Characterization of Multiple 3D LiDARs for Localization and Mapping using Normal Distributions Transform

Alexander Carballo\textsuperscript{1,4} \hspace{1em} Abraham Monroy\textsuperscript{2,4} \hspace{1em} David Wong\textsuperscript{1,4} \hspace{1em} Patiphon Narksri\textsuperscript{2,4} \hspace{1em} Jacob Lambert\textsuperscript{2,4} \hspace{1em} Yuki Kitsukawa\textsuperscript{2,4} \hspace{1em} Eijiro Takeuchi\textsuperscript{2,4} \hspace{1em} Shinpei Kato\textsuperscript{3,4,5} \hspace{1em} Kazuya Takeda\textsuperscript{1,2,4}

Abstract—In this work, we present a detailed comparison of ten different 3D LiDAR sensors, covering a range of manufacturers, models, and laser configurations, for the tasks of mapping and vehicle localization, using as common reference the Normal Distributions Transform (NDT) algorithm implemented in the self-driving open source platform Autoware. LiDAR data used in this study is a subset of our LiDAR Benchmarking and Reference (LIBRE) dataset, captured independently from each sensor, from a vehicle driven on public urban roads multiple times, at different times of the day. In this study, we analyze the performance and characteristics of each LiDAR for the tasks of (1) 3D mapping including an assessment map quality based on mean map entropy, and (2) 6-DOF localization using a ground truth reference map.

Index Terms—3D LiDAR, localization, mapping, normal distributions transform, LIBRE dataset

I. INTRODUCTION

State-of-the-art vehicle navigation, in particular localization and obstacle negotiation, cannot be conceived without referring to LiDARs (Light Detection And Ranging, sometimes Light Imaging Detection And Ranging for the image-like resolution of modern 3D sensors). Ever since the first implementations of SLAM (simultaneous localization and mapping) for robotics \cite{1} and practical demonstrations on vehicles \cite{2}, LRF (laser range finders) and LiDARs have had a major role to realize high accuracy 2D occupancy maps and 3D pointcloud maps. Current MMS (mobile mapping systems) employ several LiDARs together with IMUs, cameras, GNSS, odometry, to capture with the highest fidelity all the environments elements involved in driving.

The rapid developments in the field of intelligent transport systems (ITS) have created a large demand for LiDARs for obstacle negotiation, navigation, sensing in intelligent highways, and so on. Each operational design domain (ODD) defines several key requirements expected from LiDARs: measurement range, measurement accuracy, repeatability, point density, scanning speed, configurability, wavelengths, robustness to environmental changes, sensing on adverse weather, small form factors, and costs. As such, a large number of LiDAR manufacturers have emerged in recent years introducing new technologies to address different needs \cite{13}. With so many different manufacturers and technologies becoming available, it is necessary to properly assess the characteristics and performance of each device. Furthermore, with LiDAR costs still remaining high, it can be difficult to select the best LiDAR in terms of cost to performance.

We released the LIBRE dataset covering multiple 3D LiDARs \cite{14, 15}. It features 12 LiDARs, each one a different model from diverse manufacturers, includes data from four different environments and configurations: dynamic traffic which corresponds to traffic scenes captured from a vehicle driving on public urban roads around Nagoya University; static targets related to objects (reflective targets, black car and mannequins) placed at known controlled distances, and measured from a fixed position; adverse weather consisting on static objects placed at a fix location and measured from a moving vehicle while exposed to adverse conditions (fog, rain, strong light); and indirect interference in which dynamic traffic objects are measured from a fixed position by multiple LiDARs simultaneously and exposed to indirect interference conditions. Our dataset also includes ground truth 3D pointcloud maps and vector maps (HD maps).
created by a professional mobile mapping system (MMS).

