Streaming Point Cloud Segmentation of GNSS/SLAM-LiDAR and Multi-beam Scanning Sonar Data for Urban River Mapping

Masafumi Nakagawa [1*], Naoto Kimura [1], Nozomi Sadachika [1],

Takeshi Komori [2], Nobuaki Kubo [2], Etsuro Shimizu [2]

[1] Shibaura Institute of Technology, Japan

[2] Tokyo University of Marine Science and Technology, Japan

*mnaka@shibaura-it.ac.jp

Abstract: *Simultaneous localization and mapping (SLAM) can provide point clouds and self-position data for 3D mapping, virtual reality, augmented reality, mixed reality, UAV flight control, and autonomous vehicle control. In 3D mapping, 3D object recognition based on model fitting and machine learning is required for the automation of 3D mapping and modeling using dense point clouds generated by SLAM and mobile mapping. The machine learning-based 3D object recognition is classified into clustering and segmentation. The segmentation is fundamental processing for 3D mapping, scan-to-BIM, and autonomous vehicles using point clouds. The segmentation can be classified into image-based and 3D-based approaches. Each approach has strengths and weaknesses, thus we focus on the integration of image-based and 3D-based approaches embedded in SLAM and mobile mapping. In this research, we propose a methodology to improve the performance of streaming point cloud processing based on image-based and 3D-based point cloud segmentation for 3D mapping of urban river environments. We also developed a methodology of streaming point cloud segmentation embedded in GNSS/LiDAR-SLAM and multi-beam scanning. In our experiments, we used a water-borne mobile mapping system at an urban river as GNSS and non-GNSS environments. We acquired dense streaming point clouds above water surfaces with GNSS/LiDAR-SLAM consisting of two LiDARs and precise point positioning based on real-time kinematic positioning with a centimeter-level augmentation service using a quasi-zenith satellite system. In parallel, we also acquired dense streaming point clouds underwater surfaces with a multi-beam scanning sonar with RTK-GNSS positioning. Moreover, we experimented with streaming point cloud segmentation of acquired massive point clouds to classify streaming point clouds into bridges, revetments, buildings, and underwater surfaces. Through the experiment using the streaming point clouds, we confirmed that our methodology can improve the scalability of point cloud processing and achieve high-speed processing and precise classification as well as conventional image-based and 3D-based point cloud segmentation approaches.*

Keywords: Streaming point cloud, Point cloud segmentation, Simultaneous localization and mapping, LiDAR, Multi-beam scanning sonar

Introduction

a. SLAM and point cloud processing:

Simultaneous localization and mapping (SLAM) is a process that simultaneously estimates self-position and point clouds for autonomous mobile robots and other applications such as virtual reality, augmented reality, and mixed reality. In the field of surveying and construction, SLAM is a popular methodology to simplify point cloud acquisition in real-time processing or post-processing for ICT construction and digital twinning of urban areas. SLAM can be classified by input data into visual SLAM (SLAM using streaming images), Depth SLAM (SLAM using streaming depth images), and LiDAR-SLAM (SLAM using streaming LiDAR point clouds). In the field of surveying and construction, LiDAR has advantages, such as a wider 3D measurement range and more resistance to illumination changes, thus, it is often used to measure large areas and objects under changing weather conditions. Moreover, LiDAR-SLAM is used to improve the efficiency of surveying in local areas using UAV-LiDAR, handheld LiDAR, and wearable LiDAR.

When utilizing point clouds acquired by LiDAR-SLAM for topographic mapping, form surveying, ground surveying, or indoor space mapping, real-time processing is not required in most cases at present. Errors are adjusted by post-processing after the measurement, then, point cloud mapping, 3D modeling, and 3D mapping are applied. Therefore, the automatic processing of LiDAR-SLAM is used for only point cloud acquisition in the current state. However, in the advanced digital twinning of cities and construction, point clouds obtained in automated driving and construction may be used as high-resolution and high-frequency measurement data of cities. Moreover, there is a possibility to discuss a new market of mapping for autonomous robots, such as automated driving vehicles, unmanned construction vehicles, unmanned aerial vehicles, and autonomous boats. Thus, more massive point clouds will be used for mapping and map revision with real-time processing. Although there are various technical issues to be addressed such as the development of high-capacity communication infrastructure, the most technical issue is that it will be hard to apply a conventional mapping approach using point clouds with automated systems that include a lot of visual interpretation work. To address this technical issue, we focus on pipeline processing-based mapping and modeling from streaming point clouds obtained using LiDAR-SLAM. Moreover, although mesh modeling and voxelization are required in conventional 3D mapping and modeling, object recognition is also required for the full automation of 3D mapping and modeling. The object recognition processing based on machine learning can be classified into clustering and segmentation. In clustering, point clouds are grouped by statistical processing. In segmentation, point clouds are grouped based on predefined rules and knowledge. When the estimation of recognized object attributes is automated, segmentation is applied.

Segmentation of point clouds is common processing in 3D mapping, scan-to-BIM, and automated driving, where deep learning has been increasingly applied.

b. Point cloud segmentation:

Conventional studies on point cloud segmentation are categorized into image-based segmentation and segmentation in 3D space. There are methodologies for image-based segmentation, such as PointSeg (Wang et al., 2018), RangeNet++ (Milioto et al., 2019), and SqueezeSegV3 (Xu et al., 2020). PointSeg is a semantic segmentation methodology for road object extraction that applies a convolutional neural network (CNN) to LiDAR point clouds converted to a panorama image. The PointSeg is based on SqueezeNet (Iandola et al., 2017), which achieves the same performance as AlexNet (Krizhevsky et al., 2012), a CNN for object recognition from images, with only 2% of the number of parameters.

