Spatial Similarity-based Multiple LiDAR Synchronization and Data Falsification Detection for Reliable 3D Mapping

Arata Nagasaka [1*] and Masafumi Nakagawa[1]

[1]Shibaura Institute of Technology, 3-7-5, Toyosu, Koto-ku, Tokyo, 135-8548

*mh23025@shibaura-it.ac.jp

Abstract: *In recent years, there has been an increasing trend to use spatiotemporal data acquired by light detection and ranging (LiDAR) scanners for digital twins. Here, spatial coverage is an important element for 3D mapping to obtain highly accurate simulation results in digital twinning. However, urban spaces have many obstacles, and it is not easy to improve coverage by simply installing fixed scanners. Consequently, we focus on the integration of many LiDARs in self-driving cars and autonomous mobile robots to improve coverage. However, integrating time-series point clouds acquired by multiple LiDAR scanners poses challenges in terms of spatial and temporal synchronization because there are variations in measurement accuracy and time accuracy for each LiDAR scanner, and there are also issues with obtaining the scanner's location information. Furthermore, there are concerns about LiDAR spoofing and data cracking, so the reliability of the data needs to be guaranteed. Therefore, we propose a reliability guarantee methodology based on the spatial similarity between a highly reliable and a less reliable LiDAR scanner. We assumed an urban space where moving objects exist and conducted verification under various conditions. This study first verifies the proposed method using a mobile scanner and a fixed scanner, then tests time synchronization using scanners with different scanning patterns, finally examines the change in spatial consistency by inserting an AI-generated point cloud. Through the feasibility study of time synchronization of LiDARs, we confirmed that the accuracy of time synchronization is affected by feature value and the scanning pattern of the LiDAR scanner, leading to a decrease in accuracy. However, the feasibility of detecting LiDAR spoofing was also revealed. Future challenges include improving stability and processing speed in situations that lead to a decrease in accuracy. This study will contribute to the realization of secure 3D map creation and digital twins.*

Keywords: Digital twin, Time synchronization, LiDAR spoofing, Point clouds

Introduction

Light detection and ranging (LiDAR) scanners are sensors that can acquire time-series shape information as point clouds of real 3D space. LiDAR scanners can be used to sense urban spaces for digital twins. A digital twin is a concept for solving urban issues based on a cycle consisting of acquiring physical space information with sensors, recoding it in digital space, simulating city behavior, and feeding the simulation results back into the physical space. Classic digital twins were devised to improve processes in the manufacturing industry (Michael Grieves et al., 2017), but in recent years, there has been a growing movement to apply them to urban spaces. For example, the Tokyo Metropolitan Government in Japan is conducting a demonstration of sensing urban space to provide advanced administrative services in the future. In digital twins, high spatial coverage is important for obtaining highly accurate simulation results. As a scanning method, a demonstration experiment is being conducted in China (Junxuan Zhao et al., 2019) in which LiDAR is installed on the side of the road to measure traffic flow, and we consider that roadside LiDAR will become more widespread in the future. However, urban spaces have many obstacles, making it difficult to improve coverage by simply installing LiDAR scanners in fixed locations. Thus, we focus on the integration of multiple LiDARs in self-driving cars and autonomous mobile robots to improve coverage. The Tokyo Metropolitan Government project tested a participatory digital twin that uses scanners installed on smartphones and collects data from a variety of participants. However, there are differences in accuracy and difficulties in integrating data obtained from various sensors. In addition, time synchronization is not easy even under normal circumstances because the time accuracy of the sensors varies depending on the time source, from nanosecond-level global navigation satellite system (GNSS) to hundreds of millisecond-level network time protocol (NTP). Furthermore, there are currently concerns about LiDAR spoofing and data cracking, so the reliability of the data needs to be guaranteed. Research into attacks against LiDAR, known as LiDAR spoofing, is also being actively conducted. It has been shown that attacks that insert nonexistent objects or delete existing ones by irradiating a LiDAR scanner with a pulse are possible (Takumi Sato et al., 2024), which poses a threat to autonomous driving and digital twins. There are also point cloud generative AI, such as Point-E. And Point-E is based on a diffusion model and can output a point cloud in less than a few minutes by using a GPU when given a text prompt (Alex Nichol et al., 2022). Thus, any point cloud can be easily generated. Therefore, in this study, we propose a reliability guarantee methodology based on the spatial similarity between a highly reliable and a less reliable LiDAR scanner.

