Adaptive Sensor Fusion of LiDAR and Stereo Camera for Robust Autonomous Navigation in Outdoor Environments

Kenta I.*[1], Akito A.[1], Arata N.[1],

Kazuyuki H.[2], Shotaro K.[2], Masafumi N.[1]

[1]Department of civil engineering, Shibaura Institute of Technology

3-7-5, Toyosu, Koto-ku, Tokyo, 135-8548, Japan

Email: ah21014@shibaura-it.ac.jp, mnaka@shibaura-it.ac.jp

[2]Watanabe Engineering Co., Ltd.

7-4-69, Noda-tyo, Fukushima City, Fukushima, 960-8055, Japan

Email: mim@watanabe-office.jp

Abstract: *Autonomous vehicles are widely promoted as a solution to reduce traffic accidents and improve logistics efficiency. However, many technical challenges remain before they can be put into practical use, such as improving the accuracy of self-position estimation and object recognition. Sensors installed in autonomous vehicles must be cost-effective while delivering high accuracy and real-time performance. Although previous research has achieved high-precision sensing, continuous output increases data processing demands data and power consumption. High-accuracy recognition using light detection and ranging (LiDAR) and stereo cameras has been reported, but most approaches require significant computational resources, such as GPU processing, which hinders real-time performance. On the other hand, integrating data from different types of sensors is considered effective for 3D measurement and navigation under changing weather conditions. However, the performance largely depends on the integration method. This study proposes a method for dynamically optimizing the output of LiDAR and stereo cameras based on environmental conditions to improve measurement performance in both sunny and rainy weather. In addition, the proposed method was applied to LIO-SAM, which combines non-repetitive scanning LiDAR with visual simultaneous localization and mapping (Visual SLAM) using a stereo camera, and its effectiveness was evaluated. However, sufficient self-position estimation accuracy was not achieved. Frame integration, horizontal plane estimation, and mask processing using reflection intensity values were attempted, but improvements remained limited. Future work will focus on developing point cloud correction techniques specific to non-repetitive scanning LiDAR and refining frame-to-frame interpolation using velocity estimation.*

Keywords: sensor fusion, LiDAR, stereo camera, LIO-SAM, Visual SLAM, autonomous vehicle

Introduction

In recent years, Japanese society has faced social challenges such as frequent traffic accidents and a severe labor shortage in the logistics industry. Many of these traffic accidents are caused by human error, stemming from factors such as an increasing number of elderly drivers and distracted driving. These incidents often involve not only collisions between vehicles, as well as dangerous incidents involving pedestrians and cyclists. Meanwhile, the logistics industry suffers from a critical shortage of workers due to long working hours, an ageing workforce, and difficulty securing personnel. Widespread adoption of autonomous vehicles is anticipated as a solution to these challenges. Their practical implementation could significantly reduce traffic accidents by minimizing human error resulting from operational mistakes and fatigue. In the logistics sector, unmanned delivery and truck platooning are expected to improve working conditions and reduce operational burdens by cutting labor costs and enhancing transport efficiency. However, there are still many challenges to the practical implementation of autonomous vehicles. Achieving, high precision and real-time performance in self-localization using sensors, as well as object recognition of surrounding vehicles and pedestrians is paramount. While previous research has achieved high-precision recognition, operating numerous sensors at a fixed output level constantly poses challenges in terms of increased maintenance and operational costs. For instance, although high-precision recognition methods using light detection and ranging (LiDAR) and stereo cameras have been reported, many of these require large-scale parallel processing via a graphics processing unit (GPU) or a field programmable gate array (FPGA). This often necessitates a high-cost hardware processing environment. Furthermore, sensor fusion technology, which integrates data from multiple heterogeneous sensors, is crucial for maintaining stable sensing accuracy under various weather and environmental conditions. For example, integrating information from different sensor types, such as LiDAR, cameras, and inertial measurement units (IMUs), has demonstrated the potential to achieve high-precision, robust object recognition and self-localization in unmanned autonomous vehicles (UAVs) (Harun, 2022). However, the integration methodology significantly influences performance. Incorrect design can lead to processing redundancy and increased computational resource requirements. Furthermore, sensor fusion involves real-time and synchronization challenges that require, optimization of both hardware and algorithms. Evaluations of simultaneous localization and mapping (SLAM) utilizing multiple sensors, such as cameras, LiDAR, IMUs, and GMSS highlight the importance of robust self-localization and map construction in dynamic environments

(Cadena et al., 2016). However, operating multiple sensors continuously carries the risk of an increased processing load. Therefore, achieving both high accuracy at low cost through dynamic, environment-adapted control represents a major future challenge. This research therefore proposes a methodology to enhance the performance of measurements under both clear and rainy conditions by dynamically optimizing the output of LiDAR and stereo cameras according to environmental conditions. Furthermore, the effectiveness of this methodology will be verified through its application to LIO-SAM processing using non-repeat-scan LiDAR with a wide scanning range and Visual SLAM processing using stereo cameras.

Methodology

As shown in Figure 1, the proposed methodology comprises LiDAR SLAM, LIO-SAM integrating IMU, Visual SLAM, and GNSS positioning. It also optimizes the measurement and navigation modes using these components.

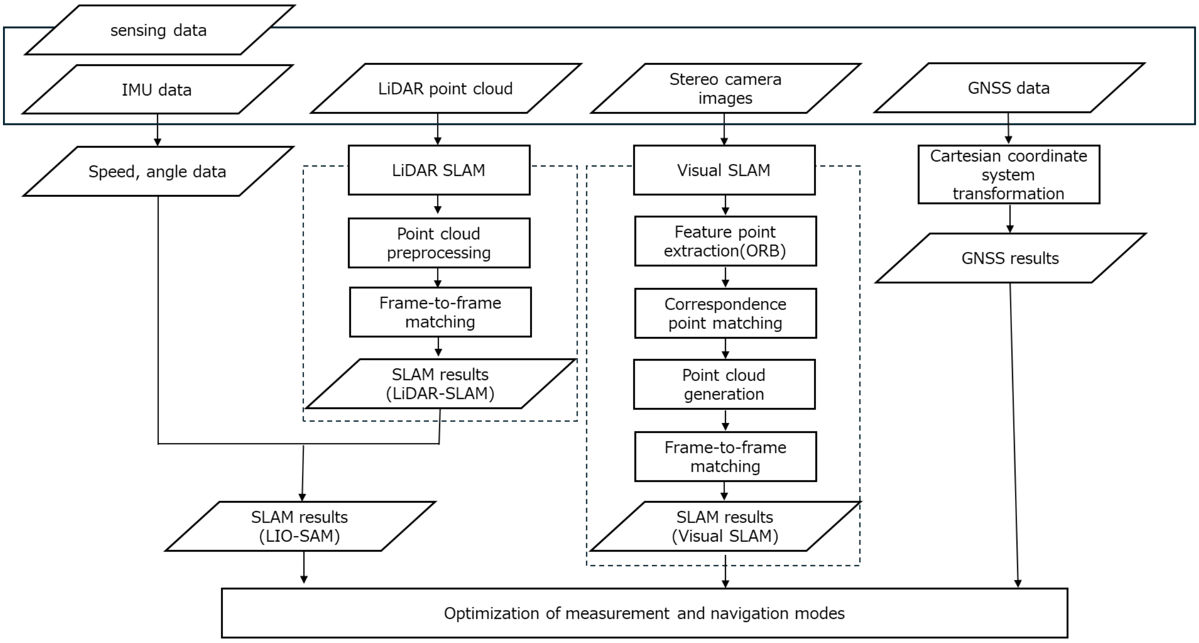
sensing data
IMU data
LiDAR point cloud
Stereo camera images
GNSS data
Speed, angle data
LiDAR SLAM
Point cloud preprocessing
Frame-to-frame matching
SLAM results (LiDAR-SLAM)
Visual SLAM
Feature point extraction(ORB)
Correspondence point matching
Point cloud generation
Frame-to-frame matching
SLAM results (Visual SLAM)
Cartesian coordinate system transformation
GNSS results
SLAM results (LIO-SAM)
Optimization of measurement and navigation modes

Figure 1: Proposed Methodology.

