the vehicle, which causes the error of yaw angle to increase in a ribbon-like pattern in Figure 3.5.

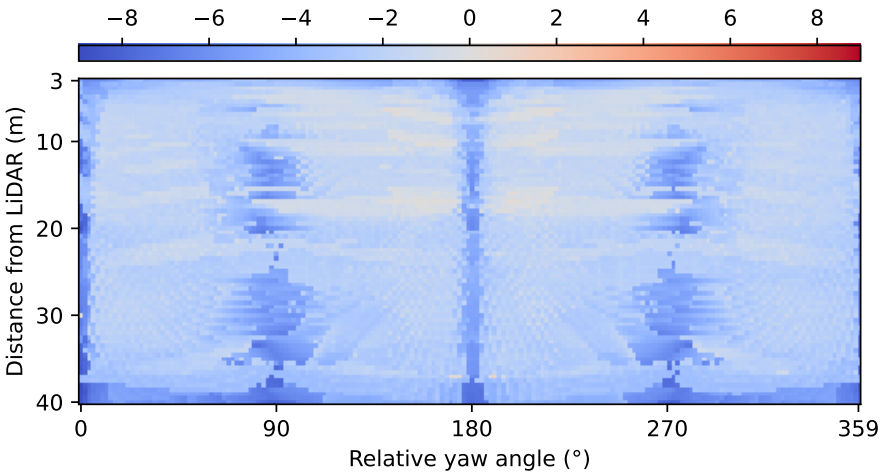


Fig. 3.7: Error of the Bounding Box Size (m^2)
When two edges are not entirely sensed by the roadside LiDAR due to self-occlusion, the bounding box size will shrink. The region near (170°, 18 m) appears reddish, meaning the estimated bounding box is larger than the actual size of the vehicle. This is due to the rear-view mirrors, as depicted in Figure 3.6.

box. The estimated bounding box size is the product of the two sensed edges. Therefore, to some degree, it can reflect the positioning error.

Some areas in Figure 3.7 appear reddish, which means the estimated bounding box size is slightly larger than the actual 2D size of the vehicle. This is also caused by the rear-view mirror. The size of the vehicle does not include the mirrors. However, in some rear cases, when the rear-view mirror is in the LiDAR scan, the estimated bounding box will be extended to include the mirror points, making the estimated bounding box larger than the actual size. It is important to note that this occurrence does not reflect a flaw within the L-Shape Fitting Algorithm itself, but rather it underscores the necessity for a refined approach towards the integration of the mirrors in the vehicle's dimensional assessment. Generally, this will only affect the accuracy of yaw angle estimation, as discussed in Section 3.2.3, but it has little effect on the position error. Nonetheless, future

enhancements could involve the development of a methodology to differentiate between mirror reflections and the actual chassis of the vehicle in the point cloud data. This would enable a more precise dimensional analysis and reduce the overestimation of the bounding box size, thereby improving the integrity of the yaw angle estimations.

Chapter 4

Proposed V2I-Based Cooperative Localization System

In the previous Chapter, we analyzed the ability of the roadside LiDAR in terms of positioning accuracy for vehicles. We want to further the accuracy, reliability, and effective range of a roadside LiDAR as a localization source for CAVs. The key difference between connected and non-connected vehicles is that connected vehicles can communicate with other participants in the C-ITS. Such communication plays a pivotal role in modern transportation systems, offering a multitude of benefits. A number of standard messages are being exchanged between the ITS-S, which enhances road safety by enabling vehicles to communicate with each other, pedestrians, and infrastructure, providing real-time alerts about potential hazards and collision risks. Among them, we found the Cooperative Awareness Messages (CAM) are suitable for an RSU to acquire information about a connected vehicle such that it can provide more accurate and reliable localization results for the vehicle.

CAM are short messages broadcasted by vehicles to other ITS-S to provide real-time information about their state, such as position, speed, heading, and dynamics. The primary purpose of CAM is to enhance the awareness of a vehicle's immediate environment, contributing to road safety and traffic efficiency. CAMs are typically transmitted using Dedicated Short-Range Communications (DSRC) or Cellular-V2X (C-V2X) technologies, ensuring low latency and high reliability essential for safety-critical applications. According to the European Norm of Cooperative Awareness Basic Service compiled by ETSI [65], a CAM typically includes the following information:

- Basic Vehicle Information: Includes the vehicle's ID, type, and dimensions.
- Dynamic State Information: Details about the vehicle's current state, like speed, heading, and acceleration.
- Positioning Information: Reference location data, typically obtained from GPS or other self-localization techniques.
- Optional Data: Can include vehicle's intention, emergency vehicle warnings, etc.

The detailed design of how we propose to use CAM to enhance roadside LiDAR-based cooperative localization is depicted in Figure 4.1. In the proposed method, we assume that the RSU which can communicate with connected vehicles are equipped with LiDAR. Connected vehicles will send their information via CAM which the RSU can receive and interpret. Compared to conventional roadside LiDAR vehicle detection and tracking systems, the proposed system added the bounding box estimation and position refinement modules to provide more accurate location data for connected vehicles. The bounding box estimation algorithm and its performance analysis have been discussed in the previous section. The position refinement module is based on V2I communications, which will

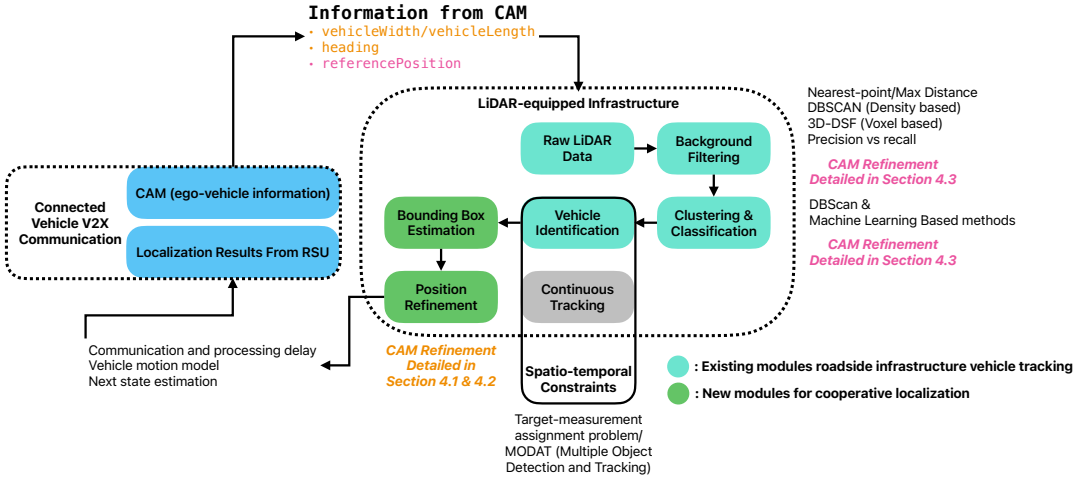


Fig. 4.1: Semantic Diagram of the Proposed System

be discussed in the following sections divided by the type of the information.

4.1 Prior Information on Vehicle Size via V2I Communication

Considering that the size of the vehicle does not change over time, we proposed to leverage V2I communication and let the RSU acquire the dimension information from CAM of the connected vehicle so that the RSU can compute a more accurate localization result to help the vehicle self-localization. After knowing the ground truth dimensions of a vehicle's bounding box, the RSU can adjust the estimated bounding box precisely by aligning the bounding box to one of the corner points. Among the four corner points determined by the L-shape fitting algorithm, the corner closest to the roadside LiDAR is the most reliable one to align with for a more accurate position. We designate this point as the *alignment point*. Aligning the ground truth bounding box with this corner allows us to refine the vehicle's position. The detailed algorithm is listed in Algorithm 1.

Algorithm 1 Bounding Box Size Correction Algorithm

Require: Estimated bounding box B, vehicle size $\langle width, height \rangle$, roadside LiDAR location M
Ensure: size corrected localization result C
  $P \leftarrow$ nearest vertex in B to M ▷ The alignment point
  $P_l, P_r \leftarrow$ two adjacent vertexes of P in B
  $\boldsymbol{L} \leftarrow$ NORMALIZED$(P_l - P)$
  $\boldsymbol{R} \leftarrow$ NORMALIZED$(P_r - P)$
  if $\|P_l - P\| < \|P_r - P\|$ **then**
    $C \leftarrow P + (\boldsymbol{L} \cdot width + \boldsymbol{R} \cdot height)/2$
  else
    $C \leftarrow P + (\boldsymbol{R} \cdot width + \boldsymbol{L} \cdot height)/2$
  end if

After applying the size correction algorithm, we ran the same pilot experiment and regenerated a new error heat map shown in Figure 4.2. Upon comparing with Figure 3.3, it

becomes evident that although accuracy degraded in some instances, the overall precision markedly improved. Specifically, between the distances of 6 m and 36 m, we achieved an accuracy within 0.1 m for most angles, satisfying the requirements for vehicle localization. When comparing Figure 4.2 with Figure 3.5, the off-by-90°error happens when the places where the error increases.