Literature Review

Point cloud registration can be categorized into local and global registrations. Local registration requires an initial position, while global registration does not. Local registration is widely used in robotics and civil engineering, but it has the problem of falling into a local optimum, and in recent years, research has been conducted to improve its robustness. To improve the robustness of local registration, there is a generalized iterative closest point (G-ICP) (Aleksandr V. Segal et al., 2010), which performs plane-to-plane matching, and an algorithm that performs registration by managing features of multiple planes similar to the plane-to-plane algorithm (Songlin Chen et al., 2019), which has improved robustness to noise. Global registration focuses on deriving a globally optimal solution. In global registration, there are registration algorithms such as Go-ICP (Jiaolong Yang et al., 2015) that can guarantee results independent of the initial position, and it is an approach to the ICP algorithm from the perspective of mathematical optimization. However, there are technical issues, such as the slow processing speed and registration accuracy, depending on the algorithm used. There are challenges when applying these registration algorithms to digital twins. Local registration requires highly accurate position information on the LiDAR. Meanwhile, while global registration guarantees a globally optimal solution for the evaluation function, it is necessary to verify its real-time performance and resistance to LiDAR spoofing. In addition, there is little discussion about the consistency of the time axis direction of stream point clouds as well as time synchronization and its guarantee between LiDAR scanners that do not belong to the same sensor network. Previous research has focused on detecting attacks on sensors in self-driving cars through spatiotemporal analysis (Cuiping Shao et al., 2022). However, this is confined to the car's system, and we believe that further development of methods is necessary to apply them to digital twins of cities.

Methodology

In this study, we consider time synchronization with spatiotemporal guarantees based on the degree of spatial coincidence between any LiDAR data (target point clouds) and highly reliable LiDAR data (reference point clouds). Here, we define the geometric consistency between LiDAR data, focusing on any frame, as spatial similarity (Figure 1). We also extend geometric consistency to the time direction, and the degree of spatial similarity between multiple frames of the reference point clouds and any frame of the target point clouds is defined as the time similarity (Figure 2). Then, we define both spatial similarity and time similarity together as the spatiotemporal similarity.

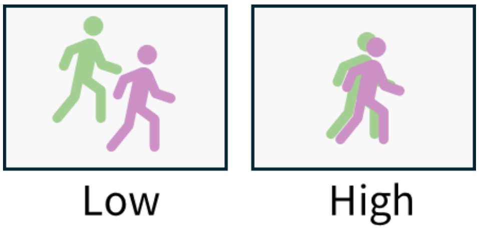

Figure 1: Spatial similarity.

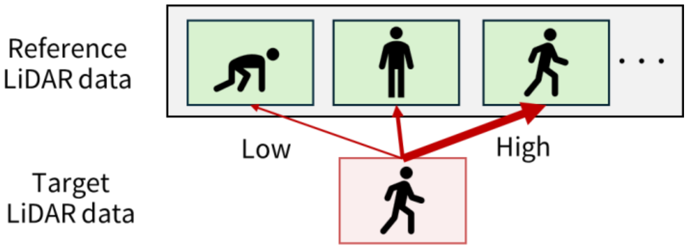

Figure 2: Time similarity.

The proposed methodology is shown in Figure 3. In this methodology, we focus on an urban space including many moving objects, fixed-point scanners installed as surveillance sensors, scanners mounted on self-driving cars, and autonomous mobile robots as moving scanners. During preprocessing, the two types of LiDAR scanners acquire time-series point clouds in the same area. Next, observation position identification is performed with the base map generation, based on the initial alignment, to estimate the relative positions of scanners. After data cleaning, the synchronization process is performed by deterministic spatiotemporal accuracy guarantee. The synchronization process requires that each sensor is synchronized with time accuracy equivalent to the sampling rate of the scanner. In other words, when LiDAR scanning is performed at a sampling rate of 10 Hz, which is a typical rate for mobility LiDAR scanners, accuracy at the NTP level (100 ms) is required at each scan. Moreover, data of at least one frame before and after are required, because the time synchronization process is based on relative temporal scanning matching. In addition, temporal scanning data without moving objects are required for base map generation.