PointSeg achieves both processing efficiency and prediction performance by a model learned with a label mask and projected panorama image of KITTI data (Geiger et al., 2012) consisting of LiDAR point clouds and stereo images oriented with GPS/IMU mounted on a vehicle. The processing also performs real-time semantic segmentation at 90 fps on a single GPU. RangeNet++ is the methodology using depth images, and SqueezeSegV3 is the methodology with the similarities and differences between the image features of camera images and point clouds. PointSeg, RangeNet++, and SqueezeSegV3 have the same main target, such as automatic driving and autonomous robots, and the same approach based on point clouds rendering into range images to achieve real-time processing.

There are several methodologies for point cloud segmenting in 3D space, such as Volumetric CNN (Qi et al., 2016a), PointNet (Qi et al., 2016b), and PointNet++ (Qi et al., 2017). Volumetric CNN performs 3D convolution based on the processing of converting a point cloud into voxels and binarizing the data within each voxel. The processing cost is high and local features of the point clouds are lost due to voxel data convolution. PointNet combines the features of the entire point clouds into local features to avoid the difficulty of unordered point clouds. Moreover, PointNet uses T-Net, a neural network that outputs an affine matrix when point clouds are input, to keep the invariance of the point cloud to rotation and translation by aligning the direction of the input point clouds. However, information missing of neighboring point clouds occurs. Therefore, in PointNet++, learning local features of point clouds is applied through an iterative processing of inputting point clouds clustered with sampling and grouping to the PointNet for more accurate segmentation of point clouds.

c. Objective:

We focused on preprocessing consisting of simple clustering and segmentation of point clouds, which tends to improve the processing cost and prediction accuracy in deep learning of point clouds. Moreover, although there are many previous studies on the segmentation of streaming point clouds for road spaces, there are no examples of applications for streaming point clouds measured from boats on urban rivers, because of few training data. Therefore, we aimed to develop a simple segmentation methodology of streaming point clouds used for preprocessing such as an automatic labeler for annotation to improve the performance of deep learning with streaming point clouds.

The image-based point cloud segmentation methodology can improve processing cost and time. Thus, the image-based point cloud segmentation methodology is suitable for the segmentation of large-scale streaming point clouds. However, the number of geographic features in a frame depends on the scanning range per frame. Furthermore, it is not easy to segment objects that require observation in multiple frames to map a single geographic feature. By contrast, the methodology of 3D spatial segmentation of point clouds is suitable for extracting many features from multiple frames of streaming point clouds covering large objects. However, the processing cost is worse than the image-based point cloud segmentation methodology. Therefore, we focus on these advantages and disadvantages, and we propose a methodology of streaming point cloud segmentation that combines a high-speed processing function of the image-based point cloud segmentation and the function of feature extraction from point clouds in the methodology of 3D spatial segmentation of point clouds. Furthermore, by embedding the proposed methodology into the LiDAR-SLAM processing, we verify that our proposed methodology can achieve the improvement of the segmentation cost, the simultaneous mapping with point cloud acquisition, and 3D modeling. We use a streaming point cloud dataset obtained with a water-borne mobile mapping system for the validation experiments because the proposed methodology combines LiDAR data, multi-beam scanning sonar data, external orientation parameters estimated with SLAM and GNSS, and GPS time.

Methodology

a. GNSS/LiDAR-SLAM:

GNSS/LiDAR-SLAM is a processing that simultaneously achieves seamless GNSS/non-GNSS positioning and point cloud generation by integrating GNSS positioning and external orientation parameter estimation using LiDAR-SLAM (Nakagawa et al., 2023). The methodology consists of LiDAR-SLAM based on iterative closest point scan matching (Chen et al., 1992), correction of external orientation parameters using GNSS positioning results, and point cloud integration using external orientation parameters (Figure 1).

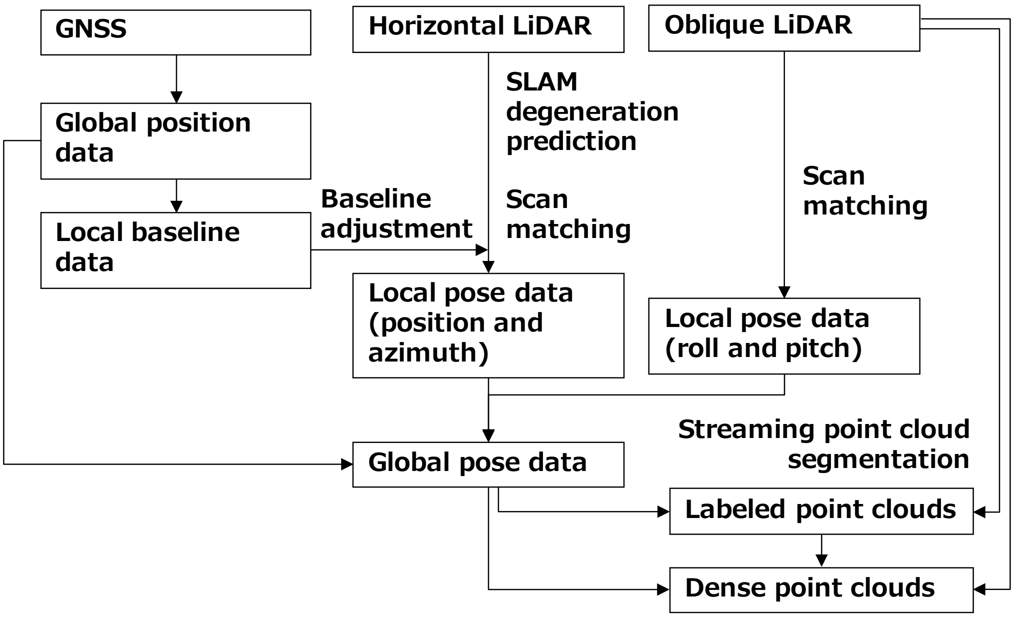

Figure 1: Proposed methodology.