Using the dynamic traffic data of our LIBRE dataset, this study focuses on analyzing the characteristics of ten 3D LiDARs for vehicle localization and mapping, considering the LiDAR pointcloud data alone, this is, without assistance of any other navigation and dead reckoning systems such as IMU, GNSS, odometry, or vehicle speed. To the best of our knowledge, this is the first work studying so many LiDARs for vehicle navigation, covering a range of manufacturers, models, and laser configurations (the sensors are listed in Table I). To define a common ground for comparison, we use the Normal Distributions Transform (NDT)[16,17,18] in the self-driving open source platform Autoware[19] (see also Kato et al.[20]).

The contributions of this work are summarized as follows:

- A unique study of the properties for multiple 3D LiDARs for creation of 3D maps and 6-DOF localization.
- Evaluation of 3D map quality using the mean map entropy (MME) and mean plane variance (MPV) scores for each LiDAR.

This paper is structured as follows: Section II revisits NDT scan matching and its properties. Section III describes the dynamic traffic data and experimental procedures in this study. Sections IV and V discuss the actual result for mapping localization and localization. Finally, this paper is concluded in Section VI.

1 All sensor images’ not to scale and copyright owned by their respective manufacturers.

### II. Normal Distributions Transform

#### A. NDT Scan Matching

The Normal Distributions Transform (NDT) scan matching method is attributed to Biber et al.[21]. In NDT, the reference scan (pointcloud) is divided into uniform 3D grid, and each cell (voxel) keeps the mean and distribution of the sub-pointcloud assigned to it, thus the ND voxels represent normal distributions.

Following the notation by Takeuchi et al.[16], the reference pointcloud (map or model) \( M \) is a vector of \( M \) points \( \mathbf{x}_i = (x_i, y_i, z_i)^T \) with \( i \in [1..M] \), and \( x_k \) is the \( k \)-th ND voxel with \( M_k \) points. The \( k \)-th ND voxel mean \( \mu_k \) and covariance matrix \( \Sigma_k \) are given by Eq. (1):

\[
\begin{align*}
\mu_k &= \frac{1}{M_k} \sum_{i=1}^{M_k} x_{ki} \\
\Sigma_k &= \frac{1}{M_k} \sum_{i=1}^{M_k} (x_{ki} - \mu_k)(x_{ki} - \mu_k)^T
\end{align*}
\]

Given an input scan (pointcloud) \( X \) with \( N \) points, \( x_i \) with \( i \in [1..N] \), the 3D coordinate transformation of \( X \) into \( M \) is achieved by \( x'_i = Rx_i + t' \), with \( R \) the rotation matrix and \( t' \) the translation vector. Thus, the pose transformation

---

**TABLE I: LiDARs tested in this study, by manufacturer and number of channels (rings[1]) Acronyms are frame rate (FPS), vertical field-of-view (vFOV), vertical resolution (vRes), laser wavelength (\( \lambda \)), and diameter \( \phi \).**

| Channels | FPS[Hz] | vFOV[deg] | vRes[deg] | \( \lambda \)[nm] | \( \phi \)[mm] | Weight(kg) |
|----------|---------|-----------|-----------|-----------------|--------------|------------|
| VLS-128AF | 128 | 128 | 64 | 5-20 | 64 | 116 |
| VLS-128 | 128 | 128 | 64 | 5-20 | 64 | 116 |
| HDL-32E | 16 | 16 | 64 | 5-20 | 64 | 116 |
| VLP-32 | 16 | 16 | 64 | 5-20 | 64 | 116 |
| VLP-16 | 8 | 8 | 32 | 5-20 | 32 | 64 |
| Pandar-64 | 40 | 40 | 64 | 5-20 | 64 | 120 |
| Pandar-40 | 40 | 40 | 64 | 5-20 | 64 | 120 |
| OS1-64 | 1.52 | 1.52 | 16 | 16 | 1.52 | 1.52 |
| RS-Lidar32 | 1.17 | 1.17 | 16 | 16 | 1.17 | 1.17 |