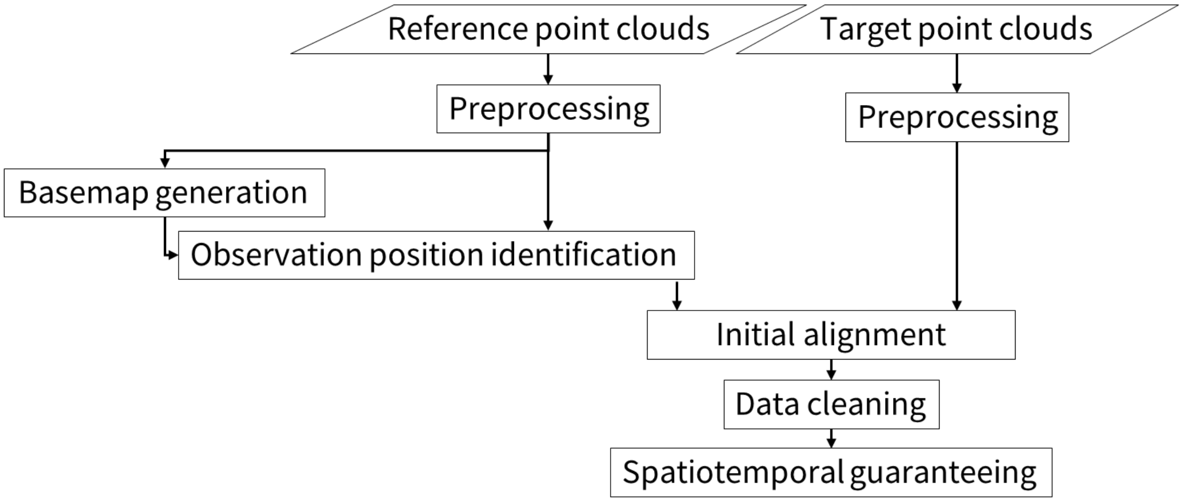

Figure 3: Proposed methodology.

a. Preprocessing

The point clouds acquired by the reference LiDAR scanner are converted from the sensor coordinate system to a global coordinate system (such as public coordinates) based on the position and orientation information measured in advance, and voxelized in global coordinate space. Similarly, the orientation of the target point clouds is transformed, and horizontal rectification is performed to set the elevation angle of the ground surface to 0, based on ground surface detection. However, we do not convert the positions to global coordinates here because we assume that the target LiDAR scanner does not have reliable position information, unlike the reference LiDAR scanner. Moreover, a range restriction is applied to remove spare point clouds as noises. These preprocessing steps generate a stable voxel space and improve the accuracy of subsequent processing.

b. Observation position identification

Next, the observation position identification process is performed to identify the target scanner from the reference scanner (Figure 4). First, motion detection is applied to detect differences using a base map without moving objects. Moving objects are deleted from the reference scanner point clouds before base map generation. A neighborhood search using a K-Dimensional (K-D) tree is used for the difference detection. When a point has no neighboring points, it is assumed to be a difference. A nearby search is a type of search algorithm, such as full search and space partitioning methodology. The K-D tree is a space partitioning method. Although the full search requires a large amount of calculation when large data are processed, the space partitioning method has the advantage of suppressing the expansion of the calculation amount. Therefore, the space partitioning method is generally used for neighborhood search in point cloud processing (Heng Yang, 2020). Here, the difference point clouds are assumed as moving objects. Then, point cloud clusters for each object are obtained by the different point cloud clustering based on Euclidean distance. The relative location of the target scanner is found from the point cloud clusters with position data obtained by a positioning device mounted on the target scanner.

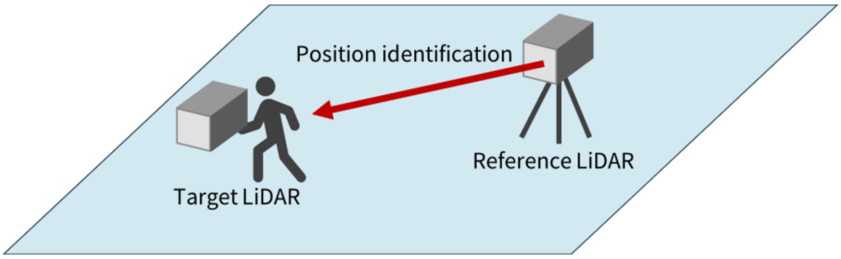

Figure 4: Position identification.

c. Data cleaning

In data cleaning, the ground and walls are estimated based on point cloud interpolation and plane estimation, and the processing range is narrowed down to the area surrounded by the estimated ground and walls. Data cleaning is required because differences in point density or acquisition range between the target and reference point clouds affect the spatiotemporal accuracy assurance process. Point cloud interpolation involves estimating the ground surface after voxelization and applying a process to fill in any gaps in the voxels.

d. Spatiotemporal accuracy guaranteeing

Position guarantee is achieved through point cloud registration. In point cloud registration, we apply G-ICP, which is more robust than the conventional ICP algorithm. Time guarantee is achieved by synchronizing the position-guaranteed point clouds with the time of the reference point clouds by assigning the time of the reference point clouds to the target point clouds that have a high degree of time agreement. Root mean square error (RMSE) is used for residual error evaluation.