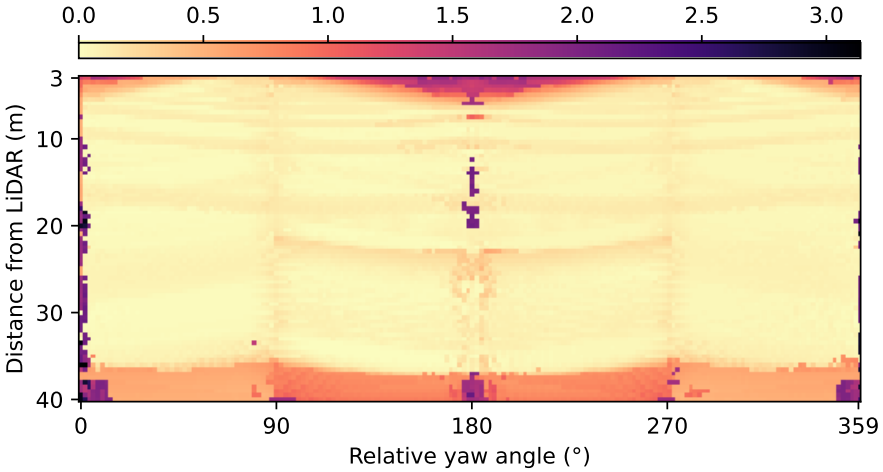


Fig. 4.2: Error of the Center Point in Distance (m)

A LiDAR frame is inspected for demonstration the size correction algorithm in Figure 4.3. In this configuration, only the front part of the vehicle is sensed by the roadside LiDAR due to self-occlusion, the estimated bounding box is much smaller than the actual size. Therefore the estimated center point is also distant from the ground truth. By applying the size correction algorithm, which align the actual-sized rectangle to the front and left edge of the bounding box, we get the size-corrected center, which much closer to the ground truth.

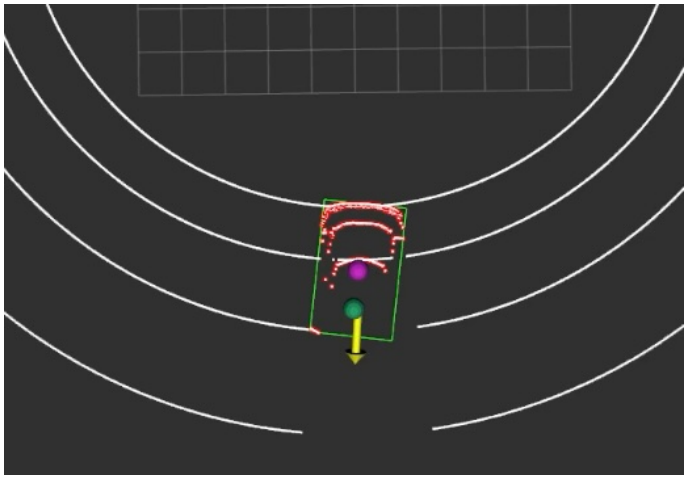


Fig. 4.3: A Case of Partial Vehicle Perception and Size Correction Algorithm In-Action
The green rectangle is the estimated bounding box before the size correction algorithm, with the purple point being its center point. The green point is the center point after the size correction algorithm. The yellow is the ground truth of the vehicle's pose. It is evident that the corrected center point is much closer to the ground truth than the non-corrected one.

4.2 Real-Time Information on Vehicle Yaw Angle via V2I Communication

The off-by-90°problem discussed in Section 3.2.3 is also critical since the roadside localization degrades significantly when it happens. The nature of this problem is that we cannot correctly identify the longer edge of the bounding box so that we always get the direction perpendicular to the ground truth yaw angle. Thankfully, we can use the real-time information from the CAM to give the algorithm a heuristic on the approximate yaw angle of the vehicle. According to the European Norm [65], a connected vehicle will broadcast its heading with the heading accuracy of the vehicle movement in the CAM. The RSU can then use this heading value to compare with the direction of the two edges: the edge that has a smaller intersection angle with the heading angle is determined to be the longer edge so that this edge gives the yaw angle. As long as the error in yaw angle estimation is less than 45°, this method can determine the correct heading edge.

4.3 Real-Time Information on Vehicle Reference Position via V2I Communication

The real-time reference vehicle position will be used to improve the efficiency and performance of background filtering. When performing background filtering, we cannot guarantee precision and recall simultaneously. By letting the RSU know the rough position of the vehicle, the RSU can only process the proximity to that rough position to reduce the false positives. Moreover, all the points out of proximity do not need to be processed, thus reducing the computation time and enabling more powerful LiDAR models to be used on the roadside to provide highly accurate localization results.

Chapter 5

Simulation Experiment of Roadside LiDAR-Based Localization

5.1 Implementation of Roadside Position Estimation Module

The roadside module used in this thesis is written in C++, with the significant dependencies being `rclcpp` for communicating with ROS and `pcl` for manipulating point cloud data. The project generates two main executables which can be run via the `ros2 run` command: `capture`, and `listener`.

`capture` is for background LiDAR frame capturing. It accepts one argument as the LiDAR model and is used to find the corresponding LiDAR topic name. When using this executable to capture the background frame, there should be no vehicle or pedestrian in the effective range of the LiDAR. We assume that LiDAR is publishing the point cloud data to the ROS topic named /[model_name]/pointcloud. After executing `capture`, it will immediately exit and save the background frame point cloud data to `reference_[model_name].pcd`. The `.pcd` file contains the 3D coordinates of the background points and the metadata for the frame.

In `listener`, we implemented the main algorithm for roadside positioning. It also accepts one argument to specify the model of the LiDAR sensor. In order to perform background filtering, the LiDAR used to capture background frames has to be the same as the LiDAR used on the RSU when sensing vehicles. `listener` listens to /[model_name]/pointcloud and /awsim/ground_truth/vehicle/pose. The detailed listened topics and their data type can be found in Table 5.1.

Table 5.1: Subscription List of `listener`

Topic Name and Data Type	Description
/[model_name]/pointcloud `sensor_msgs::msg::PointCloud2`	Point cloud data captured from the roadside LiDAR.
/awsim/ground_truth/vehicle/pose `geometry_msgs::msg::PoseStamped`	Ground truth of the vehicle pose.

In `listener`, we first read the reference frame generated by `capture` based on the provided LiDAR model in the argument. Then, we create a k-d tree [66] with the reference frame. The k-d tree is a multidimensional balanced search tree that can partition the space efficiently to be queried in an on-line manner. The space complexity of this data structure is $\mathcal{O}(n)$. Both "insert" and "search" actions have an average of $\mathcal{O}(\log n)$ time to perform,

so the total time consumed to build the k-d tree is $\mathcal{O}(n \log n)$, assuming the total number of points is n. After we built the k-d tree for the reference frame, given a certain point, we can query the nearest point to the given point in $\mathcal{O}(\log n)$.

Upon receiving a LiDAR frame from the roadside LiDAR, we first perform background filtering to obtain the vehicle points. A threshold ε is set for this step. For each point in the perceived LiDAR frame, we query the nearest point in the reference frame; then, we calculate the distance from the perceived point to its nearest point. If the distance is less than ε, we verdict this point is a background point; otherwise, it is a vehicle point. After acquiring the filtered vehicle points, we project all the points to the 2D plane by removing the height value of the points. Then, the L-Shape Fitting Algorithm was applied to the 2D points to get the estimated bounding box. Since the roadside LiDAR can be installed at different locations, a displacement vector is applied to the estimated bounding box according to the position of the roadside LiDAR.

Table 5.2: Publication List of `listener`

Topic Name and Data Type	Description
/roadside/boundingbox `geometry_msgs::msg::PolygonStamped`	The estimated bounding box from RSU (before size correction).
/roadside/size_corrected `geometry_msgs::msg::PolygonStamped`	The estimated bounding box with ground truth bounding box size corrector.
/roadside/filtered `sensor_msgs::msg::PointCloud2`	Background filtered point cloud for the perceived vehicle.

According to Section 4, we also implemented the position refinement algorithm to further improving the accuracy of the roadside position module. `listener` publishes the localization results together with the filtered vehicle points through ROS messages, details can be found in Table 5.2.

5.2 Experiment Setup

To evaluate the efficacy of our proposed method in a practical setting, we employed an Autoware and AWSIM co-simulation environment. We used an AWSIM map based on the Nishishinjuku district in Tokyo, Japan, for this experiment. Figure 5.2 offers an aerial perspective of the 3D test environment. The 3D model and the HD Point Cloud Map are provided by Tier IV[*1], which are required to perform NDT scan matching-based localization. The detailed parameters and specifications are in Table 5.3.

As illustrated in Figure 5.1, our experimental setup is organized in the following manner: in this simulation-based experiment, we used two different models of LiDAR to compare the localization results, VLP-16 and VLP-32C. The number denotes how many lines of lasers are inside the sensor. Usually, more laser lines provide more resolution and increased detection accuracy, but it also increases the data throughput and number of points that need to be processed. The technical specification of the three LiDAR models used in the experiment can be found in Table 5.4.