- Velodyne states HDL-64S2 accuracy is \( \pm 2cm \) for 80% of channels, and \( \pm 5cm \) for the remaining; vRes for \( +2^\circ .. -8.33^\circ \) is \( 1/3^\circ \), and for \( -8.33^\circ .. -24.33^\circ \) is \( 1/2^\circ \). Minimum (or finest) resolution, as these sensors have variable angle difference between beams.
- Hesai and RoboSense state that accuracy for \( 0.3m..0.5m = \pm 0.05m \), then \( \pm 0.02m . 0.5m..200m \). Subset states accuracy for \( 0.8m .. 2m = \pm 0.03m \), for \( 2m .. 20m = \pm 0.015m \), for \( 20m .. 60m = \pm 0.03m \), and over \( 60m = \pm 0.10m \). *Velodyne VLS-128 initial release (63-9480 Rev-3) and the latest Velodyne VLS-128 Alpha Prime (63-9679 Rev-1).
with Eq. 2. The reference map \( M \) corresponds to the complete drive starting at location marked with 2. Trajectory colored in blue with starting location marked with 9 corresponds to route A, yellow trajectory with start location marked with 9 is route B, and trajectory colored in green with starting location marked by 9 is route C. (b) is the ground truth points and complexity for matching.

In a similar way, the voxel grid filter is a down-sampling method applied on the input scan. A 3D grid is created on the input cloud, the local cloud in a voxel is replaced by the centroid (for the fields of \( x, y, z \) and intensity). Voxel grid filter preserves the general coverage (in distance) of the input cloud and reduces noise, but does not preserve the ring structure: the centroid may correspond to a point in between two rings unrelated to the original scan. By reducing the input cloud complexity according to the voxel grid cell size, the number of iterations to achieve convergence may be reduced.

3) VoxelGrid Filter: Similar to the map resolution, the voxel grid filter is a down-sampling method applied on the input scan. A 3D grid is created on the input cloud, the local cloud in a voxel is replaced by the centroid (for the fields of \( x, y, z \) and intensity). Voxel grid filter preserves the general coverage (in distance) of the input cloud and reduces noise, but does not preserve the ring structure: the centroid may correspond to a point in between two rings unrelated to the original scan. By reducing the input cloud complexity according to the voxel grid cell size, the number of iterations to achieve convergence may be reduced.

4) Number of LiDAR Beams: As show in Table I, there are multiple 3D LiDARs available for vehicle navigation, and one of their distinctive attributes is the number of beams (lasers or channels) and their distribution. A high number of beams and a fine horizontal resolution mean the size of the input pointcloud is large and thus more time is necessary of the derivation of these values are given in [16], the work by Magnusson [18] includes a very derivation. Magnusson et al. [17] and Sobreira et al. [22] present detailed comparisons of performance of NDT versus the Iterative Closest Point (ICP) algorithm and others.

B. NDT Evaluation Metrics

1) Iteration: Iteration corresponds to the number of cycles (and therefore processing time) of the Newton’s iterative method until achieving matching convergence. If the initial guess for the transformation parameters \( t \) is close enough to the actual transformation, then the number of iterations is small.

2) Fitness Score: Fitness Score is the degree of correspondence between two scans, obtained by the average sum of distances between closest points. When this score is small then correspondence between pointclouds is high. However, if the reference cloud lacks areas which are part of the input cloud, the distance between closest neighbors increases and so this score.

3) Transformation Probability: Transformation Probability, although strictly speaking not a probability, is the score of one point, obtained by dividing the fitness score by the number of points \( N \) of the input scan.

C. NDT Precision and Performance Factors

1) Input Cloud Down-sampling: The input cloud of conventional LiDARs may consist on tens to hundreds of thousands points. As such, the time to achieve convergence will increase with the size of the input cloud and may limit real-time response. Down-sampling of the input cloud reduces points and complexity for matching.

2) Resolution of Reference Map: In a similar way, the complexity of the map affects NDT matching performance. Resolution corresponds to the size of the ND voxels, a large resolution improves processing time but reduces features and so matching becomes unstable. On the other hand, very small resolution means higher processing time and causes incorrect associations between input cloud and very close ND voxels, therefore accuracy is affected.