Experiments

We verified our proposed methodology with three approaches. The first was a verification of the entire proposed methodology using fixed and moving scanners. The second was a verification of time synchronization of scanners with two different scanning patterns. The third was a verification of the change of spatial coincidence under a LiDAR attack. Table 1 shows the specifications of the equipment used in our experiments. Two types of LiDAR scanning patterns were used. The first is a horizontal linear scanning pattern. The second is a Lissajous curve scanning pattern. A Lissajous curve is obtained by combining two orthogonal simple harmonic motions. Compared with the horizontal linear scanner, the scanning with the Lissajous curve has the advantage of higher density in the vertical direction (Figure 5).

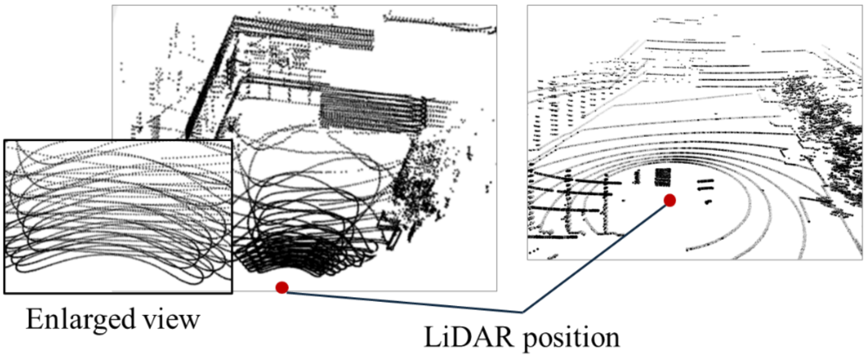

Figure 5: Examples of Lissajous scanning (left) and horizontal linear scanning (right).

Table 1: Sensor specifications.

Product Name (Manufacturer)	HORIZON (Livox)	AVIA (Livox)	VLP-32C (Velodyne)
Distance measurement accuracy [m]	2	2	3
Maximum distance measurement range [m]	260	320	360
FOV (H × V)	81.4 × 25.1	70.4 × 77.2	360 × 40
Point rate [points / s]	240,000	240,000	300,000
Frame rate [frames / s]	10	10	10

Product Name (Manufacture)	ZED-F9P (u-blox)
Positioning accuracy [cm]	1
Sampling rate [Hz]	1

Product Name (Manufacture)	MTi-G-719 (Xsens)
Angle measurement accuracy [°]	0.2-0.8
Sampling rate [Hz]	100

a. Verification of the proposed method using fixed and moving scanners

In our experiments, we aimed to verify the entire proposed methodology using a fixed LiDAR (reference) and a synchronized mobile LiDAR (target). A walking measurement experiment was conducted in a rooftop garden on a building at Toyosu Campus, Shibaura Institute of Technology (Tokyo, Japan). The experimental environment included obstacles and pedestrians. We selected two types of LiDAR scanners with Lissajous curve scanning patterns. The first LiDAR, the HORIZON (Livox), was used as the fixed scanner, and the second LiDAR, the AVIA (Livox), was used as the mobile scanner. We also selected an attitude and heading reference system (MTi-G-710) for attitude estimation. Position acquisition was performed using a GNSS antenna and receiver (ZED-F9P, u-blox). We acquired position data with real-time kinematic (RTK)-GNSS positioning using a private electronic reference station at the Etchujima Campus, Tokyo University of Marine Science and Technology (Tokyo, Japan). The sampling rate of each sensor was adjusted to 10 Hz. We aimed for 10 cm absolute accuracy based on practical accuracy for crash prevention among pedestrians and technical limitations such as sensor accuracy and voxel size. Moreover, the coordinates were converted to the Japan Plane Rectangular CS IX

(EPSG:6677) system, the global coordinate system used in our experiment. Figure 6 shows the mobile scanner system used in our experiment.

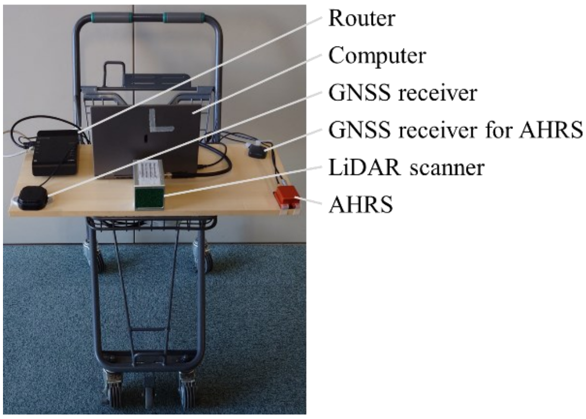

Figure 6: Mobile scanner system.

b. Verification of time synchronization between scanners with different scanning patterns

We focused on the time guarantee processing of the proposed methodology and aimed to verify whether synchronization between LiDARs is possible with different scanning patterns. The experiment was conducted at a large trampoline playground (Figure 7). We used two fixed scanners: a horizontal and linear scanning LiDAR scanner (VLP-32C, Velodyne) and a Lissajous curve scanning LiDAR scanner (AVIA, Livox). In this experiment, GNSS positioning was not performed to determine the true positions. Therefore, point cloud registration was performed with G-ICP, and the quality of the point cloud registration and synchronization was checked visually, because the coordinate conversion and time synchronization with GNSS were not performed.