We initially positioned multiple overlapping LiDAR models at the exact location and recorded their individual reference frames. Subsequently, we activated AWSIM-Autoware co-simulation and manually navigated the vehicle along the designated road. Before start-

[*1] https://tier4.jp/en/

Autoware	`awsim-stable` Branch
AWSIM	v1.1.0
ROS	ROS 2 Humble
Test Vehicle Model	Lexus RX450h 2015
Vehicle Dimensions	Width: 1895 mm, Length: 4890 mm
Vehicle LiDAR Model	Velodyne VLP-16
Normal Distribution Voxel Grid Resolution	2.0 m
Roadside LiDAR Model	Velodyne VLP-16, VLP-32C, VLS-128
Roadside LiDAR Height	2.0 m
LiDAR Reporting Frequency	10 Hz
Roadside LiDAR Pitch and Roll Angle	0°

Table 5.3: Simulation Experiment Parameters and Specifications
The specification for testing computer is the same as in Table 3.1.

Table 5.4: Technical Specifications of LiDAR Models

LiDAR Model	Range	Vertical FoV	# of Points in the Reference Frame
VLP-16	100 m	30° (-15°∼15°)	24769
VLP-32C	200 m	40° (-25°∼15°)	55671
VLS-128	300 m	40° (-25°∼15°)	223680

ing to drive the testing vehicle, we give the initial guess for vehicle self-localization and ensure that the initial localization is always aligned with the actual position. Then, we deployed the roadside localization module as discussed in Section 5.1 to listen to raw point cloud output topics with pre-existing knowledge of the reference frames and the vehicle's bounding box dimensions.

Because the vehicle is manually controlled, we cannot guarantee the exact input for each trial if we want to execute the same experiment multiple times. Since the performance of NDT scan matching highly depends on the initial guess, and the initial guess for localization is provided by the localization results from the last frame, the localization error will vastly differ from different trails, even if we always drive from the same start to the same end. In order to solve this, we capture the `rosbag` for raw data during the experiment to be able to reproduce the same exact experiment over and over again. The data we collected contains the roadside LiDARs' raw data, the ground truth of the vehicle's position, and Autoware's localization results, as shown in Figure 5.1.

The vehicle-mounted and roadside LiDAR systems operate at a frequency of 10 Hz, thus delivering localization results at the same rate. Replaying this `rosbag` allows us to reproduce the same conditions across multiple trials. Upon replaying the recorded `rosbag`, we use our roadside localization module to generate localization results from the RSU; then, we aggregate position data from the ground truth, vehicle self-localization, and roadside localization modules. The error metric we used in this experiment is Mean Absolute Error (MAE), which is the distance between the ground truth and the localization results.

In the Autoware localization framework, an Extended Kalman Filter (EKF) is used to fuse NDT scan matching with other odometers. However, in AWSIM, the Global Navigation Satellite System (GNSS) data and the onboard Inertial Measurement Unit (IMU) consistently report accurate results with no errors. Therefore, fusing these additional data sources would result in unrealistically accurate localization. Thus, we turned off

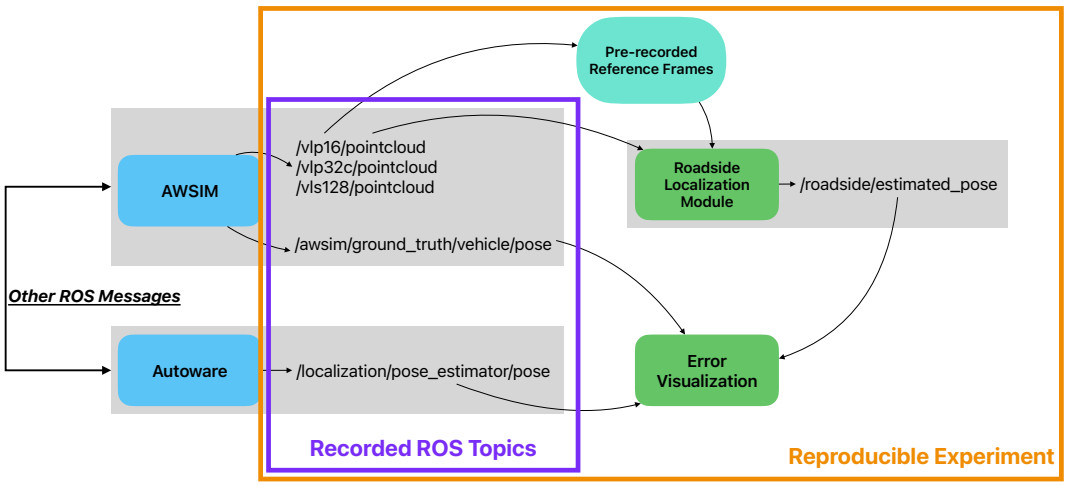


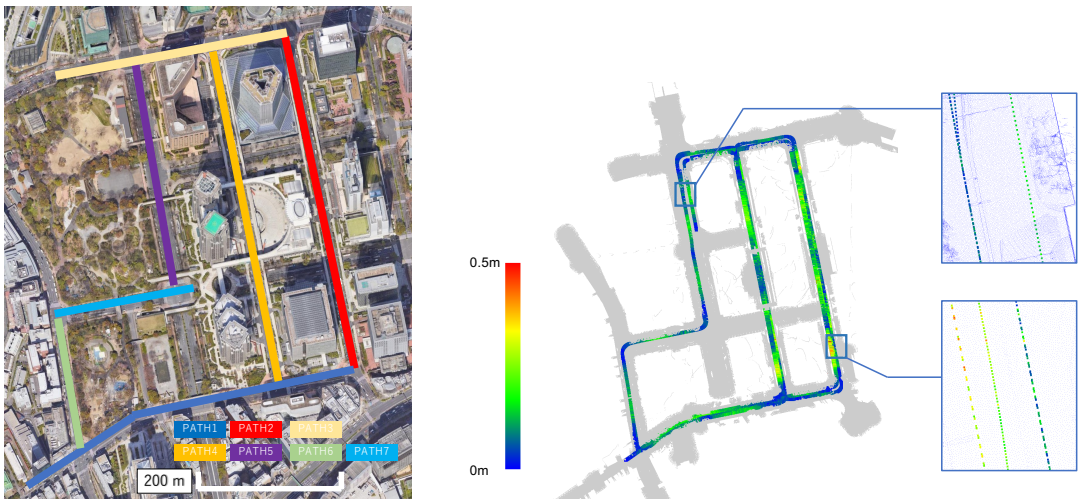
Fig. 5.1: The Data Flow of the Simulation Experiment
To be able to execute the same experiment, we only manually drove the vehicle once, during which we recorded the roadside LiDAR's raw data, the vehicle's ground truth position, and the ROS messages generated by Autoware. For subsequent experiments, we can adjust the algorithms and parameters used in the roadside localization module and replay the `rosbag` with the updated roadside module without having to manually drive the vehicle again. Thus, we eliminated the human factors that introduce different error patterns for multiple experiment trials.

input sources other than NDT scan matching for a more realistic evaluation.

5.3 Evaluation and Analysis

Before we perform the experiment, we want to know the performance of NDT-based self-localization in the Nishishinjuku map to decide where to perform the experiment. Matsumoto *et al.* [67] collected localization error in Nishishinjuku using an AWSIM and Autoware co-simulation environment. Inside AWSIM, they used a vehicle mounted with a Velodyne VLP-16 to perform localization and moved the vehicle across the entire map to cover all the major roads in the map. The testing vehicle reports the LiDAR scans via ROS messages to Autoware, where the NDT-based localization is performed. Then, they compared the localization results with the ground truth position to create and visualize the localization error, as depicted in Figure 5.2.

Nishishinjuku is a feature-rich area with scattered buildings distributed evenly in the map. Therefore, for the majority of the roads, the map-based localization works well. However, we still observed some decline in localization accuracy on specific roads, namely the southern parts of Path 2 and Path 4. This reduction in accuracy could be due to the fact that the road is flanked by two tall buildings, limiting the vehicle's ability to capture longitudinal features and thereby introducing localization error. Existing literature has also discussed how a lack of longitudinal features can lead to a decrease in localization performance [17]. Besides we also evaluated and compared the localization accuracy in a tunnel environment in the Nishishinjuku map. The LiDAR was set up at a height of 2 m and oriented with zero roll and pitch angles for all scenarios. This is because such configuration is also used in roadside LiDAR detection and tracking tasks [38] and is considered optimal.



(a) Experimental Area (b) Localization Error

Fig. 5.2: Existing Study on Evaluation of Localization Error in the Nishishinkjuku Map [67].
We can observe that the southern parts of Path 2 and Path 4 have the highest error value among all major roads in this map. Based on this, we decided to perform our experiment on these two areas in our first two scenarios.

5.3.1 Scenario 1: Range Test

In this scenario, the vehicle was driven through the southern part of path 2 in Figure 5.2. The position of the roadside LiDAR and the selected road is shown in Figure 5.3.