3) VoxelGrid Filter: Similar to the map resolution, the voxel grid filter is a down-sampling method applied on the input scan. A 3D grid is created on the input cloud, the local cloud in a voxel is replaced by the centroid (for the fields of \( x, y, z \) and intensity). Voxel grid filter preserves the general coverage (in distance) of the input cloud and reduces noise, but does not preserve the ring structure: the centroid may correspond to a point in between two rings unrelated to the original scan. By reducing the input cloud complexity according to the voxel grid cell size, the number of iterations to achieve convergence may be reduced.

4) Number of LiDAR Beams: As show in Table I, there are multiple 3D LiDARs available for vehicle navigation, and one of their distinctive attributes is the number of beams (lasers or channels) and their distribution. A high number of beams and a fine horizontal resolution mean the size of the input pointcloud is large and thus more time is necessary of the derivation of these values are given in [16], the work by Magnusson [18] includes a very derivation. Magnusson et al. [17] and Sobreira et al. [22] present detailed comparisons of performance of NDT versus the Iterative Closest Point (ICP) algorithm and others.

B. NDT Evaluation Metrics

1) Iteration: Iteration corresponds to the number of cycles (and therefore processing time) of the Newton’s iterative method until achieving matching convergence. If the initial guess for the transformation parameters \( t \) is close enough to the actual transformation, then the number of iterations is small.

2) Fitness Score: Fitness Score is the degree of correspondence between two scans, obtained by the average sum of distances between closest points. When this score is small then correspondence between pointclouds is high. However, if the reference cloud lacks areas which are part of the input cloud, the distance between closest neighbors increases and so this score.

3) Transformation Probability: Transformation Probability, although strictly speaking not a probability, is the score of one point, obtained by dividing the fitness score by the number of points \( N \) of the input scan.

C. NDT Precision and Performance Factors

1) Input Cloud Down-sampling: The input cloud of conventional LiDARs may consist on tens to hundreds of thousands points. As such, the time to achieve convergence will increase with the size of the input cloud and may limit real-time response. Down-sampling of the input cloud reduces points and complexity for matching.

2) Resolution of Reference Map: In a similar way, the complexity of the map affects NDT matching performance. Resolution corresponds to the size of the ND voxels, a large resolution improves processing time but reduces features and so matching becomes unstable. On the other hand, very small resolution means higher processing time and causes incorrect associations between input cloud and very close ND voxels, therefore accuracy is affected.

3) VoxelGrid Filter: Similar to the map resolution, the voxel grid filter is a down-sampling method applied on the input scan. A 3D grid is created on the input cloud, the local cloud in a voxel is replaced by the centroid (for the fields of \( x, y, z \) and intensity). Voxel grid filter preserves the general coverage (in distance) of the input cloud and reduces noise, but does not preserve the ring structure: the centroid may correspond to a point in between two rings unrelated to the original scan. By reducing the input cloud complexity according to the voxel grid cell size, the number of iterations to achieve convergence may be reduced.

4) Number of LiDAR Beams: As show in Table I, there are multiple 3D LiDARs available for vehicle navigation, and one of their distinctive attributes is the number of beams (lasers or channels) and their distribution. A high number of beams and a fine horizontal resolution mean the size of the input pointcloud is large and thus more time is necessary of the derivation of these values are given in [16], the work by Magnusson [18] includes a very derivation. Magnusson et al. [17] and Sobreira et al. [22] present detailed comparisons of performance of NDT versus the Iterative Closest Point (ICP) algorithm and others.
for processing. Finer vertical angular resolutions mean better estimation of the voxel grid vertical centroid.

5) Matching Initialization: It is important to define the initial position and pose before a matching cycle with new scan data. As the vehicle moves continuously over time, the position computed in the previous cycle is used as a guess for the new cycle. In addition, the use of dead reckoning such as speed, acceleration, odometry and other navigation systems such as IMU and GNSS, can help improve the estimation of the initial position before the next cycle.