Figure 7: Examples of trampoline-type play equipment.

c. Verification of spatial coincidence change during LiDAR attack

Here, we aimed to verify the changes that occur in spatial consistency using point clouds, assuming a malicious attack had been carried out during the spatiotemporal accuracy guarantee process. We inserted point clouds generated using the prompt “A tree” with Point-E into target point clouds obtained by a LiDAR scanner after the scale adjustment (Figure 8). The number of points in the inserted point cloud was 4096. Although the target point clouds were time-series data, the position where the point clouds were inserted was not changed among scanning frames.

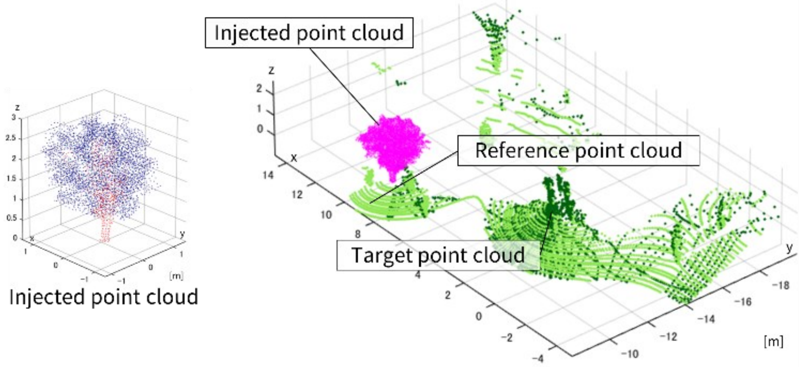

Figure 8: Inserted point clouds.

Result and Discussion

a. Verification of the proposed method using fixed and moving scanners

Figure 9 shows that passersby and mobile scanner systems were detected by differential detection. By contrast, point clouds were not recorded in the base map and scanning noises also existed as differences.

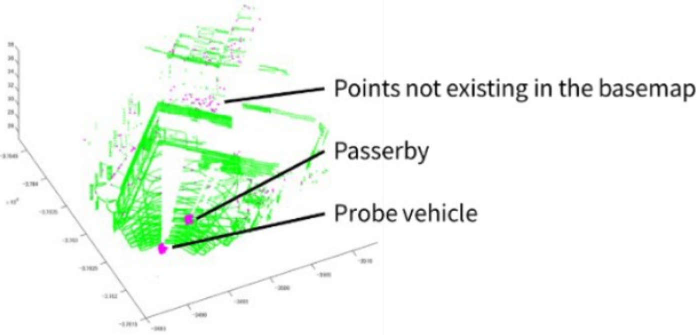

Figure 9: Differential detection result.

We discuss the results of motion detection based on background subtraction with a base map. The base map was generated by superimposing data recorded at times when there were no moving objects. However, urban areas include high-traffic areas where no scenes without moving objects exist. Nonetheless, there has been active discussion about motion detection in the field of robotics. One of the representative motion detection methods is a fast and robust method using occupancy grid maps, even when temporary occlusions exist (Naoki Suganuma et al., 2010). An occupancy grid is a 2D binary map projected from 3D point clouds to represent occupied or unoccupied areas in each grid. In this paper, we improved the occupancy grid-based method with elevation data as elevation-based occupancy grid mapping. Figure 10 shows the result of the elevation-based occupancy grid map, where 0 (dark blue) indicates no occupancy and estimated ground surfaces. Other values indicate features with elevation data.

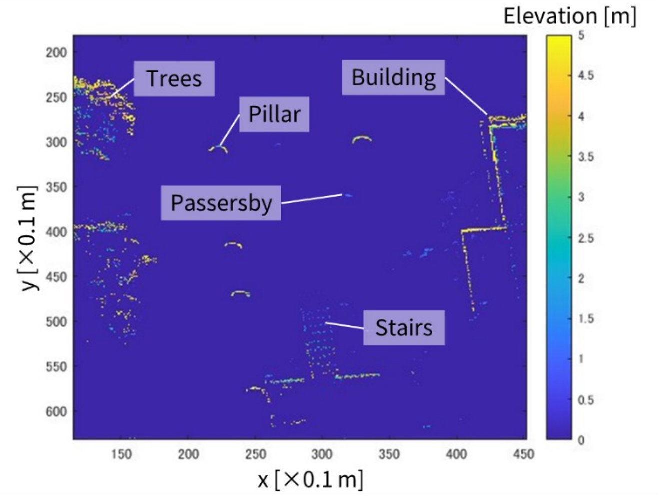

Figure 10: Elevation-based occupancy grid map.