The LiDAR data for GNSS/LiDAR-SLAM is acquired using multi-layered LiDAR, which can acquire point clouds including multiple cross-sectional information. Our proposed GNSS/LiDAR-SLAM with seamless GNSS/non-GNSS positioning is a loosely coupled processing integrated GNSS data as the reference data and LiDAR-SLAM results as inertial data. In the processing, GNSS position data are used for data rectification of accumulated errors in the self-position and attitude estimation using LiDAR-SLAM. The GNSS/LiDAR-SLAM can omit a loop closure for the accumulated error adjustment; thus, this methodology is suitable for measurement in rivers where loop closure cannot be formed.

Precise point positioning (PPP) using the centimeter-level augmentation service (CLAS) is used as real-time kinematic PPP (RTK-PPP) (Mimura et al., 2020) for global position data acquisition in GNSS/LiDAR-SLAM. The PPP-RTK positioning using CLAS is slightly less accurate than RTK-GNSS positioning and cannot be used outside of Japan. However, the PPP-RTK methodology using CLAS has high availability and convenience because positioning is available without reference station installation, constant communication with the reference stations, and consideration of baseline length constraints to keep positioning accuracy.

b. Segmentation of streaming point clouds:

An overview of the segmentation of streaming point clouds is shown in Figure 2. In the segmentation of streaming point clouds, range image generation, label image generation, and point cloud labeling are applied to the point clouds obtained in each scene as the basic processing. The labeled point clouds are integrated into labeled point clouds of the following scenes along the horizontal axis direction in Figure 2.

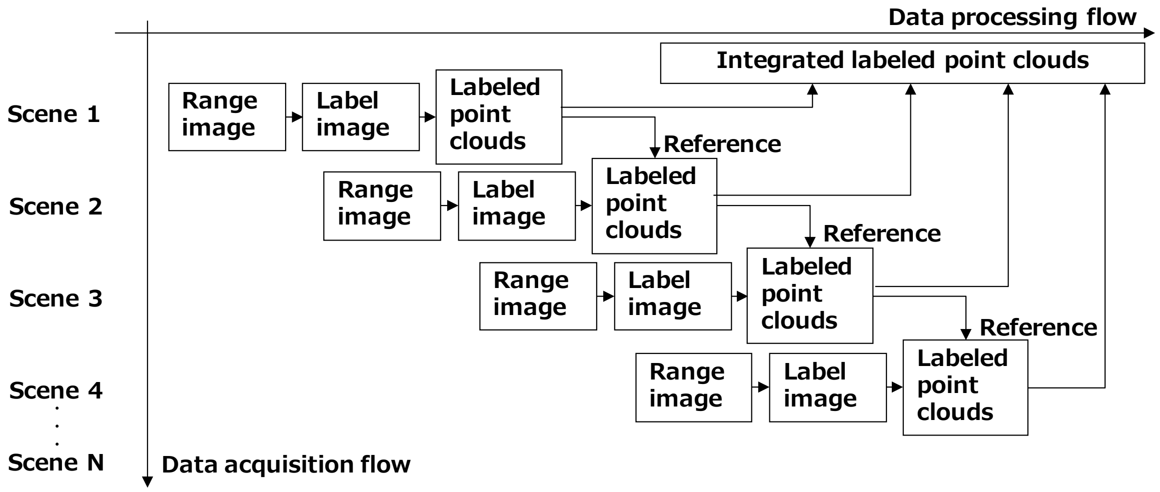

Figure 2: Streaming point cloud segmentation.

In range image generation using LiDAR data, 3D range images are generated with the x-axis as scanning lines, the y-axis as LiDAR channels, and the z-axis as point cloud coordinate values (X, Y, Z), LiDAR reflection intensity values, and label information, as shown in Figure 3. In the z-axis direction, identified label numbers and normal vectors (in X, Y, and Z directions) can be added to the range images. Moreover, in the range image generation using multi-beam scanning sonar data, 3D range images are generated with the x-axis as scanning lines, the y-axis as sonar channels, and the z-axis as point cloud coordinate values (X, Y, Z) and label information.

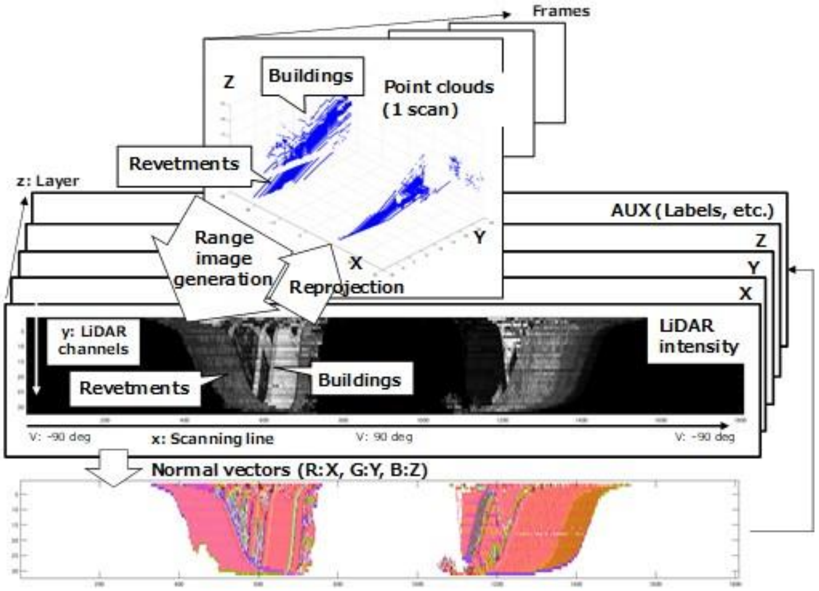

Figure 3: Range images from scanning data.

Figure 4 shows the details of the label image generation and point cloud labeling in Figure 2 as an overview of the labeling processing in the segmentation of streaming point clouds. The vertical axis indicates each segment and the horizontal axis indicates the data acquisition flow. The labeling process consists of four types of processes: new label creation, label connection, label integration, and labeling termination. When no segment corresponding to the previous scene exists, a new label is created. When no segment corresponding to the next scene exists, labeling is terminated. The label connection assigns the same label number when segments are mapped between consecutive scenes. In the label integration, when segments with different labels are label-connected in the mapping of segments between consecutive scenes, the connected segments are overwritten with the same label to avoid inconsistencies in labeling.