III. MULTIPLE LiDAR DYNAMIC TRAFFIC DATA

A. Data Collection

The target was to collect data in a variety of traffic conditions, including different type of environments, varying density of traffic and times of the day. We drove our instrumented vehicle three times per day and collected data for the following key time periods:

- Morning (9am-10am)
  - Pedestrian traffic: medium-low
  - Vehicle traffic: high
  - Conditions: people commuting to work, university students and staff arriving on the campus. Clear to overcast weather.

- Noon (12pm-1pm)
  - Pedestrian traffic: high
  - Vehicle traffic: medium-low
  - Conditions: large number of university students and staff heading to and from the various cafeterias and restaurants around Campus. These are largely accessible by walking, so vehicle traffic is relatively low around this time. Clear to overcast weather.

- Afternoon (2pm-4pm)
  - Pedestrian traffic: low
  - Vehicle traffic: medium-low
  - Conditions: foot traffic around campus is low during this busy work and class period, and vehicle density is also relatively normal or low. Clear to overcast weather.

Fig. 1 shows the vehicle used for data capture. The 3D LiDAR on top was replaced for each experiment only after the three time periods were recorded, only one LiDAR was used at a time to avoid noise due to mutual interference. Data from other sensors (RGB camera, IR camera, 360° camera, event camera, IMU, GNSS, CAN) was also recorded together with LiDAR data, together with timestamps, using ROS [23]. In addition, we collected calibration data for each new LiDAR setup to perform extrinsic LiDAR to camera calibration, using a checkerboard and various other points of interest. Clear lighting conditions were ensured to record such data. Vector map files (HD map) for public road outside of the Nagoya University campus, are also provided.

B. Evaluation Routes

From the main trajectory shown in Fig. 2 we selected 3 different routes for this study:

- Route A (blue): about 749 m of private roads inside Nagoya University, narrow, surrounded by trees and buildings, gentle slopes; maximum velocity allowed inside the campus is 30 km/h.
- Route B (yellow): about 475 m of public urban roads surrounding Nagoya University campus, wide, multiple lanes, large intersections, traffic signs and marks, mostly flat; maximum velocities between 40 km/h to 50 km/h.
- Route C (green): about 797 m of public urban roads surrounding Nagoya University campus, two lanes, intersections, hilly (7.6% average slope); maximum velocity 40 km/h.

For each route, we used the previously recorded sensor LiDAR log data included as part of our LIBRE dataset. To keep relative consistency in driving conditions, we chose the morning time rides for this study. In total, thirty (ten LiDARs times 3 routes) different evaluations for 3D map creation and for vehicle localization.

Fig. 3 shows the elevation errors due error accumulation during map creation.

IV. MAPPING EVALUATION

For each LiDAR and for each route, we created a 3D map using NDT for mapping implementation in Autoware. The NDT resolution was defined as 1 m, the maximum number of iterations to 50, the minimum range was 3 m and maximum range 200 m. To add new scans, especially during congested
traffic, the minimum shift (translation) parameter was set to 1 m. After mapping the maps were not down-sampled to keep their integrity for this analysis, thus some of the maps are too large to be used for practical self-driving. The specs of the computer used in this evaluation are: Intel Xeon CPU E3-1545M with 4 cores/8 threads, NVIDIA Quadro M1000M GPU with 4GB GPU memory and 512 GPU cores, 32GB main RAM and a external 2TB SSD storage for log data recording.