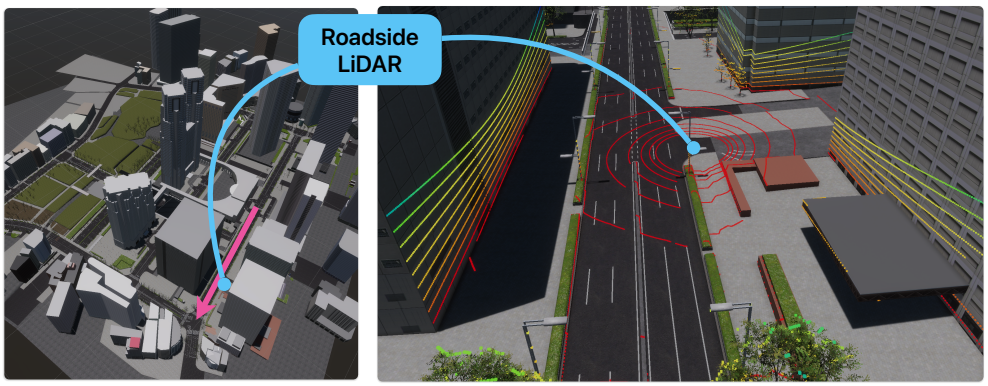


Fig. 5.3: Configuration of the Scenario 1.
The left picture shows the panorama of the Nishishinjuku area. The pink arrow in the right picture is where we conducted the experiment for scenario 1, and the blue dot is where we put the roadside LiDAR. The LiDAR visualization of the roadside LiDAR used in this figure is VLP-16.

In the experimental results Figure 5.4, we can clearly see that our proposed method significantly improves the NDT-based localization on the target road in the effective range. The graph's two outlier points at around 17 m were due to the off-by-90° issue, as discussed in Section 3.2.3. This did not happen in the pilot experiment because there was no

occlusion in the pilot experiment. However, in a real-world setting, various occlusions like shrubbery, street lamps, and elevated roadways can hinder LiDAR's ability to capture points, making our method more susceptible to errors. However, if we use a more powerful LiDAR model, VLP-32C, no extreme error will happen on the target road, enhancing both stability and accuracy.

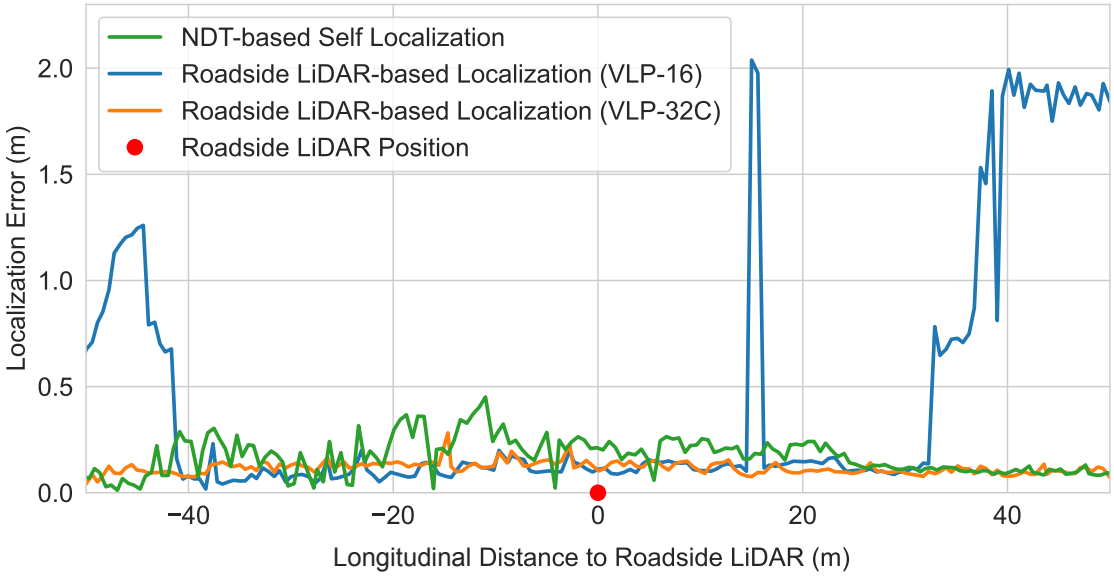


Fig. 5.4: Localization Error Comparison in Scenario 1
The x-axis is the longitudinal distance to the roadside LiDAR, and the y-axis represents the localization error in MAE. The effective range for VLP-16 is around from -30 m to 30 m. Two outliers at around 17 m are caused by off-by-90° issue, which is mitigated later in Figure 5.5.

Even though the off-by-90° issue rarely happens in the effective range, once it happens, it may create up to 2 m of errors, which is unacceptable for localization. To solve this off-by-90° issue, which is observed when using VLP-16 in this scenario, we implemented the heuristic angle search method discussed in Section 4.2 to let the RSU not have to rely on determining the yaw angle by the longer edge of the estimated bounding box. Figure 5.5 shows the localization results for VLP-16 after we implemented heuristic angle search. We can confirm that the off-by-90° issue no longer exhibits.

We further examine the data in the effective range for the two LiDAR models presented in Figure 5.6. In the range from -36 m to 36 m, both the VLP-16 and VLP-32C models consistently exhibit sub-30 cm level accuracy. Notably, the VLP-32C model extends this high level of accuracy across a broader range, from -50 m to 50 m, outperforming NDT-based localization in a statistical sense.

Table 5.5: The MAE of NDT-Based Localization and the Proposed Method with Two Different LiDAR Models for Scenario 1.

Range	NDT-based Self Localization	Proposed Method	
		VLP-16	VLP-32C
-36 m to 36 m	0.1927	0.1724	0.124
-50 m to 50 m	0.1704	0.4596	0.1167

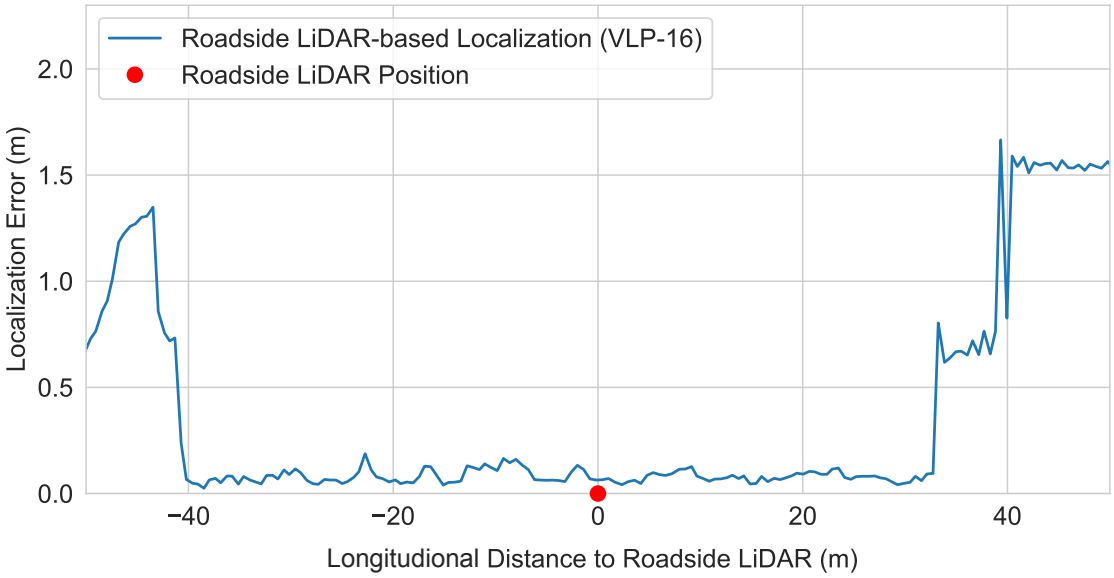


Fig. 5.5: Localization Error for VLP-16 with Heuristic Angle Search

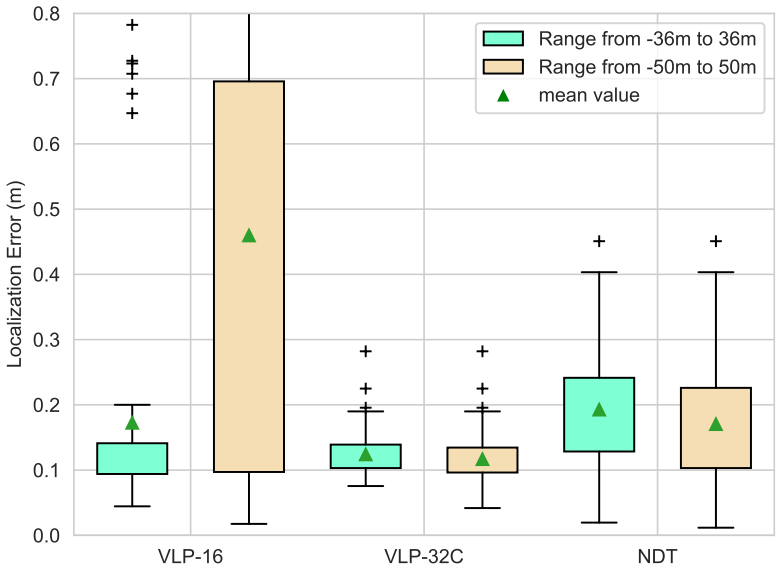


Fig. 5.6: Box Plot of MAE for Different Localization Methods in Scenario 1
The boxes represent 25th percentile to 75th percentile.