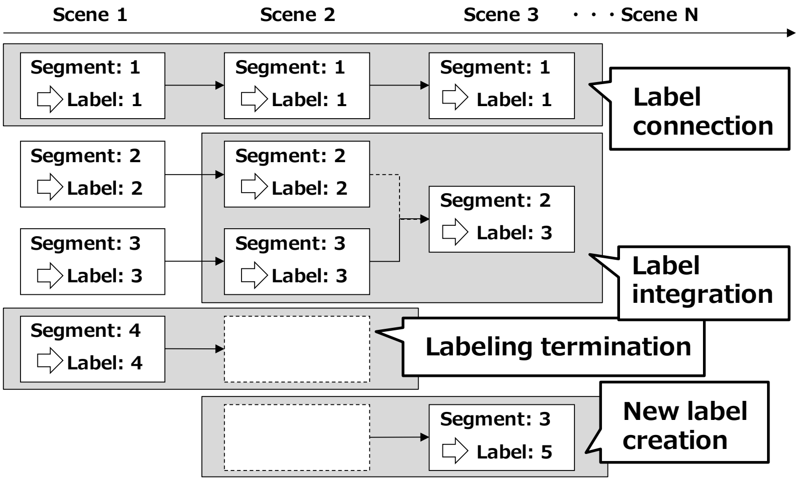

Figure 4: Labeling processing in streaming point cloud segmentation.

The results obtained from the streaming point cloud segmentation are managed in a segment layer that handles labeled segments obtained from each scene, a cluster layer that handles segments integrated by labels, and a 3D model layer (CityGML layer) that handles the semantics of segments in the cluster layer (Figure 5). In this study, we classify point clouds through the streaming point cloud classification into bridges, revetments, and buildings.

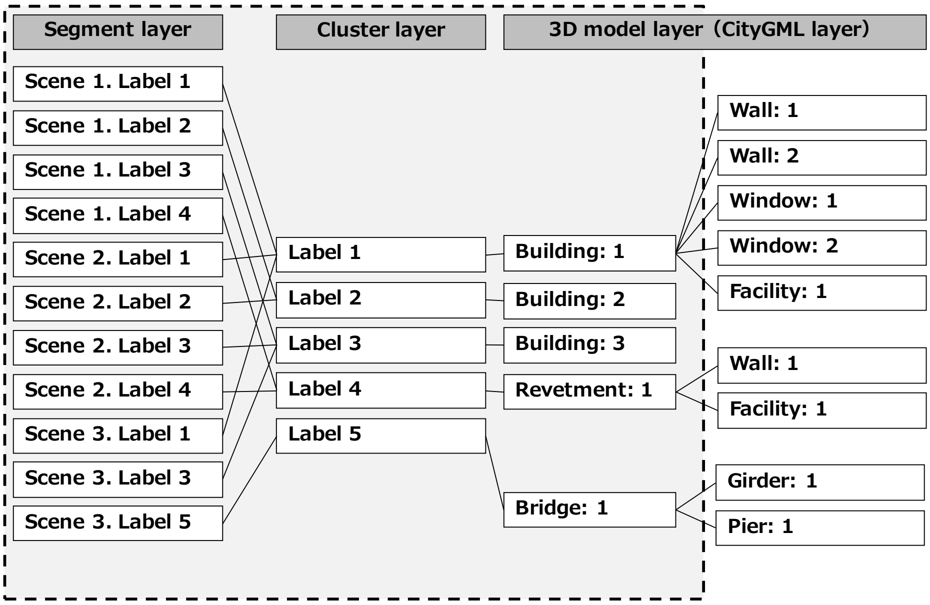

Figure 5: Feature classification using labeled point clouds.

c. Classification of streaming above water point clouds:

In the streaming point cloud classification, bridges, revetments, and buildings are extracted and classified, as shown in Figure 6. First, bridges are extracted from point clouds using bounding boxes generated from the river width and bridge height estimated from LiDAR scanning data. After bridge extraction, revetments and buildings are extracted from point clouds, as shown in Figure 7. First, the point clouds are rendered into a range image. Next, the height of the water surface is estimated from the scanning line. Then, the top edge of the revetment is estimated using the discontinuity from the water surface on the scanning line. The height of the water surface is obtained from the change points on the scan line that are lower than the LiDAR position and extracted using the plane symmetry of the point clouds due to the specular reflection of LiDAR. Using the top edge of the revetment as the threshold of approximate ground level, revetments and buildings are classified from the point clouds. The threshold is determined independently for the left and right revetments.

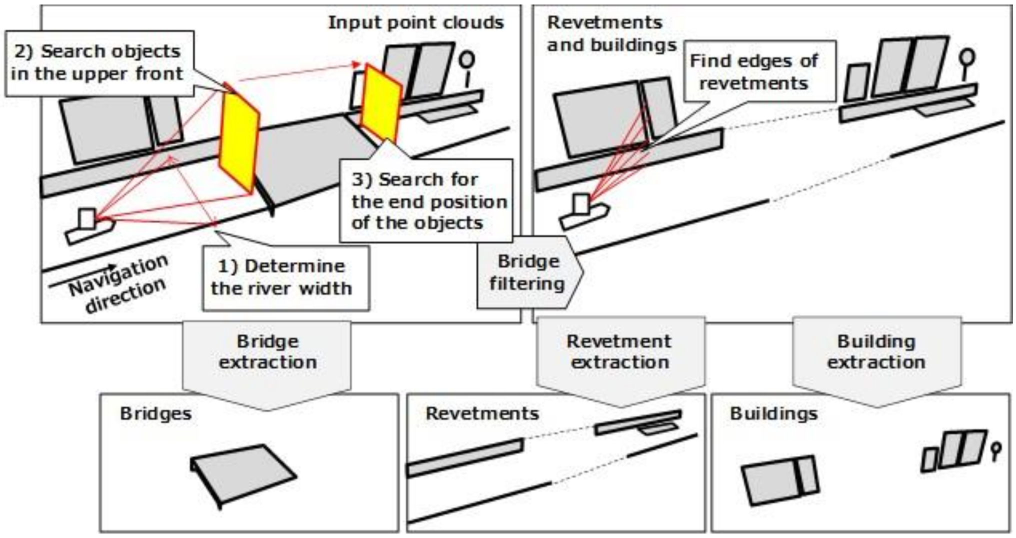

Figure 6: Feature classification using streaming point clouds (bridges, revetments, and buildings).