The LIO-SAM process comprises the following steps. First, it integrates non-repeating scan LiDAR point clouds into single point clouds. Next, it applies various preprocessing steps to the point clouds to enhance accuracy. Then, it performs iterative closest point (ICP) on the corrected inter-frame point clouds to calculate translation and rotation quantities. Finally, it uses IMU data for correction processing to achieve, SLAM processing through high-precision inter-frame matching.

a. Point Cloud Integration between Frames:

The Lissajous curve characteristic of the non-repeating scan LiDAR used in this study results in differences in the range of the observed point clouds between frames. This occurs because the timing and position at which the point clouds are collected vary between frames due to differences in the scanning methodology, it is difficult for point clouds from different frames to match perfectly. Thus, the first step is to integrate consecutive frames and process the point clouds to align them and form a coherent shape. Figure 2 illustrates point cloud integration between frames. The figure shows how the integration process mutually compensates for missing areas in individual frames to form more stable point clouds.

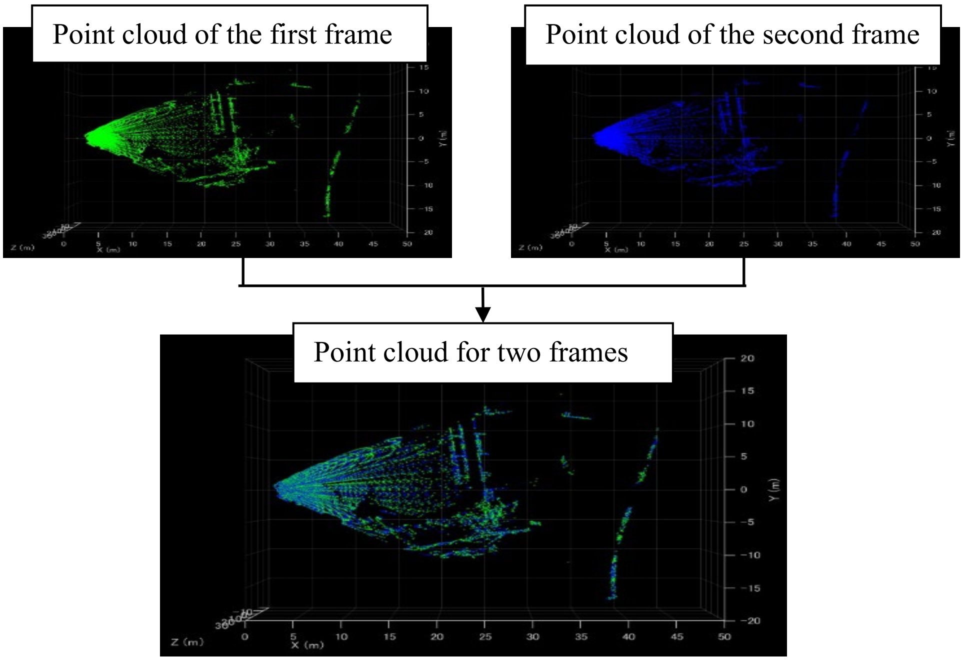
Point cloud of the first frame
Point cloud of the second frame
Point cloud for two frames
Z (m)
X (m)
Y (m)
0
5
10
15
20
25
30
35
40
45
50
20
15
10
5
0
−5
−10
−15
−20

Figure 2: Point Cloud Integration between Frames.

b. Point Cloud Preprocessing:

In this study, we first define a region of interest (ROI) or point cloud preprocessing to retain only the area necessary for analyzing the vehicle's surroundings. This eliminates unnecessary regions and reduces computational cost and misrecognition. Next, we use the reflected intensity information output by the LiDAR to identify object materials and road markings. We extract high-reflectivity objects using thresholding to improve mapping reliability. For point clouds with a large number of points, we perform downsampling is using a voxel grid filter. This replaces each voxel with a representative point, compressing the data volume while effectively removing unwanted points such as vegetation by prioritizing the retention of high-intensity points. Additionally, the horizontal plane is estimated using the random sample consensus (RANSAC). This process robustly extracts planar models, such as the road surfaces, from outliers. This process eliminates residual noise and provides stable base coordinates. Furthermore, the point clouds are clustered to remove small clusters as noise and emphasize those corresponding to large structures and road surfaces. Finally, significant feature points are extracted from the point clouds present in the vehicle's forward direction that have substantial inter-frame variation. Using these points as input for ICP achieves high-precision and robust registration. Figure 3 illustrates the overall point cloud workflow performed in this study. By removing ROIs, the area is narrowed down to only the vehicle's surroundings. Then, noise points are eliminated through clustering and filtering to extract valid feature points.

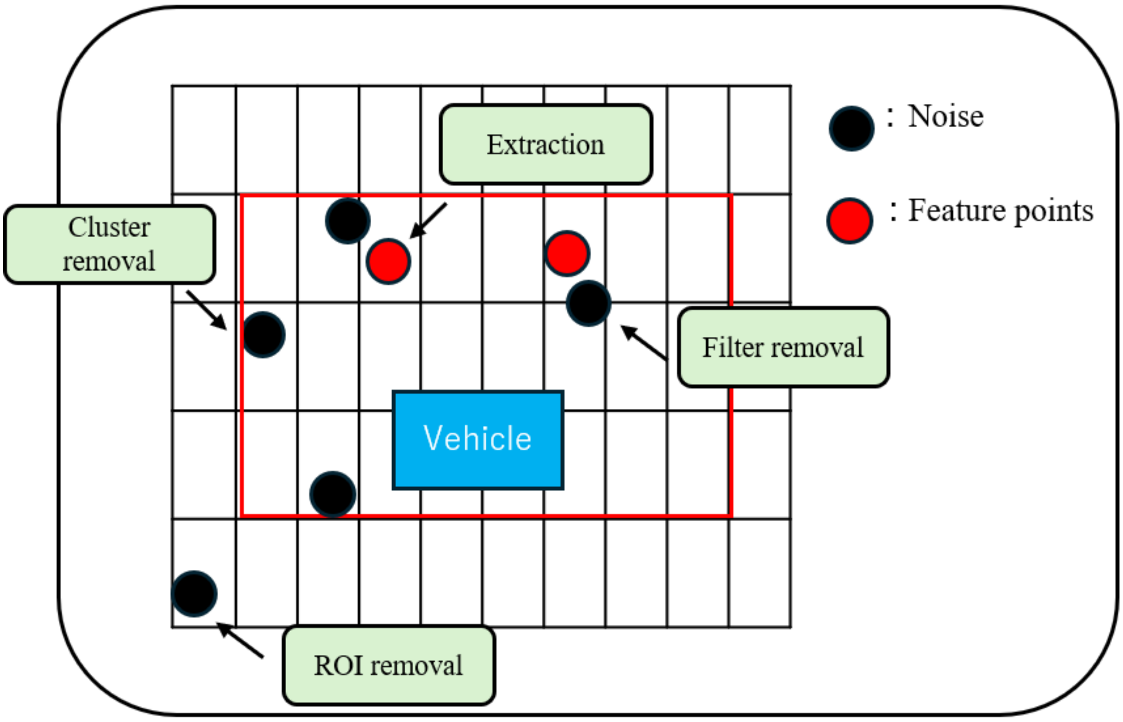

Figure 3: Point Cloud Processing.