Fig. 3 shows the elevation versus traveled distance for each LiDAR and each route while creating maps (ground truth is shown in black). Since each ride is slightly different (ex., changing lanes to avoid obstacles) the differences in trajectories in the XY plane are not considered here, instead the most important problem of drifting in elevation due to accumulated errors. While LiDARs with higher number of beams, finer vertical angular resolution, and higher range precision are closer to the ground truth, VLP-16 with lowest beam count and OS1-64 with lower range accuracy have the highest vertical errors. The reasons for this elevation error may be found on the NDT mapping algorithm itself, as reported previously in [24], and more research is necessary, however this work sheds light on the dependency of the LiDAR beams configuration. The environment may also play a role in such errors, routes B and C are regular roads while route A is a narrow road on the university campus.

We also consider as additional metrics to analyze map quality, the mean map entropy (MME) and the mean plane variance (MPV) discussed in Razlaw et al. [25]. The mean map entropy (MME) score $H(M)$ of map $M$ is given by Eq. (4) as:

$$h(x_k) = \frac{1}{2} \ln [2\pi e \Sigma(x_k)]$$

$$H(M) = \frac{1}{M} \sum_i h(x_i) \quad (4)$$

The covariance $\Sigma(x_i)$ of point $x_k$ is computed using a k-dtree neighborhood search with radius $r$. The mean plane variance (MPV) score $V(M)$ of the map is given by equation Eq. (5) as:

$$V(M) = \frac{1}{M} \sum_i v(x_i) \quad (5)$$

where $v(x_i)$ is the upper quartile of distances with the best fitting plane for the points around $x_k$ in the search radius $r$.

For the ground truth reference map provided by MMS, the MME score is 0.294893 and the MPV score is 0.669889. Table II summarizes the results during mapping. The driving time for the route, number of scans included, and so on. As expected, LiDARs with higher number of beams take more time (iterations), while the fitness score remains similar for all sensors. The MME and MPV scores seem to favour the legacy sensors HDL-64S2 and HDL-32E, possibly due to the quality of their calibration and measurement accuracy when compared with newer devices.

One element which affects the MME and MPV scores is the presence of objects in the map, in particular on the road.

Table II: Mapping results for each route and each LiDAR.

| Sensor          | Beams | N. points | Drive(s) | N. scans | Mean iter. | Std. iter. | Mean Fit.Sc. | Std. Fit.Sc. | MME     | MPV     |
|-----------------|-------|-----------|----------|----------|------------|------------|--------------|--------------|---------|---------|
| VLS128ap        | 128   | 160390220 | 204      | 1958     | 3.5805     | 3.8839     | 0.5175       | 0.0430       | 0.9073  | 0.5216  |
| VLS128          | 128   | 126705326 | 160      | 1586     | 6.1513     | 8.4786     | 0.5453       | 0.0335       | 0.5701  | 0.4045  |
| Pandar64        | 64    | 134901915 | 158      | 1571     | 6.1948     | 10.1055    | 0.5154       | 0.0315       | 0.5849  | 0.3851  |
| OSI-64          | 64    | 60912465  | 149      | 1486     | 4.3419     | 8.3900     | 0.5254       | 0.0304       | 0.5849  | 0.3851  |
| HDL-64S2        | 64    | 78402509  | 177      | 1764     | 5.9195     | 8.0350     | 0.5265       | 0.0481       | 0.2976  | 0.2900  |
| Pandar40P       | 40    | 81345147  | 153      | 1527     | 5.5200     | 8.7917     | 0.5262       | 0.0359       | 0.4923  | 0.3945  |
| VLP-32C         | 32    | 29679934  | 155      | 1528     | 3.1041     | 4.9624     | 0.5233       | 0.0379       | 0.4772  | 0.3867  |
| RS-lidar32      | 32    | 30053527  | 153      | 1505     | 4.0339     | 6.9112     | 0.5271       | 0.0404       | 0.5283  | 0.3944  |
| HDL-32E         | 32    | 35845169  | 155      | 1547     | 3.9774     | 6.9487     | 0.5256       | 0.0323       | 0.0299  | 0.2896  |
| VLP-16          | 16    | 13215302  | 147      | 1447     | 2.918       | 2.8895     | 0.5355       | 0.0406       | 0.4338  | 0.3823  |

Table II: Mapping results for each route and each LiDAR.
surface. Fig. 4 shows the reference MMS map colored by point entropy and point plane variance, both using a search radius $r$ of 1 m. The map portion covered by vector maps was curated and no motion artifacts are observed, however map data inside the Nagoya University campus clearly show trails from objects in motion during mapping and not filtered here. The traffic conditions were slightly different while driving with each sensor, sometimes there were vehicles present leaving such trails on the map, and thus MME and MPV scores rose. Using the vector maps to filter road points and then removing points inside the road with high entropy and plane variance is a possible filtering strategy to consider.