5.3.2 Scenario 2: Multiple Trials

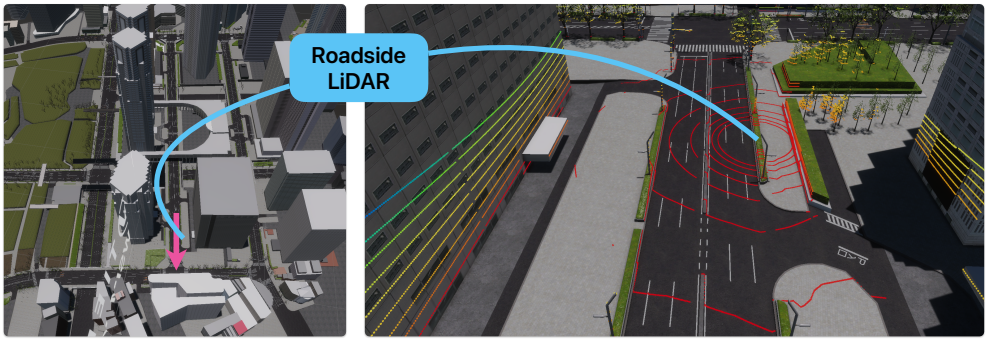


Fig. 5.7: Configuration of the Scenario 2.

For the second scenario, we moved to the southern part of Path 4 in Figure 5.2. The configuration of the LiDAR is shown in Figure 5.7. In this scenario, we performed multiple experiments to confirm that our method preserves advantage across different trails in certain areas.

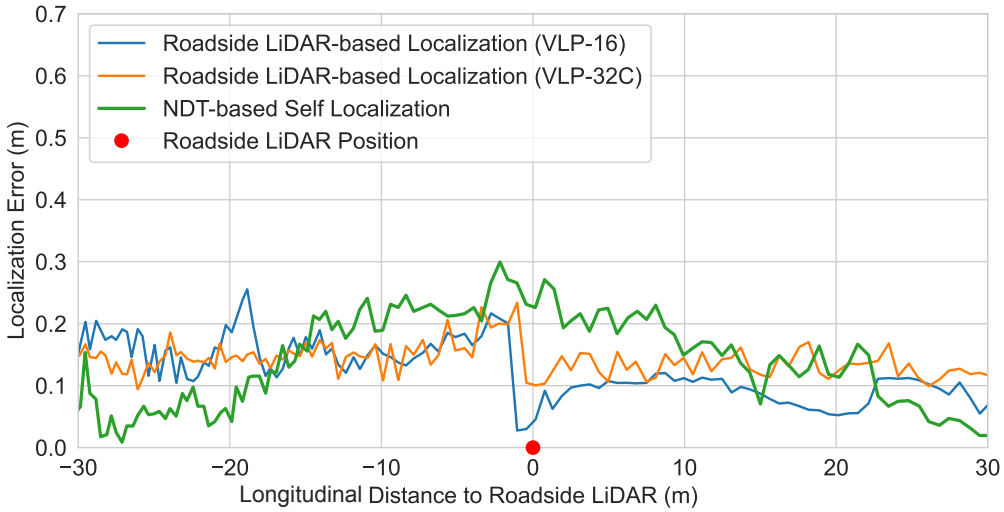


Fig. 5.8: Localization Error Comparison for Scenario 2

Figure 5.8 shows the results from one of the trials. The results show that the NDT scan matching-based localization will accumulate the error when the features become not prominent enough. Starting from -30 m in Figure 5.12, the localization error grew until 0 m. In Figure 5.9, we visualized the data collected over ten trials on the same road. We can observe that after 0 m, our method gave more accurate localization results than NDT-based localization, especially for VLP-16. We can observe that sometimes, the VLP-16 has better localization accuracy than the VLP-32C. This is because the VLP-32C has more lines of laser beams, thus more vertical resolution; the rear-view mirrors are sensed by the VLP-32C more often than the VLP-16. The accuracy will be worse when the rear-view mirror is captured in the LiDAR frame. The same effect can also be seen in Figure 4.2.

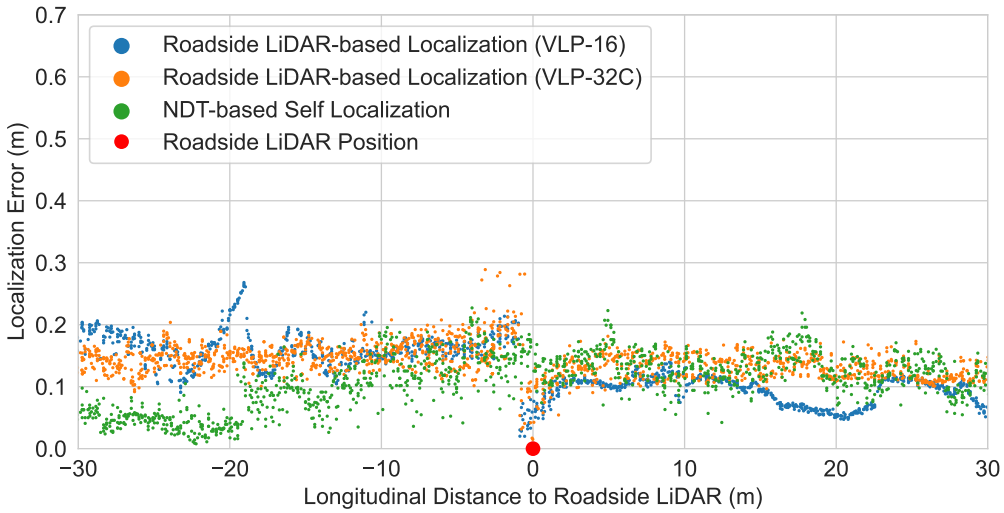


Fig. 5.9: Localization Error Comparison for Scenario 2 for Ten Trials

Based on the data used in Figure 5.9, we generated the box plot from 0 m to 30 m in Figure 5.10. We found that the NDT-based localization works better in scenario 2 than in scenario 1. This can be also confirmed by Figure 5.2b. Nevertheless, by using VLP-16 as the roadside LiDAR, we can also acquire better localization results from 0 m to 30 m, and both of the LiDAR models achieve better stability over the NDT-based localization.

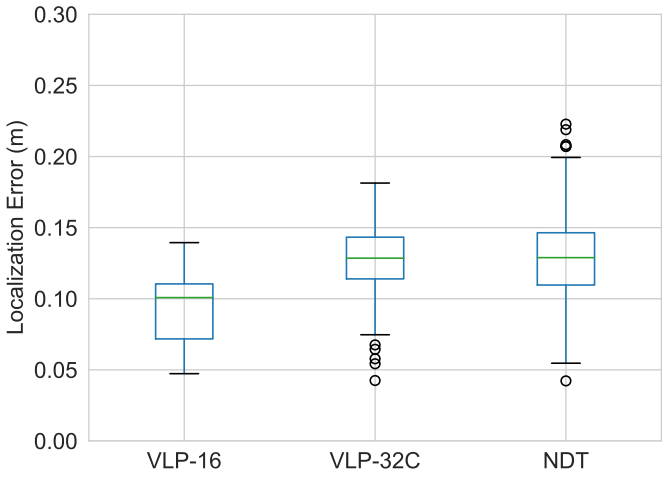


Fig. 5.10: The MAE of NDT-Based Localization and the Proposed Method for Scenario 2.
In this scenario, the NDT-based localization performs better than in scenario 1 due to local features. By using roadside LiDAR, we can still achieve more accurate localization results and better stability in certain ranges.

5.3.3 Scenario 3: Tunnel

In this scenario, we put the roadside LiDAR on the sidewalk under a tunnel, as illustrated in Figure 5.11. In the tunnel, the features are limited compared to other places. It is very evident that after entering the tunnel, the vehicle's self-localization becomes inaccurate

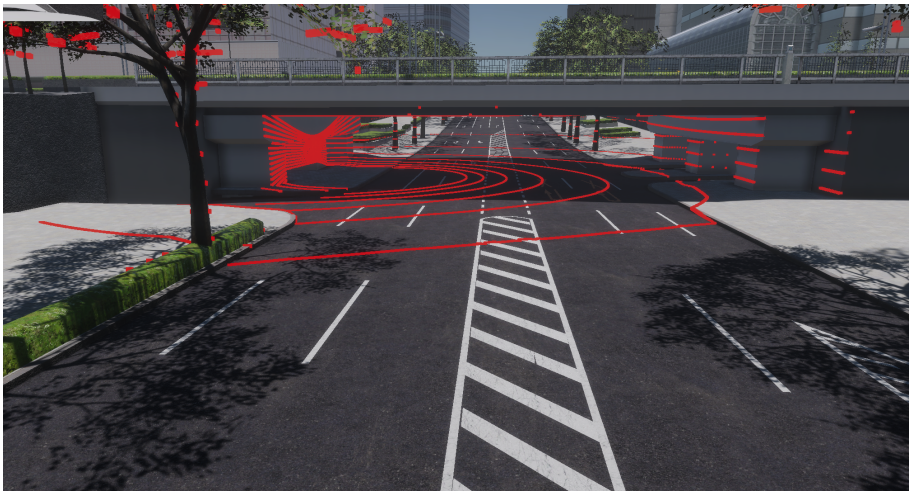


Fig. 5.11: Configuration of the Tunnel Scenario.

immediately. The NDT-based localization highly depends on the accuracy of the initial guess, and the initial guess is given by the localization result from the last LiDAR scan. As shown in Figure 5.12, after the vehicle entered the tunnel for a while, the NDT algorithm failed to converge, thus unable to generate more localization results. On the other hand, our methods do not need to rely on the features in the map and, therefore, have similar performance as in the previous two scenarios and outperform the NDT-based localization in the tunnel scenario. This renders it possible to use our method in tunnel scenarios to enhance the localization accuracy for connected vehicles and prevent the NDT algorithm from diverging by providing accurate initial guesses.