c. LIO-SAM:

LiDAR inertial odometry SLAM (LIO-SAM) enhances the accuracy of self-localization and mapping by fusing IMU data through tightly coupled processing between LiDAR and IMU. Self-localization is achieved by matching point clouds across frames using the ICP algorithm on the corrected point clouds, followed by calculating the displacement of the self-position from the resulting difference. Furthermore, the angular and velocity data from the IMU are used to improve SLAM accuracy. This enables stable self-localization independent of sensor accuracy or movement speed.

d. ORB Feature Extraction:

Visual SLAM uses image matching processing using descriptor, such as scale-invariant feature transform (SIFT), features from accelerated segment test (FAST), binary robust independent elementary features (BRIEF). In Visual SLAM using stereo images, the oriented FAST and rotated BRIEF (ORB) algorithm (Rublee, 2011) is first applied to extract feature points from the images. ORB uses a simple and fast methodology for corner extraction, based on the FAST algorithm. This algorithm collectively determines the luminance values of the 16 pixels surrounding the central pixel in a circular pattern. This makes ORB well-suited for real-time processing. However, FAST cannot adequately account for changes in scale or image rotation. In practical applications, therefore, feature points are detected at each scale using the image pyramid structure. Furthermore, the principal direction is estimated from the bright gradient around each detected feature point. Based on this information, the BRIEF descriptor's sampling pattern is rotated, thereby imparting rotation and scale invariance. Note that SIFT also exists and similarly achieves rotation- and scale-invariant feature point extraction. However, SIFT is computationally intensive, which makes it unsuitable for real-time processing. It enables robust feature point extraction that is resistant to changes in viewpoint and invariant to rotation and scale. This makes it easier to find corresponding points between images taken from different viewpoints. Figure 4 visually demonstrates ORB's rotational and scale invariance by sho

ima

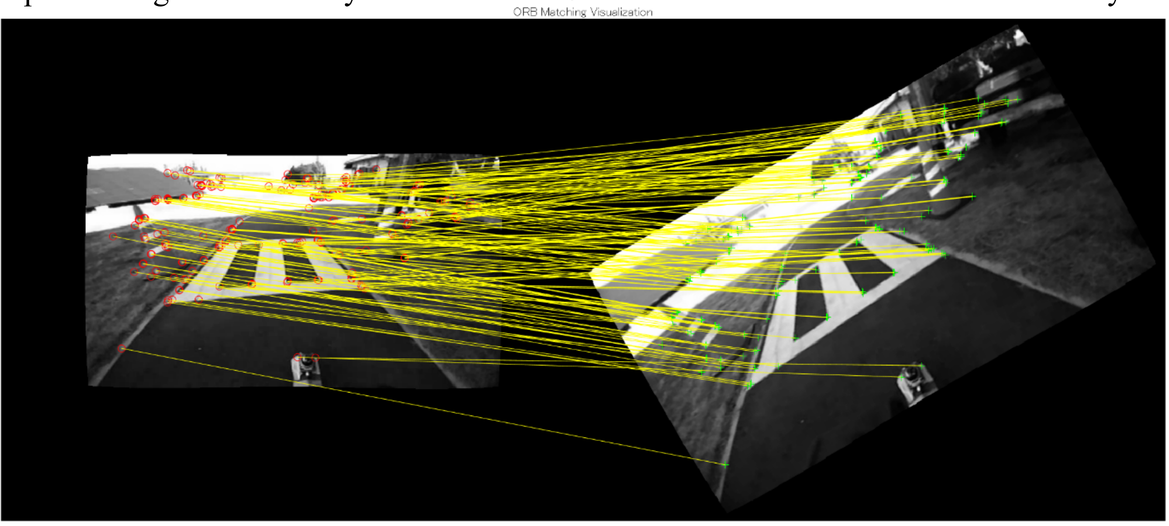

Figure 4: ORB Feature Point Matching Visualization.

e. SGM:

Furthermore, this study uses the semi-global matching (SGM) algorithm to assign precise depth information to ORB feature points. When estimating disparity from stereo images, the SGM algorithm first calculates the cost of each pixel's candidate disparity based on block matching. Then, these costs are aggregated along multiple paths, including the horizontal and vertical directions, to optimize the disparity for each pixel globally. Although full global optimization requires substantial computing power, SGM achieves significantly higher accuracy than block matching while reducing the processing load through its semi-global optimization. Moreover, SGM preserves the smoothness of the disparity field while minimizing errors near edges. This enables relatively stable disparity estimation even in areas with sparse textures or in high-contrast environments. Figure 5 shows an example of a disparity image generated by SGM. Figure 6 illustrates an example of a point cloud extracted from the disparity image. This confirms that the depth structure of the target object is reproduced in detail. Combining this with feature point extraction using ORB, this enables highly accurate and robust self-localization.

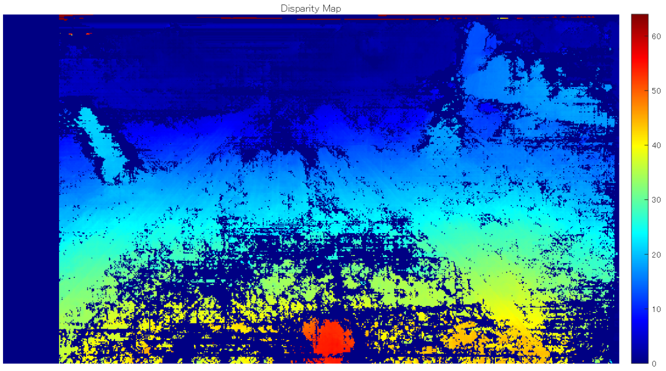

Figure 5: Disparity Image Generated by SGM.