We verified motion detection using the occupancy grid map. Although the evaluation is qualitative, we confirmed that the occupancy grid map method can detect objects (Figure 11).

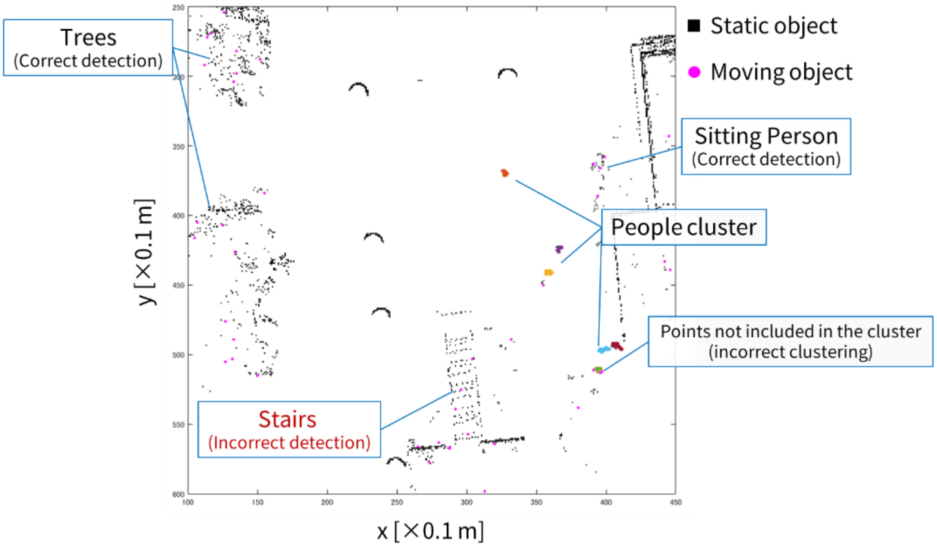

Figure 11: Motion and people cluster detection using occupancy grid maps.

We evaluated occupancy and non-occupancy by setting a time threshold for temporal data processing without conventional point cloud noise filtering. We confirmed that people can be detected by point cloud clustering of moving objects. In Figure 11, trees and sitting people were detected as moving objects. Moreover, stairs were falsely detected as moving objects. These failed estimations, such as alternating between occupied and unoccupied, occurred because non-repetitive LiDAR scanners have low time resolution at distant points. However, the failures can be reduced by a threshold adjustment of occupied and unoccupied areas. Moreover, noises can be rejected from point clouds during the clustering processing or using Kalman filters. We will apply noise reduction of point clouds to our proposed methodology in future studies.

Table 2: Processing time.

Processing order	Processing item		Processing time [ms]
1 (Asynchronous)	Preprocessing	Reference scanner	12.4
		Target scanner	1.1
2	Observation position identification		95.8
3	Data cleaning		415.7
4	Position guarantee		82.6
5	Time guarantee		5.6
Total			612.1

Table 2 shows that considerable time was spent on data cleaning. As a result, the total processing time was 612.1 ms for data acquisition at a frame rate of one frame per 100 ms. We confirmed that real-time processing is difficult when processing all frames in the current status. Next, Table 3 shows the residual error evaluation. The most probable values were data obtained by correcting the measured values because an offset was approximately 29 cm between the GNSS receiver and LiDAR scanner on the mobile scanner system. The estimated value in (1) of Table 3 represents the center of gravity of the mobile scanner system. The estimated value in (2) of Table 3 represents the position of the center of gravity of the probe vehicle after registration in the position guarantee process. The results suggest that large-scale point clouds were registered with submeter accuracy using a moving scanner.

Table 3: Residual error.

	Residual error [cm]		
	Min.	Ave.	Max.
(1) The identified center of gravity of the mobile scanner system	75	80	88
(2) The center of gravity of the mobile scanner system after the registration	8	11	17

In terms of performance, we aimed for an accuracy of approximately 10 cm, based on the required accuracy for collision prevention with pedestrians, the technical limitations of sensor accuracy, and memory size for voxelization. Figure 12 shows a difference in point density and partial loss of the object due to the difference in scanning positions. In other words, appropriate corresponding points could not be detected, affecting the accuracy of time synchronization. In the future, it is expected that the accuracy will improve using model-based matching.