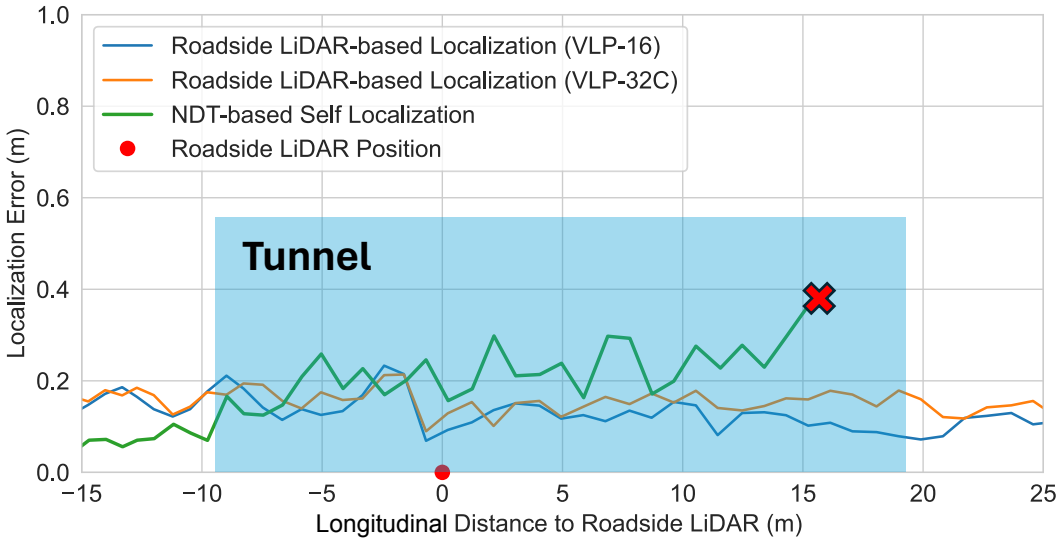
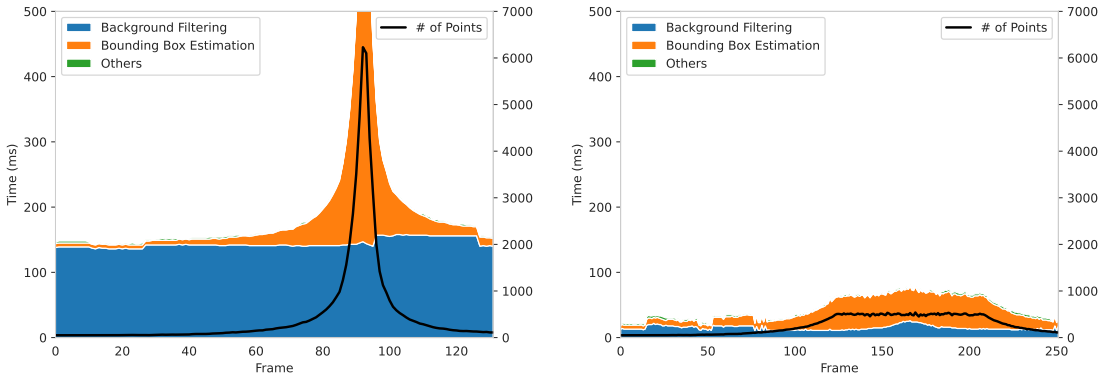


Fig. 5.12: Localization Error Comparison for Scenario 3
From around -10 m to 20 m is the tunnel area, as labeled by the blue box. The roadside LiDAR is placed in the tunnel so that it can sense the vehicles passing through the tunnel. After the vehicle entered the tunnel, the error of NDT-based self-localization began to ramp up. While the error accumulates after 15 m, the NDT algorithm cannot converge, thus not generating localization results.

5.3.4 Performance Consideration



(a) Unoptimized (b) Optimized

Fig. 5.13: Time Consumption Breakdown for VLS-128
The y-axis represents the total time used for processing a frame, including converting the ROS messages into point cloud data, background filtering, and running the L-Shape Fitting Algorithm. The x-axis is the sequential frame number of the experiment. In the simulation experiment, while driving the vehicle, the vehicle first approaches the roadside LiDAR and then gets further to the LiDAR. During this process, the number of perceived points increases and decreases before hitting the maximum at around frame number 95.

Based on the experimental results, a more powerful LiDAR module as a roadside LiDAR can provide more accurate localization results. However, we must also consider the data processing time that each LiDAR consumes since real-time is very important in cooperative localization. With less computational delay, the connected vehicle can get more up-to-date localization results. We analyzed the time consumption breakdown for a VLS-128 LiDAR as a roadside LiDAR in Figure 5.13. The average time consumed is about 150 ms with a maximum of 180 ms. With such computation time, we failed to provide real-time roadside localization results since the LiDAR reporting frequency is 10 Hz, requiring a maximum processing time of 100 ms. Because the number of points per frame is so large, the majority of the time is consumed in the background filtering process, even if the time complexity of the algorithm is only $\mathcal{O}(\log n)$.

We implemented the following two optimizations to reduce the time consumption while maintaining the accuracy of roadside localization. First, to optimize the background filtering process, we utilize the vehicle reference position we discussed in Section 4.3. The connected vehicle will periodically broadcast its reference location to the nearby ITS-S, including the RSU. Based on the performance of the NDT scan matching-based self-localization, we can safely assume that the reference position will not exceed 2 m. By receiving the vehicle's reference position, we can add an "important zone" around that position. This tells us that, outside the important zone, there will not be any points belonging to a connected vehicle, so we can only process the points inside the important zone. We set the radius of the important zone to 3 m to consider the error of vehicle self-localization and the size of the vehicle. According to Figure 3.4, we can see that the maximum of the perceived vehicle points for VLP-16 is around 1200, whereas the total number of points per frame is 14000. Ideally, if most of the background points are excluded before background filtering, we can save up to 90% processing time at the background

filtering stage.

Besides, we notice that when the number of perceived points increases, the time used on bounding box estimation also takes a considerable portion of the total time. So, we also tried to reduce the time used in the L-Shape Fitting Algorithm without reducing the accuracy. According to Section 3.2.2, the number of points actually is not proportional to the accuracy, as long as the sensed silhouette does not change. In other words, surplus points do not contribute to the accuracy of the localization. We set the target number of points to 500 so that if the perceived number of points exceeds 500, we will randomly drop points until the remaining points are 500. By doing this, we control the maximum number of points used for the L-Shape Fitting Algorithm, thus reducing the time used for bounding box estimation without degrading accuracy.

Chapter 6

Conclusion

6.1 Summary

In this thesis, we proposed a roadside LiDAR-based cooperative localization method for connected vehicles to enhance localization accuracy where vehicle self-localization falls short. An initial pilot experiment was conducted to gauge the effectiveness of roadside LiDAR in vehicle perception and identify possible sources of errors. We then introduced a novel method to let connected vehicles send their dimensions to the RSU via V2I communication. This enables the RSU to provide a more accurate localization result for these connected vehicles. The experimental result of the realistic simulation confirms that our proposed method yields better localization results than NDT scan matching-based vehicle self-localization within the effective range. We further proposed to extend the ability of RSU to acquire more real-time data from connected vehicles by letting the RSU read information from connected vehicles' CAM. By doing this, we further improved the accuracy and performance of roadside LiDAR-based localization.

6.2 Future Work

Firstly, we only consider single-vehicle scenarios in this study. When multiple vehicles are in the visible range of the roadside LiDAR, the occlusion between the vehicles will increase. Therefore, we cannot guarantee that the LiDAR can perceive the whole outline of the vehicle when occluded. We can introduce a confidence value for the localization result based on the perceived number of points to partially solve this problem. Besides, identifying the same vehicle across multiple LiDAR frames also becomes important. We can use existing vehicle tracking algorithms to solve this problem. Future research and evaluation should be done to address these added complexities when multiple vehicles are present.

Moreover, we have only examined a limited number of positioning strategies for the roadside LiDAR. On the one hand, it is necessary to find where the NDT-based vehicle self-localization is degraded so that it can be improved by roadside infrastructure; on the other hand, the differences between the results of scenario 1 and scenario 2 suggest that the position of the roadside LiDAR is also influential to the roadside localization accuracy. Other researchers have also recently drawn attention to the importance of roadside LiDAR placement issues [68]. We want to propose a framework that can systematically evaluate the localization accuracy of both vehicle self-localization and roadside localization methods at different locations and for different roadside sensor placements.