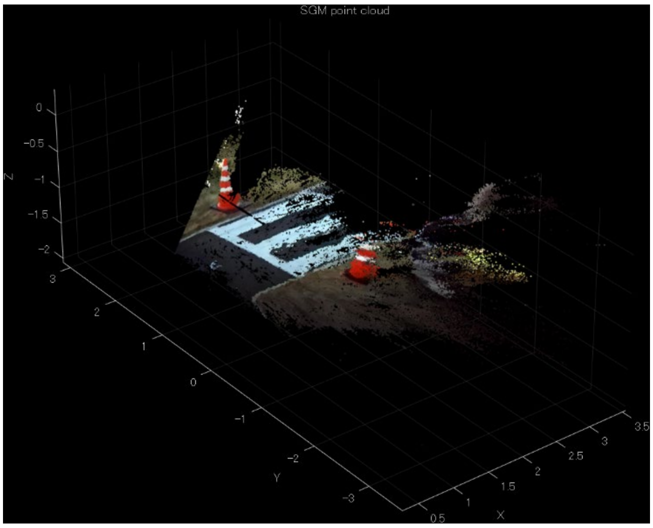

Figure 6: Point Cloud Extracted from the Disparity Image.

f. Visual SLAM:

In this study, we generate 3D points using correspondences of feature points extracted from left and right stereo images with ORB. Based on these 2D–3D correspondences, we estimate the camera pose using the PnP methodology. Furthermore, we achieve high-precision optimization while maintaining overall consistency by simultaneously handling the newly generated 3D points and camera pose simultaneously and minimizing reprojection error through bundle adjustment. This processing sequence enables the sequential estimation of self-position while simultaneously constructing a 3D map of the environment. Combined with high-precision depth estimation via ORB feature extraction and SGM, robust and precise Visual SLAM.is achieved.

g. Navigation Optimization:

We combine LIO-SAM and Visual SLAM using a tightly-coupled approach and integrate them as an optimization problem for navigation modes. This methodology incorporates multiple sensor observations are simultaneously incorporated within the same framework and corrects estimates for errors simultaneously. This approach enhances the synergistic effects between sensors more than a loosely coupled methodology, which combines the results of individual estimations at a later stage. In this study, the navigation mode optimization workflow first uses a factor graph to combine LiDAR-IMU constraints (derived from LIO-SAM) and camera observation constraints (derived from Visual SLAM) into the same graph structure. In a factor graph, the estimated self-attitude is represented by nodes, and sensor observations and prior information are represented by edges (factors). The optimal solution is derived by collectively minimizing the errors of these factors. Furthermore, applying weights adjusts the reliability of LiDAR and camera observations, respectively, thereby dynamically optimizing the outputs of LiDAR and stereo cameras according to environmental conditions. Note that GNSS positioning results, transformed from the WGS84 coordinate system to local coordinates are used as the ground truth. These processes aim to improve measurement performance in both clear and rainy weather conditions while reducing operational costs.

Experiment

A circuit track modeled on a real road (Figure 7) was selected as the test site. The experimental environment included asphalt and lawn surfaces, pedestrian crossings, lane markings, plants, and buildings. Pedestrians were also arranged as dynamic obstacles. To simulate rainy conditions, puddles were created. Then, a trolley equipped with sensors was then used to complete a lap of the circuit track. As shown in Figure 8, The trolley was equipped with a LiDAR (Avia, Livox), an omnidirectional LiDAR (Mid-360, Livox), a stereo camera (ZED2, Stereolabs), a monocular camera (AS300, SONY), and an AHRS (MTi-710G, Xsens). GNSS positioning employed standalone positioning.

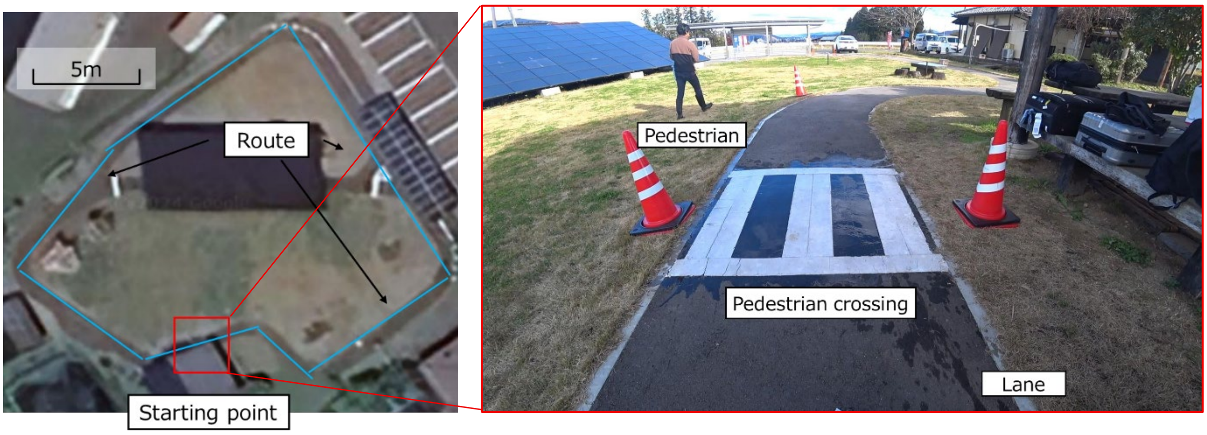

Figure 7: A Circuit Track Modelled on a Real Road.

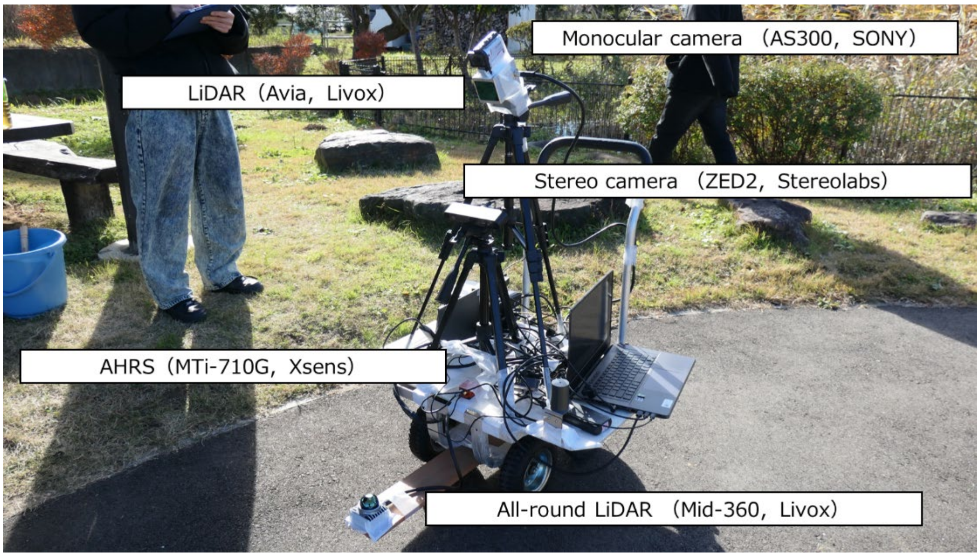
Monocular camera （AS300，SONY）
LiDAR（Avia，Livox）
Stereo camera （ZED2，Stereolabs）
AHRS（MTi-710G，Xsens）
All-round LiDAR （Mid-360，Livox）

Figure 8: 3D Measurement Device.

Results and Discussion

As shown in Figure 9, the GNSS positioning results confirmed a match between the actual driving performance and the GNSS-estimated trajectory. The smooth overall trajectory indicates favorable satellite reception conditions during positioning. However, slight discrepancies were observed between the starting and ending points, indicating the influence of cumulative error.