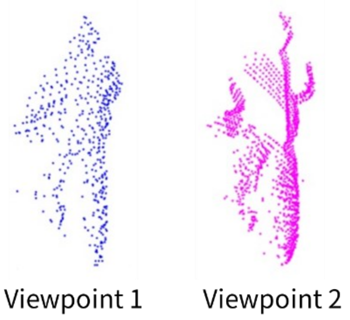

Figure 12: Changes due to different scan positions.

Previous papers have shown that registration accuracy deteriorates when combining occluded and non-occluded point clouds (Fukutomi et al., 2019). Therefore, we focus on occlusion detection before point cloud registration to improve accuracy. By contrast, we also consider a different approach such as a time synchronization method without point cloud registration, as our feature work. Figure 13 shows that the accuracy of time synchronization changes depending on the temporal and geometrical feature values in the point clouds. In this paper, we describe the temporal feature values as the number of feature points and changing objects in the point clouds. It is important to note from Figure 13 that time synchronization was achieved in scenes with few features. Therefore, additional information is required with geometrical feature values extracted from the point clouds.

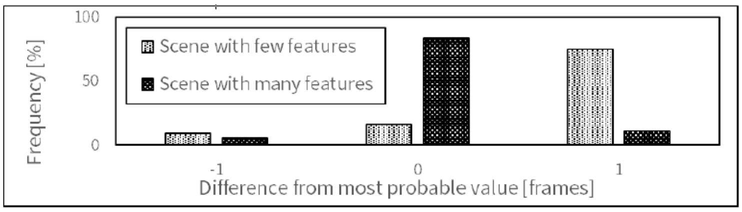

Figure 13: Time synchronization result.

b. Verification of time synchronization between scanners with different scanning patterns

In time synchronization between scanners with different scanning patterns, the rate of correct synchronization was approximately 40%. The low correct rate was due to the different scanning patterns and point densities. As shown in Figure 14, the point clouds of a pedestrian were not captured by the LiDAR scanner with a Lissajous curve.

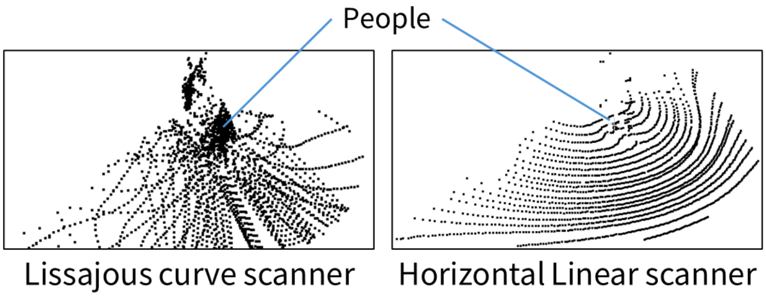

Figure 14: Changes in features due to different scanning patterns.

Time synchronization depended on global spatial coincidence because of the relative position changes of moving objects rather than on local spatial coincidence due to the posture of moving objects. Moreover, the proposed methodology became difficult when no significant change existed among the frames. Therefore, we verified whether accuracy could be improved by sampling several frames from the target frames. Figure 15 shows that temporal coarse frames improved the accuracy of time synchronization. Based on this situation, although the geometrical point cloud matches the posture of moving objects, time synchronization based on global spatial coincidence can be improved in an environment where multiple moving objects exist.

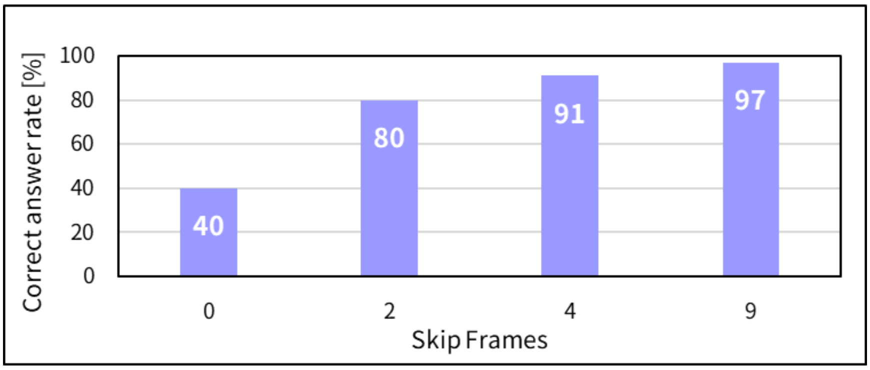

Figure 15: Relationship between the number of skipped frames and time synchronization accuracy.

c. Verification of spatial coincidence change during LiDAR attack

Figure 16 shows the change in residual error under attacks on a LiDAR scanner. When the residual error is large, spatial consistency is low. Because spatial consistency decreases under attacks, attacks on the LiDAR scanner can be detected by monitoring changes in spatial consistency.