Finally, we aim to enhance our pose estimation technique by incorporating an Extended Kalman Filter (EKF). Given that the positional data derived from RSU is often tainted with noise, the application of an EKF is anticipated to significantly refine the accuracy

of our measurements. EKF excels in extracting a cleaner signal from the noisy data, thereby ensuring reliability even amidst considerable measurement and process noise. Furthermore, we intend to address issues related to communication and processing latency by adjusting for these delays in correlation with the vehicle's velocity. This proactive approach will allow for real-time compensation and improved precision in our system's ability to determine the vehicle's exact location and orientation.

Publications

(1) **Yuze Jiang**, Ehsan Javanmard, Jin Nakazato, Manabu Tsukada, and Hiroshi Esaki. 2023. "Roadside LiDAR Assisted Cooperative Localization for Connected Autonomous Vehicles." In Proceedings of 2023 International Conference on Intelligent Computing and its Emerging Applications. (**Best Paper Award**)

(2) **Yuze Jiang**, Ehsan Javanmard, Manabu Tsukada, and Hiroshi Esaki. 2024. "Roadside LiDAR-Based Cooperative Localization with V2I Communication Enhancements." In Proceedings of 9^{th} Digital Signal Processing for In-Vehicle Systems and Safety (Under Submission)

(3) Ye Tao, **Yuze Jiang**, Pengfei Lin, Manabu Tsukada, and Hiroshi Esaki. 2023. "zk-PoT: Zero-Knowledge Proof of Traffic for Privacy Enabled Cooperative Perception." In 2023 IEEE 20th Consumer Communications & Networking Conference (CCNC), 261–68.

(4) Ye Tao, Ehsan Javanmardi, Pengfei Lin, Jin Nakazato, **Yuze Jiang**, Manabu Tsukada, and Hiroshi Esaki. 2023. "Zero-Knowledge Proof of Traffic: A Deterministic and Privacy-Preserving Cross Verification Mechanism for Cooperative Perception Data." IEEE Access.

References

[1] Rudi Volti and Maxwell G Lay. Ways of the world: A history of the world's roads and of the vehicles that used them. *Technol. Cult.*, 35(3):608, July 1994.

[2] Keshav Bimbraw. Autonomous cars: Past, present and future a review of the developments in the last century, the present scenario and the expected future of autonomous vehicle technology. In *2015 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, volume 01, pages 191–198. IEEE, July 2015.

[3] Md Masud Rana and Kamal Hossain. Connected and autonomous vehicles and infrastructures: A literature review. *International Journal of Pavement Research and Technology*, 16(2):264–284, March 2023.

[4] Korea Ministry of Land, Infrastructure and Transport and Korea Expressway Corporation Research Institute. C-ITS pilot project. `https://www.c-its.kr/english/introduction.do`. Accessed: 2023-12-19.

[5] SAE International. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles. Technical Report J3016_202104, https://www.sae.org/standards/content/j3016_202104/, April 2021.

[6] Ali Gohar and Gianfranco Nencioni. The role of 5G technologies in a smart city: The case for intelligent transportation system. *Sustain. Sci. Pract. Policy*, 13(9):5188, May 2021.

[7] CAR 2 CAR Communication Consortium. C-ITS: Cooperative intelligent transport systems and services. `https://www.car-2-car.org/about-c-its`. Accessed: 2024-1-17.

[8] Shoudong Huang and Gamini Dissanayake. Robot localization: An introduction. In *Wiley Encyclopedia of Electrical and Electronics Engineering*, pages 1–10. John Wiley & Sons, Inc., Hoboken, NJ, USA, August 2016.

[9] Sampo Kuutti, Saber Fallah, Konstantinos Katsaros, Mehrdad Dianati, Francis Mccullough, and Alexandros Mouzakitis. A survey of the State-of-the-Art localization techniques and their potentials for autonomous vehicle applications. *IEEE Internet of Things Journal*, 5(2):829–846, April 2018.

[10] E G Tsardoulias, A Iliakopoulou, A Kargakos, and L Petrou. A review of global path planning methods for occupancy grid maps regardless of obstacle density. *J. Intell. Rob. Syst.*, 84(1-4):829–858, December 2016.

[11] Ricardo Roriz, Jorge Cabral, and Tiago Gomes. Automotive LiDAR technology: A survey. *IEEE Trans. Intell. Transp. Syst.*, 23(7):6282–6297, July 2022.

[12] Madhusri Maity, Sriparna Banerjee, and Sheli Sinha Chaudhuri. Faster R-CNN and YOLO based vehicle detection: A survey. In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pages 1442–1447. IEEE, April 2021.

[13] Ryan W Wolcott and Ryan M Eustice. Visual localization within LIDAR maps for automated urban driving. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 176–183. IEEE, September 2014.

[14] Jacob Lambert, Alexander Carballo, Abraham Monrroy Cano, Patiphon Narksri, David Wong, Eijiro Takeuchi, and Kazuya Takeda. Performance analysis of 10 models

of 3D LiDARs for automated driving. *IEEE Access*, 8:131699–131722, 2020.

[15] Jesse Levinson and Sebastian Thrun. Robust vehicle localization in urban environments using probabilistic maps. In *2010 IEEE International Conference on Robotics and Automation*, pages 4372–4378. IEEE, May 2010.

[16] Clemens Dannheim, Markus Maeder, Alexander Dratva, Mattia Raffero, Stefan Neumeier, and Christian Icking. A cost-effective solution for HD maps creation. In *AmE 2017 - Automotive meets Electronics; 8th GMM-Symposium*, pages 1–5. VDE, March 2017.

[17] Ehsan Javanmardi, Mahdi Javanmardi, Yanlei Gu, and Shunsuke Kamijo. Factors to evaluate capability of map for vehicle localization. *IEEE Access*, 6:49850–49867, 2018.

[18] Randall C Smith and Peter Cheeseman. On the representation and estimation of spatial uncertainty. *Int. J. Rob. Res.*, 5(4):56–68, December 1986.

[19] P J Besl and Neil D McKay. A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, February 1992.

[20] P Biber and W Strasser. The normal distributions transform: a new approach to laser scan matching. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, volume 3, pages 2743–2748 vol.3, October 2003.

[21] Tejas Zodage. Point cloud registration as a classification problem. Master's thesis, Carnegie Mellon University, August 2021.

[22] Su Pang, Daniel Kent, Xi Cai, Hothaifa Al-Qassab, Daniel Morris, and Hayder Radha. 3D scan registration based localization for autonomous vehicles - a comparison of NDT and ICP under realistic conditions. In *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pages 1–5, August 2018.

[23] Naoki Akai, Luis Yoichi Morales, Eijiro Takeuchi, Yuki Yoshihara, and Yoshiki Ninomiya. Robust localization using 3D NDT scan matching with experimentally determined uncertainty and road marker matching. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1356–1363, June 2017.

[24] Surachet Srinara, Chi-Ming Lee, Syun Tsai, Guang-Je Tsai, and Kai-Wei Chiang. Performance analysis of 3D NDT scan matching for autonomous vehicles using INS/GNSS/3D LiDAR-SLAM integration scheme. In *2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, pages 1–4. ieeexplore.ieee.org, March 2021.

[25] Yin-Chiu Kan, Li-Ta Hsu, and Edward Chung. Performance evaluation on Map-Based NDT scan matching localization using simulated occlusion datasets. *IEEE Sensors Letters*, 5(3):1–4, March 2021.

[26] Sheng-Cheng Lee, Victor Lu, Chieh-Chih Wang, and Wen-Chieh Lin. LiDAR-Based localization on highways using raw data and Pole-Like object features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 230–237, 2023.

[27] ETSI. Intelligent transport system (ITS); vehicular communications; basic set of applications; collective perception service; release 2. Technical Report TS 103 324, June 2023.

[28] Jianqing Wu, Hao Xu, and Jianying Zheng. Automatic background filtering and lane identification with roadside LiDAR data. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE, October 2017.

[29] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press, August 1996.

[30] Pouria Babahajiani, Lixin Fan, Joni-Kristian Kämäräinen, and Moncef Gabbouj. Urban 3D segmentation and modelling from street view images and LiDAR point clouds. *Mach. Vis. Appl.*, 28(7):679–694, October 2017.

[31] Matti Lehtomäki, Anttoni Jaakkola, Juha Hyyppä, Jouko Lampinen, Harri Kaartinen, Antero Kukko, Eetu Puttonen, and Hannu Hyyppä. Object classification and recognition from mobile laser scanning point clouds in a road environment. *IEEE Trans. Geosci. Remote Sens.*, 54(2):1226–1239, February 2016.

[32] Ahmad Kamal Aijazi, Paul Checchin, and Laurent Trassoudaine. Segmentation based classification of 3D urban point clouds: A Super-Voxel based approach with evaluation. *Remote Sensing*, 5(4):1624–1650, March 2013.

[33] Aleksey Golovinskiy, Vladimir G Kim, and Thomas Funkhouser. Shape-based recognition of 3D point clouds in urban environments. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2154–2161. IEEE, September 2009.

[34] Danil Prokhorov. A convolutional learning system for object classification in 3-D lidar data. *IEEE Trans. Neural Netw.*, 21(5):858–863, May 2010.