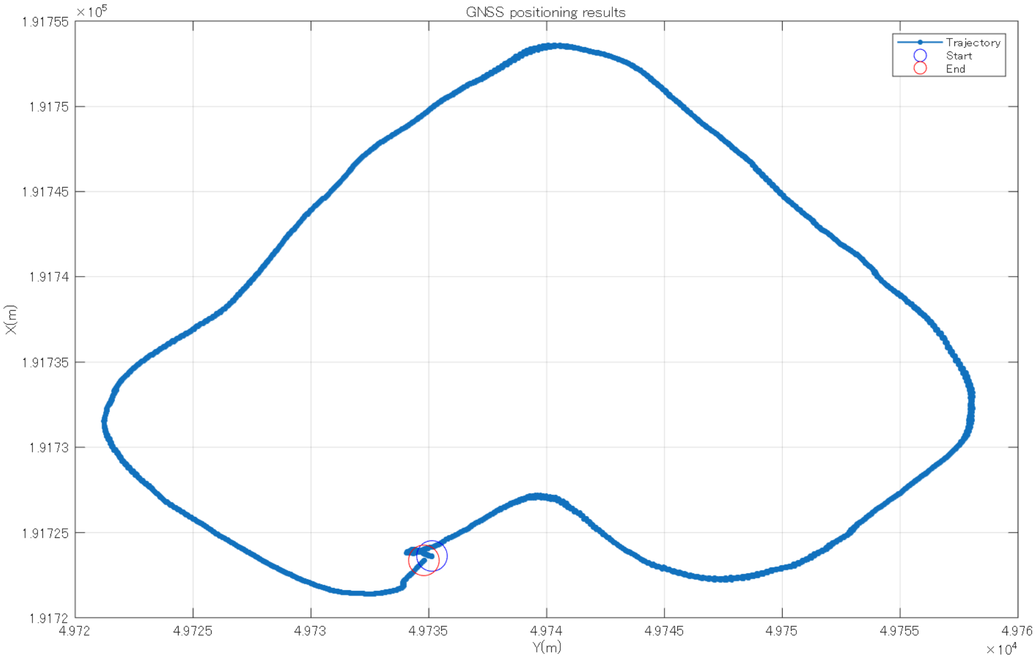

Figure 9: GNSS Positioning Results.

Figure 10 shows the result of integrating the point clouds from the first (green) and second (blue) frames. On the left side of the figure, the two point clouds are clearly separated, which visually confirms a large misalignment between the two frames. Conversely, on the right, the two clouds overlap, indicating a smaller misalignment. The left side primarily consists of plant point clouds, which are characterized by low reflectivity and irregular shapes. These plants are far from the vehicle, which likely contributing to the larger registration error. In contrast, the right side predominantly features highly planar structures such as buildings, which are closer to the vehicle. This proximity enabled stable feature capture, resulting in a smaller misalignment. These results confirm the effectiveness of preprocessing that removes unstable point clouds, such as those from plants, and prioritizes extracting stable point clouds near the vehicle.

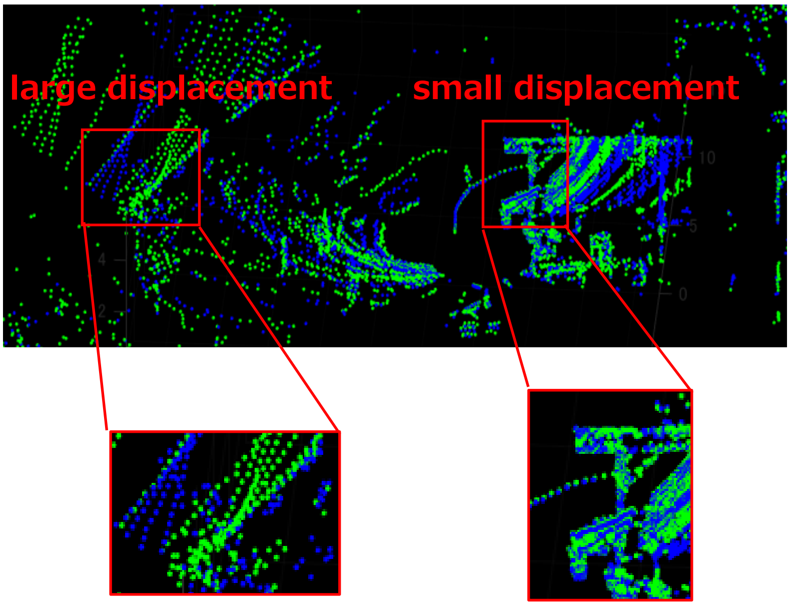

Figure 10: Inter-Frame Point Cloud Integration.

As shown in Figure 11, the point pre-processing results confirmed that by utilizing the characteristics of non-repeating scan LiDAR and performing frame integration could obtain point clouds of sufficient quality to identify building outlines and surrounding structures. Furthermore, the red points represent feature points extracted from the front areas of the vehicle, indicating that they contain useful information for shape reconstruction.

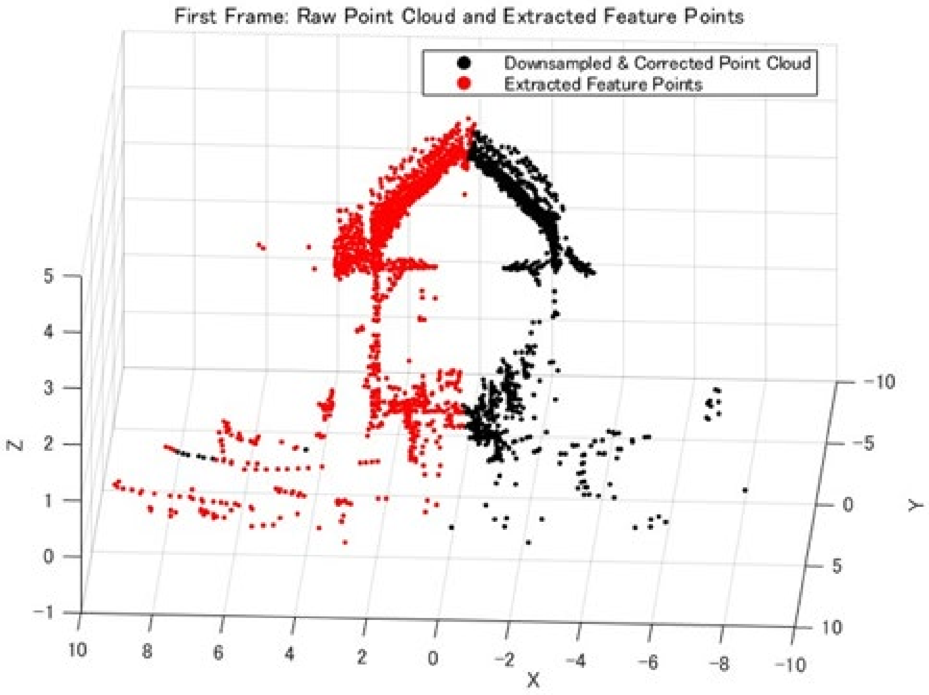

Figure 11: Point Cloud Processing Results.

As shown in Figure 12, the estimated trajectory in red superimposed on the LIO-SAM map generally aligns with the driving direction and movement direction during the experiment. Furthermore, the estimated trajectory consistently passes through high-density areas of the map. This indicates that graph optimization, including loop closure, is functioning appropriately. This ensures consistency between the map and self-position. The stability of the straight-ahead direction immediately after commencement indicates the effectiveness of the initial heading and the stabilization of the bias via IMU pre-integration and, the deskewing of the intra-frame motion. Furthermore, there was no attitude breakdown or significant ghosting occurred, even in scenarios prone to geometric degeneration, such as straight road sections or flat terrain. These results demonstrate improved observability through the combined use of LiDAR geometry and inertial constraints.