![Fig. 4: Reference MMS map coloured entropy (a) (green lower entropy, magenta is higher entropy) and by plane variance (b) (red lower variance, green higher variance) for each point.](image)

**V. LOCALIZATION EVALUATION**

We performed thirty experiments on localization, using the ground truth reference map, and NDT matching implementation in Autoware. NDT matching requires the input cloud to be down-sampled, we used the voxel grid filter with a voxel size of 2 m and the maximum distance of 200 m. Similar to NDT mapping, NDT resolution was set to 1 m, maximum number of iterations to 50, error threshold was 1 m. The initial position was manually defined for each case. Fig. 5 show qualitative results of the different LiDARs during localization at specific locations. These images show the achieved coverage by each LiDAR's original input cloud after being transformed. Examples of filtered input cloud by the voxel grid filter are shown in Fig. 7.

Fig. 6 shows the localization performance of the different LiDARs per each route. In general, VLP-16 had a very fast convergence while the LiDARs with higher number of beams struggled. The higher iterations at the beginning are due to the different initialization configurations required for each sensor and route. Route B shows the higher number of fluctuations in iterations, this is partially due to changes in the environment not included in the reference MMS map. Also, this route includes portions where maximum speed was 50 km/h, the higher speed and the constant 10 Hz scanning rate of the sensors points to the need of motion compensation in NDT.

**VI. CONCLUSIONS**

In this work, we presented a detailed comparison and characteristics of ten different 3D LiDARs, each one a different model from diverse manufacturers, for navigation tasks (3D mapping and 6-DOF localization), using as common reference the Normal Distributions Transform (NDT). We analyzed each LiDAR pointcloud data alone and without assistance of any other navigation and dead reckoning systems. Data in this study comes from dynamic traffic data in our LIBRE dataset.

While ultimate reason may lie in the NDT implementation, the elevation during mapping for all 3D LiDARs was incorrect in all cases, drifting in elevation due to accumulated errors in posture estimation, the LiDAR’s beams configuration and the surrounding environment seem to have an important effect. Correct localization was achieved for all LiDARs using a reference ground truth map. Even if all input clouds were equally filtered, the number of iterations varied largely especially for multiple beam configurations and also depending on the environment (existence of sufficient reference data).

Future directions derived from this work include, studying the elevation problem in NDT mapping, evaluation using other algorithms such as LeGo-LOAM\[26\], different down-sampling pre-processing filtering which preserves the point cloud structure, automatic tuning of localizer parameters. The application of MME based feature points for localization will be also considered.

**ACKNOWLEDGMENTS**

This work was supported by the Core Research for Evolutional Science and Technology (CREST) project of the Japan Science and Technology Agency (JST). We would like to extend our gratitude to the Autoware Foundation\[4\] to realize this project. Finally, and as a matter of course, this work would not have been possible without the invaluable support of Velodyne Lidar Inc., Ouster Inc., Hesai Photonics Technology Co., Ltd., and RoboSense–Suteng Innovation Technology Co., Ltd.

**REFERENCES**

[1] Velodyne LiDAR, Inc. Alpha Prime: Powering Safe Autonomy (63-9679 Rev-1). https://velodynelidar.com/wp-content/uploads/2019/12/63-9679_Rev-1_DATASHEET_ALPHA-PRIME_Web.pdf. [Accessed 20-January-2020].