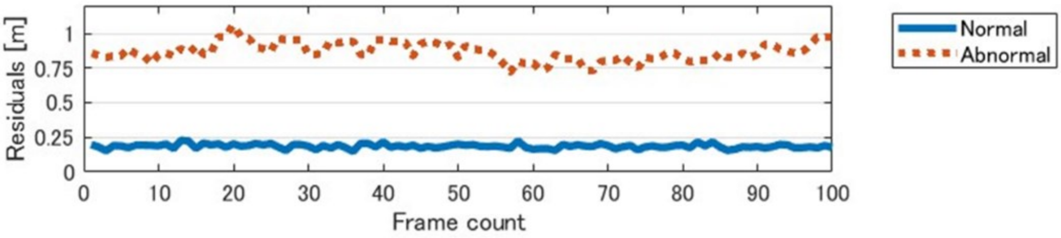

Figure 16: Changes in spatial consistency under LiDAR attacks.

However, we also observed that, when examining different scenarios, the difference between normal and abnormal becomes smaller (Figure 17). Therefore, more detailed scenarios are required to verify the relationship between attacks and spatial coincidence.

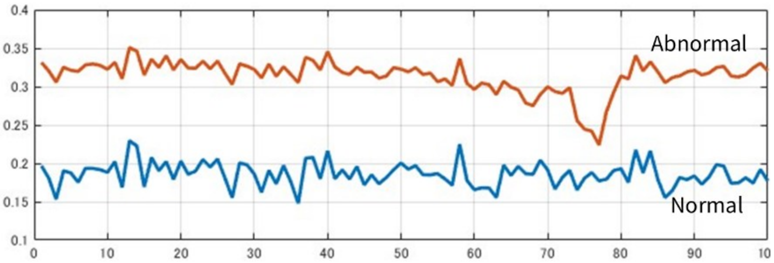

Figure 17: An example where the difference between normal and abnormal is small.

Conclusion

In this study, for urban digital twins, we proposed a method for ensuring spatiotemporal accuracy based on spatial coincidence between LiDARs. We also verified time synchronization and data tampering detection. Our results suggested that the proposed method is capable of time synchronization based on spatial coincidence. Specifically, we confirmed that the overall accuracy of the proposed method depends on the accuracy of point cloud registration. However, the proposed methodology has technical issues in environments with few features. Although the accuracy rate of the proposed method was low between LiDARs with different scanning patterns, we confirmed that the accuracy can be improved by parameter adjustment. Our prospects include the development of a method that does not depend on the accuracy of point cloud registration and verification using actual robots.

References

C. Shao, Z. Miao, B. Chen, Y. Cui, H. Li and H. Shu, (2022). An Attack Detection Method Based on Spatiotemporal Correlation for Autonomous Vehicles Sensors, 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), pp. 2187-2193.

Nichol, A., Jun, H., Dhariwal, P., Mishkin, P., Chen, M. (2022). Point-e: A system for generating 3d point clouds from complex prompts. arXiv preprint arXiv:2212.08751.

Sato, T., Hayakawa, Y., Suzuki, R., Shiiki, Y., Yoshioka, K., Chen, Q. A., (2023). LiDAR Spoofing Meets the New-Gen: Capability Improvements, Broken Assumptions, and New Attack Strategies. arXiv preprint arXiv:2303.10555.

Segal, A., Haehnel, D., Thrun, S., (2009). Generalized-icp. In Robotics: science and systems, Vol. 2.

Suganuma, N., Matsui, T., (2010). Robust environment perception based on occupancy grid maps for autonomous vehicle. IEEE, In Proceedings of SICE Annual Conference 2010, pp. 2354-2357.

Yang, H., Shi, J., & Carlone, L., (2020). Teaser: Fast and certifiable point cloud registration. IEEE Transactions on Robotics, 37(2), 314-333., No. 4, p. 435.

Yang, J., Li, H., Campbell, D., Jia, Y., (2015). Go-ICP: A globally optimal solution to 3D ICP point-set registration. IEEE transactions on pattern analysis and machine intelligence, 38(11), 2241-2254.

Zhao, J., Xu, H., Liu, H., Wu, J., Zheng, Y., Wu, D., (2019). Detection and tracking of pedestrians and vehicles using roadside LiDAR sensors. Transportation research part C: emerging technologies, 100, 68-87.