[35] Kyungpyo Kim, Chansoo Kim, Chulhoon Jang, Myoungho Sunwoo, and Kichun Jo. Deep learning-based dynamic object classification using LiDAR point cloud augmented by layer-based accumulation for intelligent vehicles. *Expert Syst. Appl.*, 167:113861, April 2021.

[36] Edouard Capellier, Franck Davoine, Veronique Cherfaoui, and You Li. Evidential deep learning for arbitrary LIDAR object classification in the context of autonomous driving. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1304–1311. IEEE, June 2019.

[37] Van-Linh Nguyen, Po-Ching Lin, and Ren-Hung Hwang. Physical signal-driven fusion for V2X misbehavior detection. In *2019 IEEE Vehicular Networking Conference (VNC)*, pages 1–4. IEEE, December 2019.

[38] Junxuan Zhao, Hao Xu, Hongchao Liu, Jianqing Wu, Yichen Zheng, and Dayong Wu. Detection and tracking of pedestrians and vehicles using roadside LiDAR sensors. *Transp. Res. Part C: Emerg. Technol.*, 100:68–87, March 2019.

[39] Jiaxing Zhang, Wen Xiao, Benjamin Coifman, and Jon P Mills. Vehicle tracking and speed estimation from roadside lidar. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:5597–5608, 2020.

[40] Edward Andert and Aviral Shrivastava. Accurate cooperative sensor fusion by parameterized covariance generation for sensing and localization pipelines in CAVs. September 2022.

[41] Motion capture systems. `https://optitrack.com/`. Accessed: 2023-12-29.

[42] K Golestan, A Jundi, L Nassar, F Sattar, F Karray, M Kamel, and S Boumaiza. Vehicular ad-hoc Networks(VANETs): Capabilities, challenges in information gathering and data fusion. In *Autonomous and Intelligent Systems*, pages 34–41. Springer Berlin Heidelberg, 2012.

[43] Sae Fujii, Atsushi Fujita, Takaaki Umedu, Shigeru Kaneda, Hirozumi Yamaguchi, Teruo Higashino, and Mineo Takai. Cooperative vehicle positioning via V2V communications and onboard sensors. In *2011 IEEE Vehicular Technology Conference (VTC Fall)*, pages 1–5, September 2011.

[44] Pengtao Yang, Dongliang Duan, Chen Chen, Xiang Cheng, and Liuqing Yang. Multi-Sensor Multi-Vehicle (MSMV) localization and mobility tracking for autonomous driving. *IEEE Trans. Veh. Technol.*, 69(12):14355–14364, December 2020.

[45] O Jetter, J Dinger, and H Hartenstein. Quantitative analysis of the sybil attack and effective sybil resistance in Peer-to-Peer systems. In *2010 IEEE International Conference on Communications*, pages 1–6. IEEE, May 2010.

[46] Xiaoning Jiang and Lingxiao Ye. Attack-resistant techniques in P2P reputation sys-

tems. In *2010 International Conference on Networking and Digital Society*, volume 1, pages 390–393. IEEE, May 2010.

[47] D Dumitriu, E Knightly, A Kuzmanovic, I Stoica, and W Zwaenepoel. Denial-of-service resilience in peer-to-peer file sharing systems. *SIGMETRICS Perform. Eval. Rev.*, 33(1):38–49, June 2005.

[48] J Dinger and H Hartenstein. Defending the sybil attack in P2P networks: taxonomy, challenges, and a proposal for self-registration. In *First International Conference on Availability, Reliability and Security (ARES'06)*, pages 8 pp.–763. IEEE, 2006.

[49] Ye Tao, Ehsan Javanmardi, Pengfei Lin, Jin Nakazato, Yuze Jiang, Manabu Tsukada, and Hiroshi Esaki. Zero-Knowledge proof of traffic: A deterministic and Privacy-Preserving cross verification mechanism for cooperative perception data. *IEEE Access*, PP(99):1–1, 2023.

[50] Alessio Fascista, Giovanni Ciccarese, Angelo Coluccia, and Giuseppe Ricci. A localization algorithm based on V2I communications and AOA estimation. *IEEE Signal Process. Lett.*, 24(1):126–130, January 2017.

[51] O Hassan, I Adly, and K A Shehata. Vehicle localization system based on IR-UWB for V2I applications. In *2013 8th International Conference on Computer Engineering & Systems (ICCES)*, pages 133–137. IEEE, November 2013.

[52] Lan-Huong Nguyen, Van-Linh Nguyen, and Yu-Hao Liu. Tracking risks from multi-path TDoA-based localization in wireless communications. In *2023 25th International Conference on Advanced Communication Technology (ICACT)*, pages 96–101. ieeexplore.ieee.org, February 2023.

[53] Chia-Ho Ou, Bing-Yi Wu, and Lin Cai. GPS-free vehicular localization system using roadside units with directional antennas. *Journal of Communications and Networks*, 21(1):12–24, February 2019.

[54] Chia-Ho Ou. A roadside unit-based localization scheme for vehicular ad hoc networks. *Int. J. Commun. Syst.*, 27(1):135–150, January 2014.

[55] Xue Liu, Tangtao Yang, Haiyang Chen, and Tony Z Qiu. Research on roadside Unit-Assisted cooperative positioning method for a connected vehicle environment. *Transp. Res. Rec.*, 2676(1):78–90, January 2022.

[56] Sinem Coleri Ergen, Huseyin Serhat Tetikol, Mehmet Kontik, Raffi Sevlian, Ram Rajagopal, and Pravin Varaiya. RSSI-Fingerprinting-Based mobile phone localization with route constraints. *IEEE Trans. Veh. Technol.*, 63(1):423–428, January 2014.

[57] Azzedine Boukerche, Horacio A B F Oliveira, Eduardo F Nakamura, and Antonio A F Loureiro. Vehicular ad hoc networks: A new challenge for Localization-Based systems. *Comput. Commun.*, 31(12):2838–2849, July 2008.

[58] Chi-Hua Chen, Chi-Ao Lee, and Chi-Chun Lo. Vehicle localization and velocity estimation based on mobile phone sensing. *IEEE Access*, 4:803–817, 2016.

[59] Yuze Jiang, Ehsan Javanmard, Jin Nakazato, Manabu Tsukada, and Hiroshi Esaki. Roadside LiDAR assisted cooperative localization for connected autonomous vehicles. In *ACM Intelligent Computing and its Emerging Applications (ICEA 2023)*, 2023. published.

[60] Xiao Zhang, Wenda Xu, Chiyu Dong, and John M Dolan. Efficient l-shape fitting for vehicle detection using laser scanners. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 54–59. IEEE, June 2017.

[61] TIER VI inc. Autoware. `https://github.com/autowarefoundation/autoware`.

[62] TIER IV inc. AWSIM. `https://github.com/tier4/AWSIM`.

[63] Robotec Ai. RobotecGPULidar. `https://github.com/RobotecAI/RobotecGPULidar`.

[64] Steven Macenski, Tully Foote, Brian Gerkey, Chris Lalancette, and William Woodall. Robot operating system 2: Design, architecture, and uses in the wild. *Science*

Robotics, 7(66):eabm6074, 2022.

[65] ETSI. Intelligent transport systems (ITS); vehicular communications; basic set of applications; part 2: Specification of cooperative awareness basic service. Technical Report ETSI EN 302 637-2, April 2019.

[66] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, September 1975.

[67] Kazuto Matsumoto, Ehsan Javanmardi, Jin Nakazato, and Manabu Tsukada. Localizability estimation for autonomous driving: A deep Learning-Based place recognition approach. In *IEEE Robotic Computing 2023*, California, USA, December 2023.

[68] Xinyu Cai, Wentao Jiang, Runsheng Xu, Wenquan Zhao, Jiaqi Ma, Si Liu, and Yikang Li. Analyzing infrastructure LiDAR placement with realistic LiDAR simulation library. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2023.

Acknowledgements

I would like to express my deepest appreciation to Professor Hiroshi Esaki and Professor Manabu Tsukada, my supervisors, for their unwavering support and invaluable guidance throughout the course of my master research. Their expertise in computer networks and autonomous driving has been instrumental in shaping both the direction and execution of this study. I am especially thankful for their willingness to challenge my ideas and push me to deepen my analytical skills.

My gratitude also goes to Professor Ehsan Javanmardi, with whom I had the privilege to work during my research. His perspectives and expertise in mapping and localization were immensely helpful. I am also thankful for the laboratory staff in the Tsukada Laboratroy, Professor Jin Nakazato, Professor Naren Bao, and Professor Muhammad Asad, for their valuable suggestions and constructive criticism during the various stages of this work.

My sincere thanks also go to my colleagues and friends at UTokyo, particularly Pengfei Lin, Shangkai Zhang, Ye Tao, Yu Asabe, Kazuto Matumoto, and Yiwei Cheng for their camaraderie, stimulating discussions, and for the fun times we shared during this journey.

I am indebted to my family, Baozhong Jiang and Jingyi Li, for their unwavering love, understanding, and support. They have been my rock during this challenging yet rewarding journey. Their belief in my abilities has been a constant source of motivation.