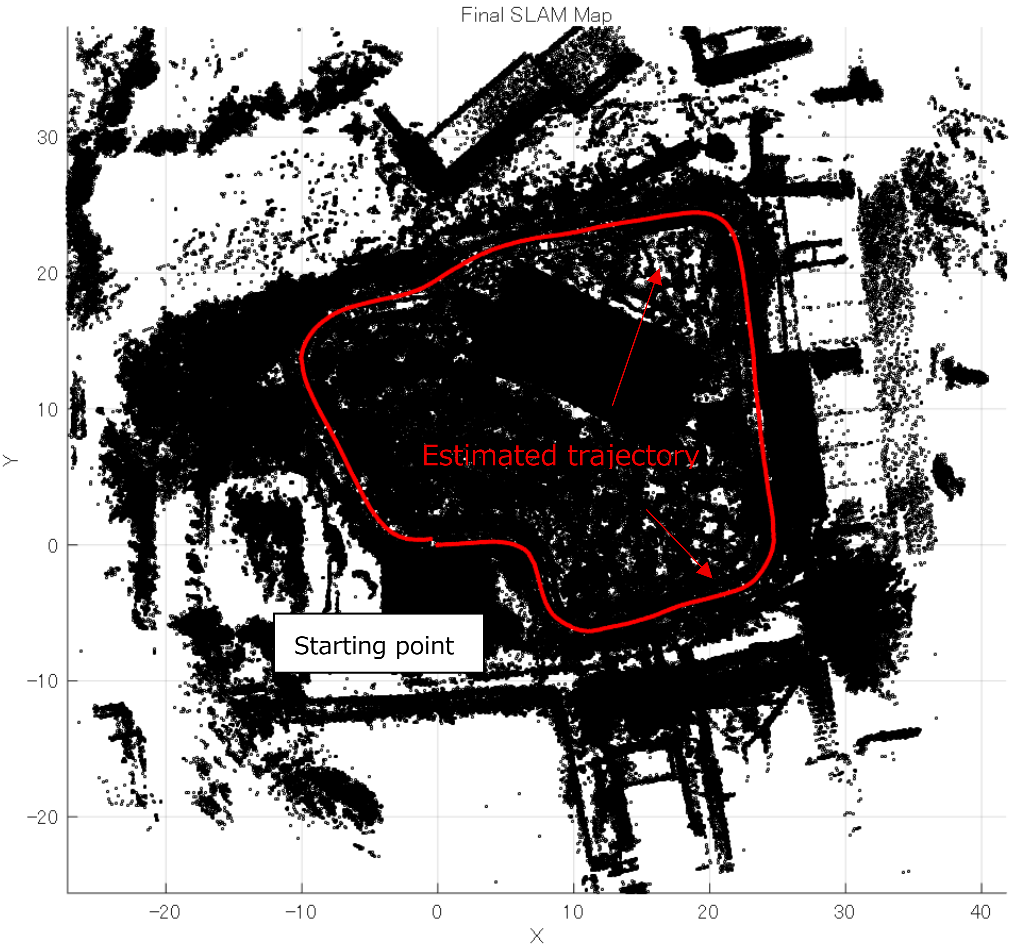

Figure 12: LIO-SAM.

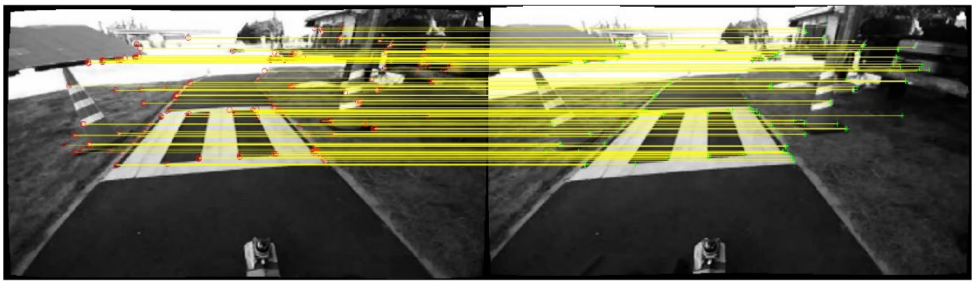

Figure 12: ORB Feature Point Matching.

As shown in Figure 13, it was confirmed that corresponding points were accurately matched between the left and right images based on ORB feature extraction. Corresponding points are arranged along the yellow peripolar lines, indicating that matching using feature descriptors operates while maintaining geometric consistency. Furthermore, the results of reconstructing three-dimensional coordinates from these corresponding points (Figure 13) show that outliers were removed using RANSAC. Although the number of points is small, the remaining points are confirmed to be geometrically reliable. Notably, the point cloud is concentrated in areas rich in features such as road markings and buildings, clearly demonstrating that a stable structure is extracted even in the initial frame. Furthermore, these results demonstrate that ORB feature points exhibit high reproducibility in texture-rich regions and, when combined with geometric constraints, can effectively suppress outlier correspondences. That is, feature-point-based stereo matching techniques have been confirmed to be effective for constructing reliable initial maps in subsequent SLAM processing.

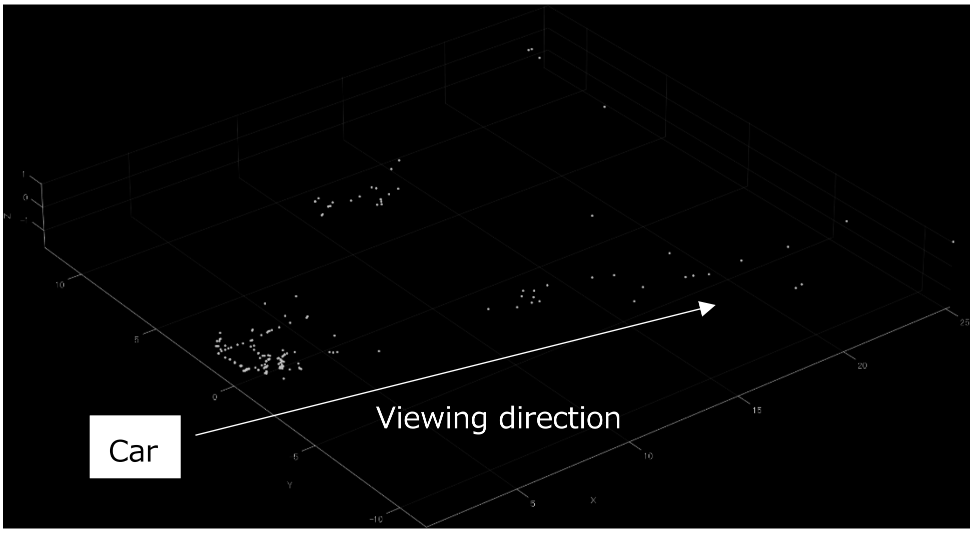

Figure 13: ORB Feature Point Cloud.