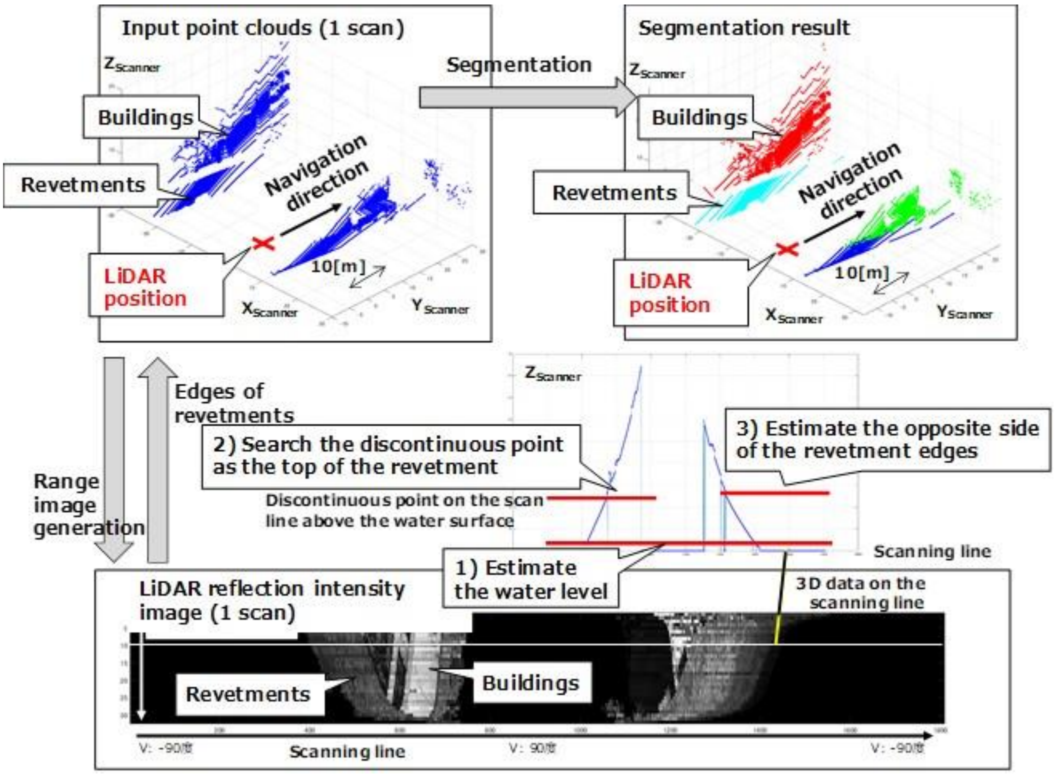
Input point clouds (1 scan)
Segmentation
Segmentation result
Buildings
Revetments
Navigation direction
LiDAR position
10[m]
Edges of revetments
Range image generation
2) Search the discontinuous point as the top of the revetment
Discontinuous point on the scan line above the water surface
3) Estimate the opposite side of the revetment edges
1) Estimate the water level
Scanning line
3D data on the scanning line
LiDAR reflection intensity image (1 scan)
V: -90度
V: 90度

Figure 7: Feature classification using streaming point clouds (revetments and buildings).

The building extraction from point clouds is shown in Figure 8. First, point clouds in space higher than an arbitrary height (one or two floors of a building) are selected so that the buildings are not connected by neighbor ground objects in the segmentation. Next, the selected point clouds are projected into a 2D cell space set in the horizontal plane for building labeling. Furthermore, when walls and windows are extracted from the point clouds, we can use edges and surfaces extracted depth and normal vector images generated from point clouds through point cloud projection processing, as shown in Figure 3.

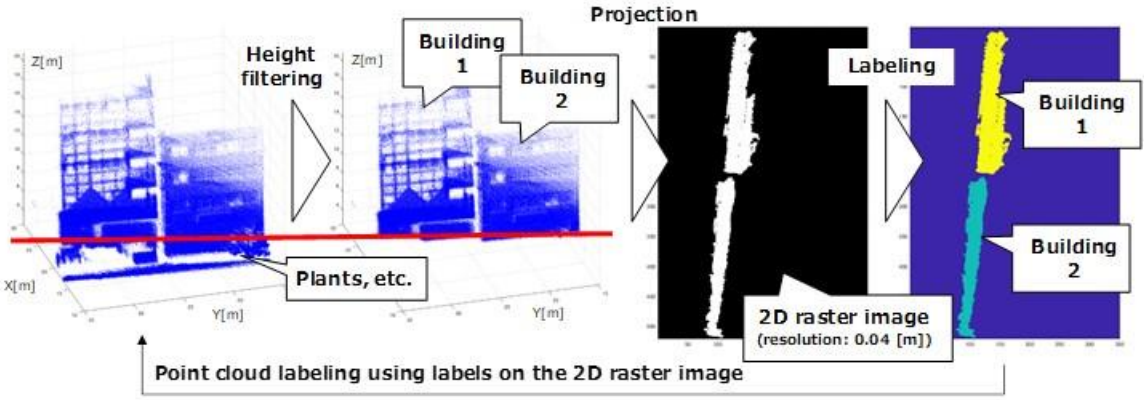

Figure 8: Building labeling with 2D cell space.

d. Classification of streaming underwater point clouds:

3D underwater maps are required for boat navigation in urban rivers for collision and stranding avoidance. Thus, we summarize the possible features that can be estimated from underwater point clouds, as shown in Table 1.