[2] ——. Alpha Puck: Ultra-High Resolution, Long Range Lidar for Autonomous Systems (63-9480 Rev-3). https://autonomoustuff.com/wp-content/uploads/2019/05/AlphaPuck_Datasheet_whitelabel.pdf. [Accessed 20-January-2020].

[4] https://www.autoware.org
Route A Route B Route C

Fig. 5: Qualitative results of localization for each LiDAR for each LiDAR, grouped by route and colored by intensity (auto scaled), scenes of the chosen location are included. For each group, the LiDAR models are indicated by the given labels.

[3] ——. User’s Manual and Programming Guide – HDL64-S2 and S2.1 (63HDL64E 8 Nov-2016). https://www.argocorp.com/cam/special/Velodyne/common/pdf/63-HDL64ES2%20HDL-64E%20S2.1%20Manual%20Rev-F.pdf. [Accessed 20-January-2020].

[4] ——. HDL-32E: High Resolution Real-Time 3D Lidar Sensor (97-0038 Rev-N). https://velodynelidar.com/wp-content/uploads/2019/12/97-0038-Rev-N-97-0038-DATASHEETWEBHDL32E_web.pdf. [Accessed 20-January-2020].

[5] ——. Ultra Puck: High Resolution Real-Time Lidar for Autonomous Systems (63-9378 Rev-K). https://velodynelidar.com/wp-content/uploads/2019/12/63-9378-Rev-K-Ultra-Puck-Datasheet_web.pdf. [Accessed 20-January-2020].

[6] ——. Puck: Versatile Real-Time Lidar Sensor (63-9229 Rev-K). https://velodynelidar.com/wp-content/uploads/2019/12/63-9229-Rev-K-Puck-Datasheet_web.pdf. [Accessed 20-January-2020].

[7] Hesai Photonics Technology Co., Ltd. Pandar64: 40-Channel Mechanical LiDAR. https://web2019.azureedge.net/uploads/Pandar64_Brochure.pdf. [Accessed 20-January-2020].

[8] ——. Pandar40P: 40-Channel Mechanical LiDAR. https://web2019.azureedge.net/uploads/Pandar40P_Brochure.pdf. [Accessed 20-January-2020].

[9] Ouster, Inc. OS1 Gen1 Datasheet. http://data.ouster.io/downloads/OS1-Gen1-lidar-sensor-datasheet.pdf. [Accessed 20-January-2020].

[10] Robosense–Suteng Innovation Technology Co., Ltd. RS-LiDAR-32 Brochure. https://qncdn.sz-sites.com/RSLidar/32%20Brochure%2020190802.pdf. [Accessed 20-January-2020].

[11] S. Thrun, “Robotic mapping: A survey,” Exploring artificial intelligence in the new millennium, vol. 1, no. 1-35, p. 1, 2002.
Fig. 6: Number of iterations per LiDAR and per route.

Fig. 7: Voxel grid filter applied to the OS1-64 pointcloud with different maximum range filters.

Morales, N. Akai, E. Takeuchi, and K. Takeda, “End-to-end autonomous mobile robot navigation with model-based system support,” *Journal of Robotics and Mechatronics*, vol. 30, no. 4, pp. S63–S83, 2018.

[21] P. Biber and W. Straßer, “The normal distributions transform: A new approach to laser scan matching,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3. IEEE, 2003, pp. 2743–2748.

[22] H. Sobreira, C. M. Costa, I. Sousa, L. Rocha, J. Lima, P. Farias, P. Costa, and A. P. Moreira, “Map-matching algorithms for robot self-localization: a comparison between perfect match, iterative closest point and normal distributions transform,” *Journal of Intelligent & Robotic Systems*, vol. 93, no. 3-4, pp. 533–546, 2019.

[23] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, “ROS: an open-source robot operating system,” in *IEEE International Conference on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, Kobe, Japan, May 2009.

[24] A. Carballo, S. Seiya, J. Lambert, H. Darweesh, P. Narksri, L. Y.