As shown in Figure 14, the trajectory estimated by Visual SLAM (red line) broadly aligned with the actual driving path and direction during the experiment, albeit with limited precision. The estimated trajectory was drawn continuously from the starting point, indicating consistent relative position estimation between frames and generally accurate capture of local movements. Furthermore, the reconstructed feature point clouds were densely distributed in areas with abundant texture, such as buildings and road markings. This indicates relatively high reliability in geometrically stable environments. Conversely, in areas with sparse texture, such as road surfaces or distant objects, the point clouds become sparse, leading to reduced estimation accuracy. These results demonstrate that feature-based Visual SLAM is highly dependent on environmental conditions. Additionally, the overall trajectory shows significant cumulative error over time, indicating a tendency for self-localization to gradually drift during long-distance travel. Therefore, while Visual SLAM is effective at grasping movement direction and local shapes, additional methodology, such as integration with IMU or GNSS, or enhanced loop closure, are necessary to ensure overall positional accuracy.

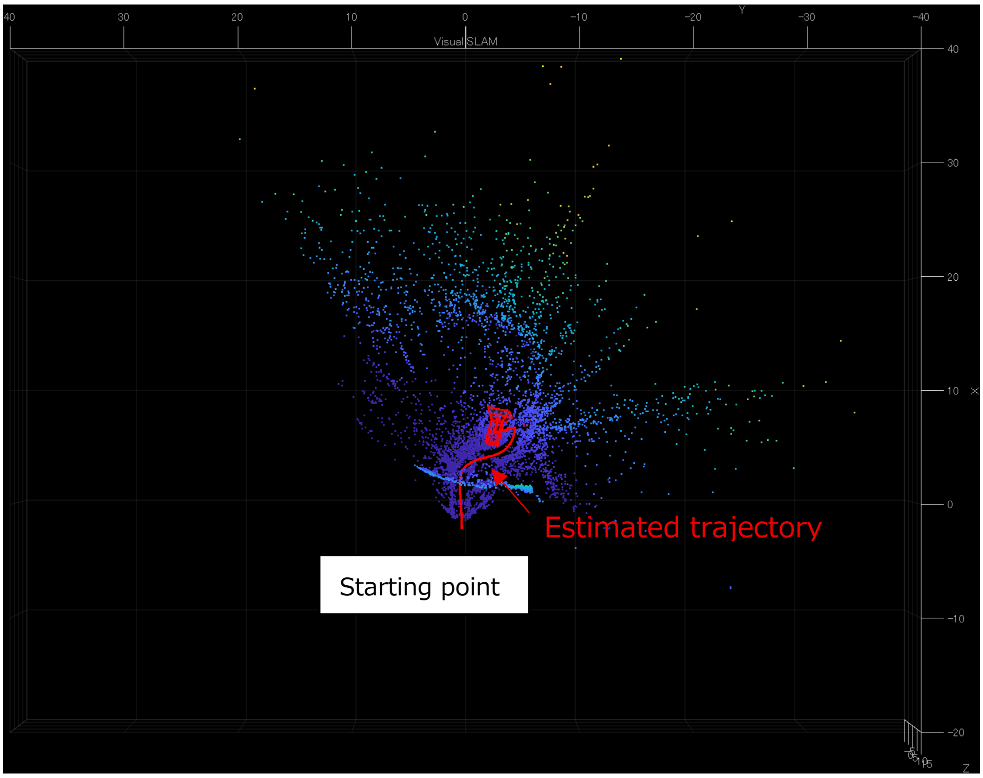

Figure 14: Visual SLAM.

As shown in Figure 15, the causes of reduced accuracy in Visual SLAM can be confirmed by the significant fluctuation in the number of stereo feature pairs with advancing frames. After frame index 600, there is a temporary sharp increase, followed by a subsequent downward trend. Similarly, a marked decrease in Tracked Map Points is observed after frame index 700, indicating a gradual loss of trackable point clouds. These trends suggest an insufficient number of stably observable feature points within the camera's field of view, which makes mapping points to the map difficult. Interestingly, even when sufficient stereo feature pairs are detected, Tracked Map Points may still decrease. This indicates that newly detected feature points are not being stably mapped to existing map points. In other words, while temporary feature point detection is possible, it does not contribute contributing to the integration into the map. Contributing factors include scenes containing numerous moving objects, areas with ambiguous descriptors such as vegetation or low-texture regions, and abrupt camera motion, which leads to large parallax or blur between frames. Furthermore, a reduction in Tracked Map Points weakens the constraints for pose estimation via PnP, making it harder to maintain consistency during bundle adjustment and loop closure. Consequently, cumulative errors can readily accumulate, leading to distortions and instability in the estimated trajectory, as illustrated in Figure 14. Therefore, to ensure the stability of Visual SLAM, it is crucial to selectively retain features that are both trackable and geometrically consistent, rather than merely increasing the number of features.

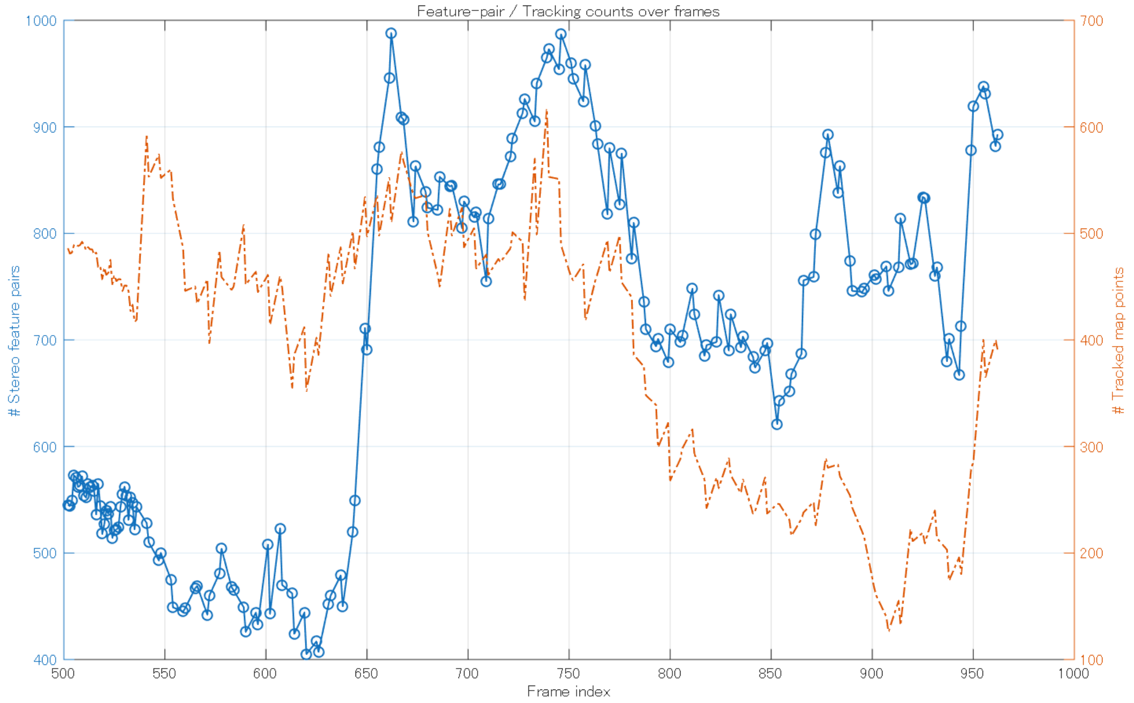

Figure 15: Stereo Feature Pairs and Tracked Map Points over Frame Index.