Table 1: Feature modeling from underwater point clouds.

Objects	Attribute information for boat navigation	Geometric information for boat navigation
Revetments	- Concrete surfaces	Objective distance in horizontal direction
	- Steel piles	
Others	- Bridge piers, abutments, mooring piles, pillars	
	- Ropes and large garbage	Objective distance in height direction
Soil surfaces	- Sediment and vegetation	
Reference data	⇧ Point cloud classification results	⇧ DSM

In collision and stranding avoidances, a DSM generated from underwater point clouds is used for navigation to estimate the distance from a boat to an object. Moreover, when using underwater point clouds for navigation, the navigation uses geometrical information with attribute information of objects. Thus, based on the survey of geographic features that can be estimated from underwater point clouds, we classify the underwater point clouds into three types: revetments, sediment surfaces, and other features.

Point cloud classification methodologies are classified into unsupervised learning, supervised learning, and knowledge-based classification. Underwater point clouds measured in shallow water areas such as urban rivers include large sparse differences and many missing regions due to occlusion. Thus, the underwater point clouds include many scenes that are unsuitable for unsupervised and supervised learning. By contrast, when we use structured point clouds or random point clouds with large sparse differences and large missing regions, knowledge-based classification is often better in terms of computational cost. Therefore, knowledge-based classification is applied for underwater point cloud classification, as shown in Figure 9.

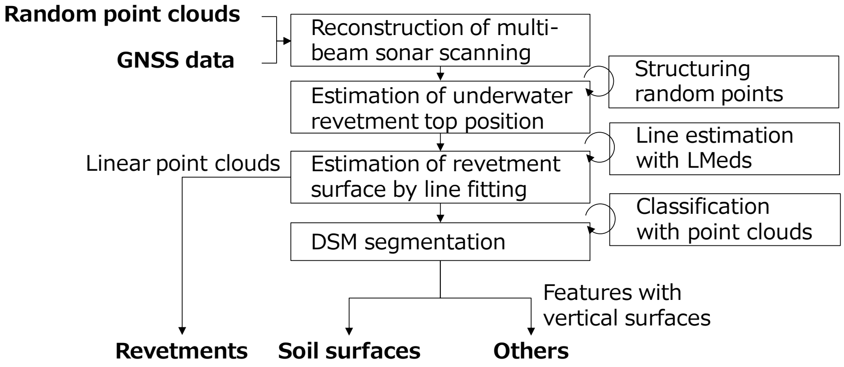
Random point clouds
GNSS data
Reconstruction of multi-beam sonar scanning
Structuring random points
Estimation of underwater revetment top position
Line estimation with LMeds
Linear point clouds
Estimation of revetment surface by line fitting
Classification with point clouds
DSM segmentation
Features with vertical surfaces
Revetments
Soil surfaces
Others

Figure 9: Classification of underwater point clouds.

Random point cloud structuring is a simple process to reconstruct multi-beam sonar scanning from random point clouds using GNSS positioning data. The processing simplifies the segmentation processing, such as RANSAC and region growing-based segmentation (Ni et al., 2016) because scanning angles can be used for the top edge estimation of underwater revetments and reduction of the number of thresholds in feature extraction, as shown in Figure 10. The threshold values for underwater point cloud segmentation in our research are only the initial slope angle of revetment (75°) and the water depth (2 m below the water surface) for obstacle detection.

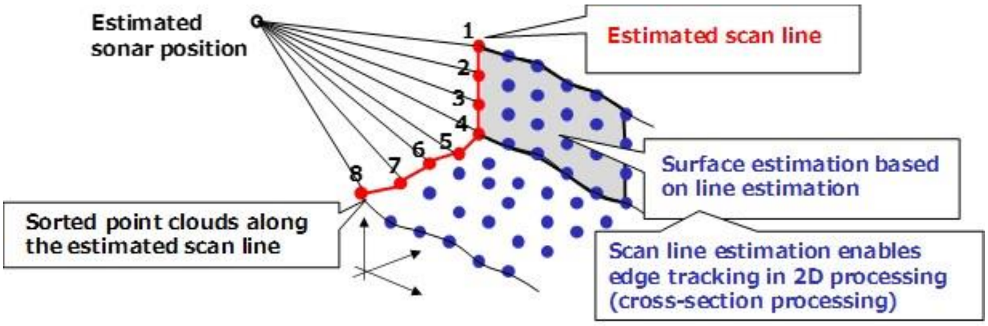
Estimated sonar position
1
2
3
4
5
6
7
8
Estimated scan line
Surface estimation based on line estimation
Sorted point clouds along the estimated scan line
Scan line estimation enables edge tracking in 2D processing (cross-section processing)

Figure 10: Structured data generation from random point clouds.

Experiments

a. Water-borne mobile mapping system:

We prepared a water-borne mobile mapping system mounted on an autonomous battery-powered boat (Raicho I), as shown in Figure 11. We used horizontal scanning LiDAR (VLP-32C, Velodyne) and oblique LiDAR (VLP-32C, Velodyne) for dense streaming point cloud acquisition and pose estimation. The LiDAR data were acquired at 10 Hz. Moreover, we used a GNSS antenna (GPS-703-GGG, Novatel) connected to a GNSS receiver (AsteRx4, CORE) for PPP based on real-time kinematic positioning with CLAS using quasi-zenith satellite systems. We also used an RTK-GNSS receiver (ZED-F9P, u-blox) for position error evaluation. The GNSS position data were acquired at 5 Hz. In addition, we used a multi-beam scanning sonar (BV5000, Teledyne BlueView) with RTK-GNSS positioning for dense streaming point clouds on underwater surfaces.

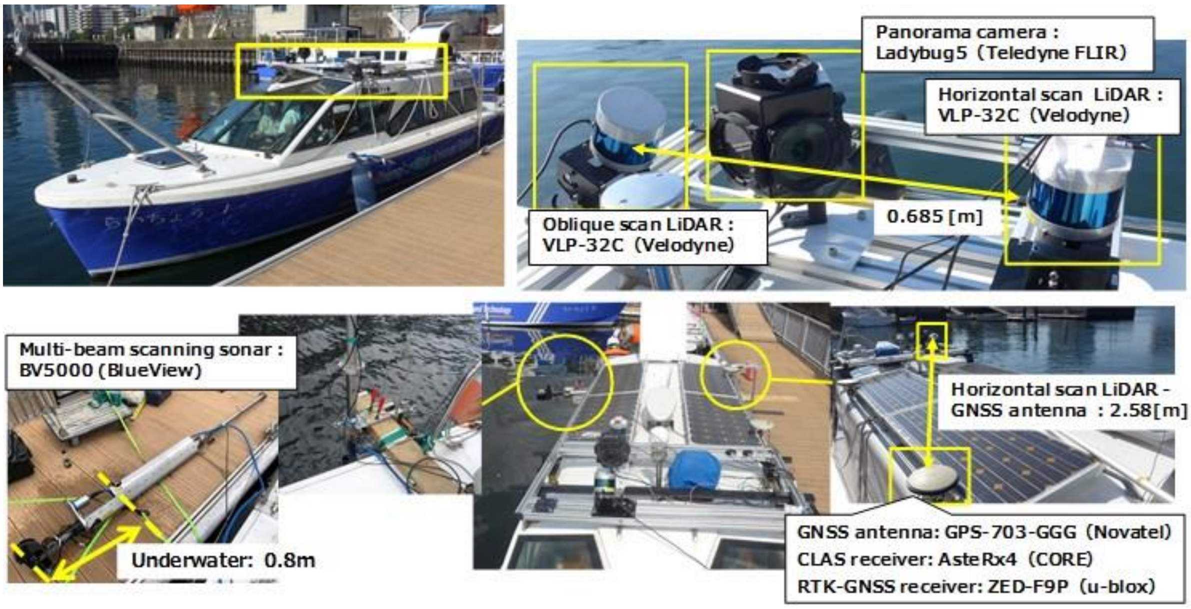

Figure 11: Water-borne mobile mapping system.

b. Data acquisition of GNSS/SLAM-LiDAR point clouds:

We acquired dense point clouds above water surfaces with GNSS/LiDAR consisting of two LiDARs and PPP based on real-time kinematic positioning with a CLAS using a quasi-zenith satellite system. The navigation route was from Tokyo University of Marine Science and Technology (Etchujima) to the Sumida-gawa River, Kanda-gawa River, Nihonbashi-gawa River (Figure 12), and back to the Etchujima campus between 10:00 and 11:30 on October 13, 2023, when the tidal level in Tokyo Bay was low, and each LiDAR acquired 66,500 frames (6,650 seconds) of point clouds (2.6 billion points in total) with the navigation speed from 4 to 6 knots in a nearly windless environment. The point clouds generated with the GNSS/LiDAR-SLAM processing are shown in Figure 13. From the measurement routes, the section from the Izumi-bashi bridge to the Shinkansen bridge on the Kanda-gawa River was selected for our experiment, as shown in Figure 14, because the section had difficulty for CLAS to estimate a re-FIX solution due to the surrounding buildings even though no obstructions existed in the zenith direction. We processed 76 million streaming point clouds (1,180 frames) with a point density of approximately 0.05 m.

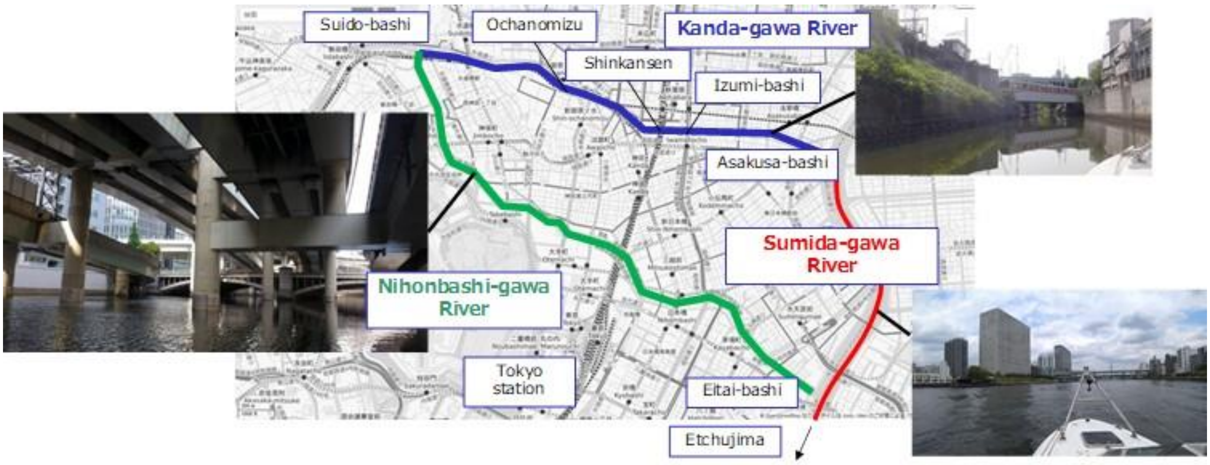

Figure 12: Navigation path.