Figure 15 shows a significant decrease in Tracked Map Points around frame 900. As shown in Table 1, stereo feature pairs fluctuated within a range of approximately 750 to 810 from frame 902 to frame 919, exhibiting no significant overall variation. Conversely, Tracked Map Points decreased from 160 points at frame 902 to 126 points at frame 908, and remained low at 132 even at frame 914. This indicates that although a consistent number of corresponding points were not be reliably matched with existing map points and were not maintained as trackable points. Frame 908 notably shows the minimum value for Tracked Map Points and coincides with the significant deviation in self-localization observed in Figure 14. This strongly suggests that the failure to obtain sufficient geometric constraints directly led to the breakdown of self-localization. Furthermore, the abrupt increase in Tracked Map Points to 223 at frame 919 is notable. It indicates the temporary acquisition of trackable feature points, likely due to the re-entry of strong geometric constraints (e.g., building edges or road markings) into the field of view. Thus, even when stereo feature pairs are stably present, significant fluctuations in the number of Tracked Map Points can result in insufficient geometric constraints being secured for the attitude estimation required by Visual SLAM, leading to unstable estimation accuracy. Therefore, this table suggests that the primary cause of the decline in Visual SLAM accuracy is caused not only insufficient feature point detection, but also a reduction in the proportion of detected feature points that successfully establish correspondence with map points and contribute to tracking.

Table 1: Comparison Before and After the Problem Frame of Tracking Map Points.

Frame	**902**	**907**	**908**	**913**	**914**	**919**
Stereo Feature Pairs	757	769	746	768	814	771
Tracked Map Points	160	140	126	155	132	223

Table 1 indicates that frame 908 has the fewest value of Tracked Map Points (126 points), suggesting that self-localization estimation failed significantly in this frame. Figure 16 was consequently created to analyze its details. As shown in Figure 16 visualizes, the spatial distribution of feature point matching in frame 908 and reveals that the detected feature points are concentrated in the lower part of the image. Feature points corresponding to distant objects or structures in the upper part of the image are largely absent, as expected. Figure 17 illustrates the vertical distribution and shows a pronounced peak around rows 50 to 160, with the number of matches decreasing significantly beyond this range. This indicates that the feature points were not uniformly distributed across the entire scene and lost the geometric constraints from distant features, which are crucial for self-localization. Furthermore, feature points detected based on local textures such as road surfaces and shadows, are challenging to track consistently between frames. This is thought to have led to a substantial reduction in Tracked Map Points. The primary factors causing the significant breakdown in self-localization at frame 908 are considered to be the skewed spatial distribution of detected feature points and the reduced trackability, rather than the absolute number of detected points.

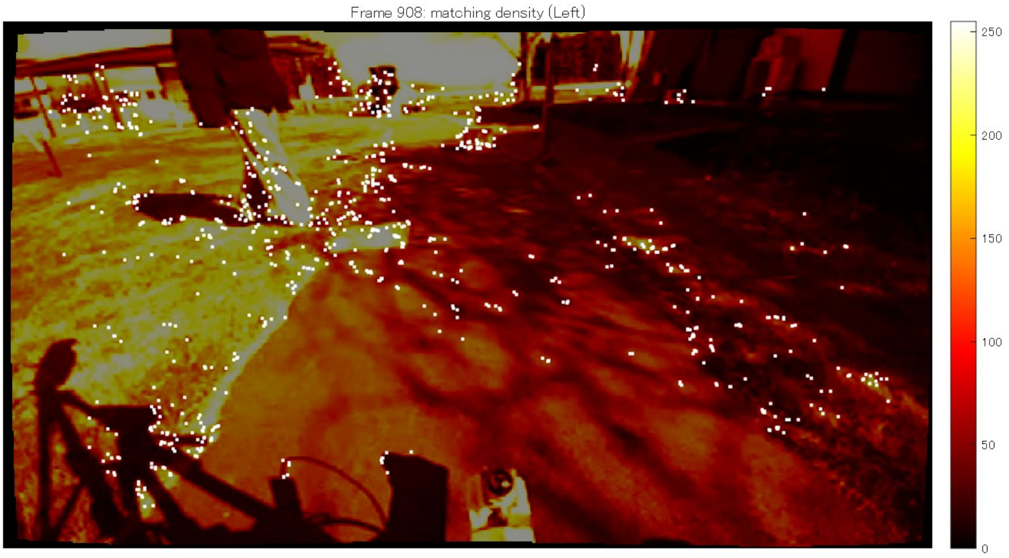

Figure 16: Visualization of the Spatial Distribution of Feature Point Matching in Frame 908.

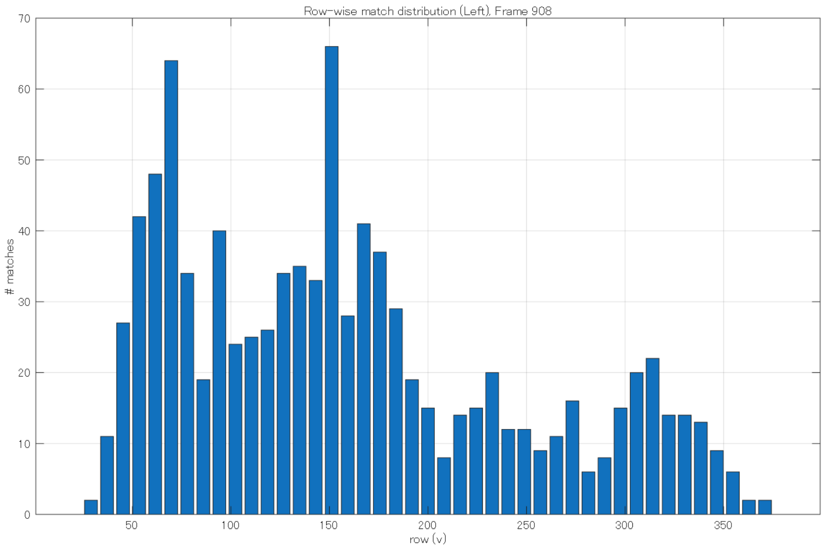

Figure 17: Bar Chart showing Vertical Distribution.

Conclusion and Recommendation

This study compared and verified GNSS positioning and point cloud preprocessing using non-repeat-of-scan LiDAR, LIO-SAM, and Visual SLAM using stereo cameras, clarifying the characteristics and challenges of each methodology. Although GNSS showed good overall consistency, it retained cumulative error. Point cloud preprocessing identified structures by removing redundant points and extracting neighboring points. LIO-SAM achieved stable self-localization by combining an IMU and LiDAR. However, Visual SLAM exhibited accuracy degradation during prolonged estimation. Notably, it reached a minimum of Tracked Map Points at frame 908, causing significant self-localization deviation. Heatmap analysis confirmed that this occurred due to uneven feature point distribution and reduced trackability. This indicates that spatial distribution and stability of feature points, rather than their sheer number, are key to self-localization. Future work should focus on enhancing the robustness of Visual SLAM through the uniform extraction of feature points, the removal of moving objects, and the improved integration of LiDAR and IMU.

References

Harun, M. H. (2022). Sensor Fusion Technology for Unmanned Autonomous Vehicles (UAV): A Review of Methods and Applications. pp. 1–6.

Cadena, C. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. pp. 1309–1332.

Rublee, E. (2011). ORB: An efficient alternative to SIFT or SURF. Proceedings of the 2011 International Conference on Computer Vision (ICCV), pp. 2564–2571.