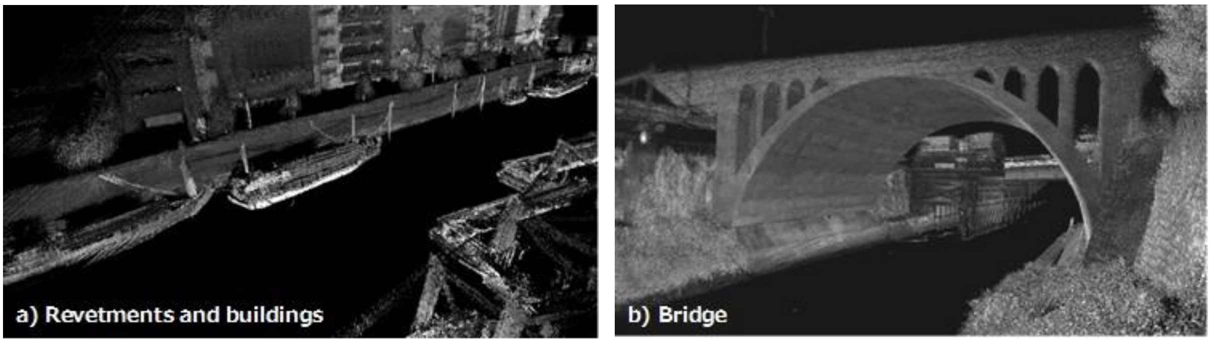

Figure 13: Point clouds generated with GNSS/LiDAR-SLAM.

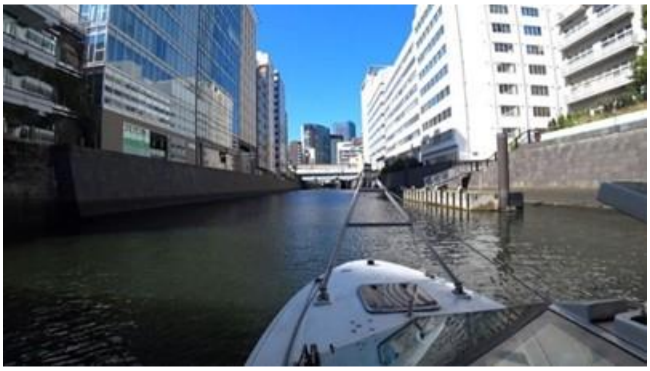

Figure 14: Selected section for LiDAR point cloud processing.

Figure 15 shows an overview of the streaming point clouds used in our experiment. Image a is the point clouds obtained by oblique scan LiDAR at a scan, image b is a part of the point clouds integrated with the position and attitude estimation results, and image c is the oblique scan LiDAR strip image. The strip image is a rendered image with the vertical angle of the scanning line in the vertical direction and the frame number in the horizontal direction as a quick-look image of the measured section. In our experiment, the LiDAR reflection intensity values on channel 17 (center channel of LiDAR) of the oblique scan LiDAR were assigned to each pixel in the strip image.

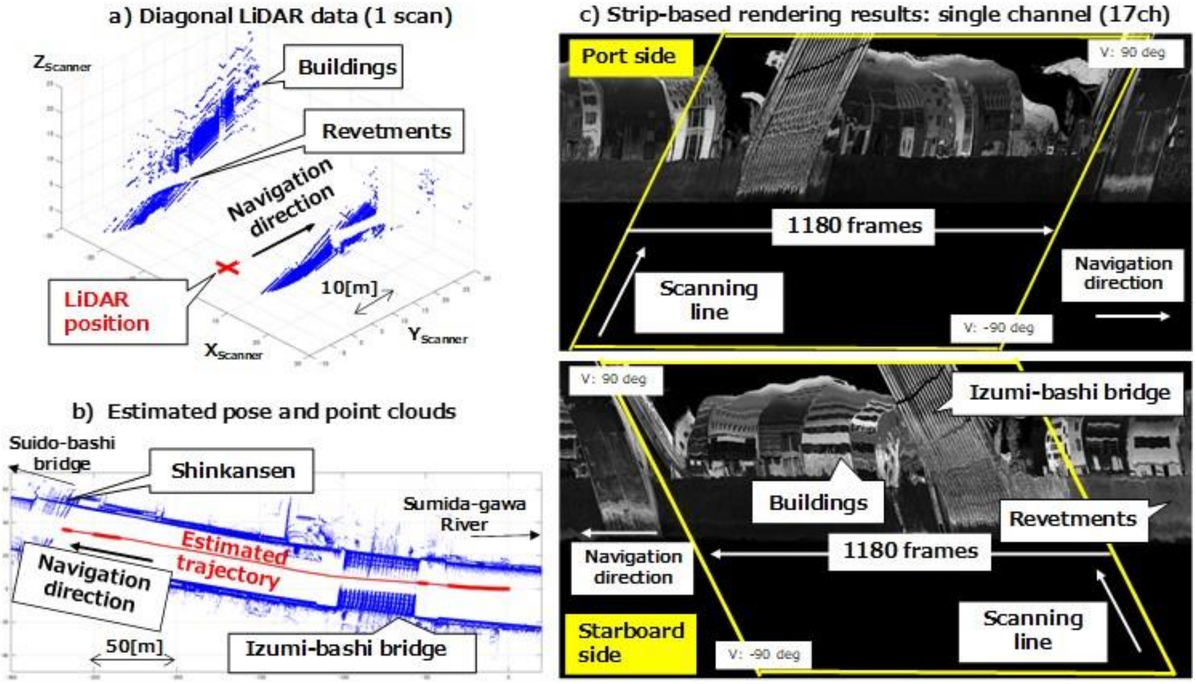

Figure 15: Clipped streaming point clouds.

c. Data acquisition of multi-beam scanning sonar point clouds:

On September 15, 2023, at low tide, we acquired 17 million points at the section from Sumida River to Ochanomizu along the left and right banks of the Kanda-gawa River during two round trips. The navigation speed was approximately 3 knots to prevent the attitude of the mounted transmitter/receiver from being displaced by water currents. The point cloud density was 0.06 to 0.10 m in the navigational direction and 0.04 m in the scanning direction. The point clouds at the bottom of the river were not acquired because the water depth was too shallow to be measured (Figure 16). Although only the point clouds in the RTK-Fix solution should be processed for the measurement by the multi-beam scanning sonar because the result depends on the RTK-GNSS positioning results, in this study, all point clouds of the RTK-Fix, RTK-Float, and dead reckoning Fix solutions were included in the processing. Eight sections were selected visually considering the geological structure of the Kanda-gawa River, as shown in Figure 17.

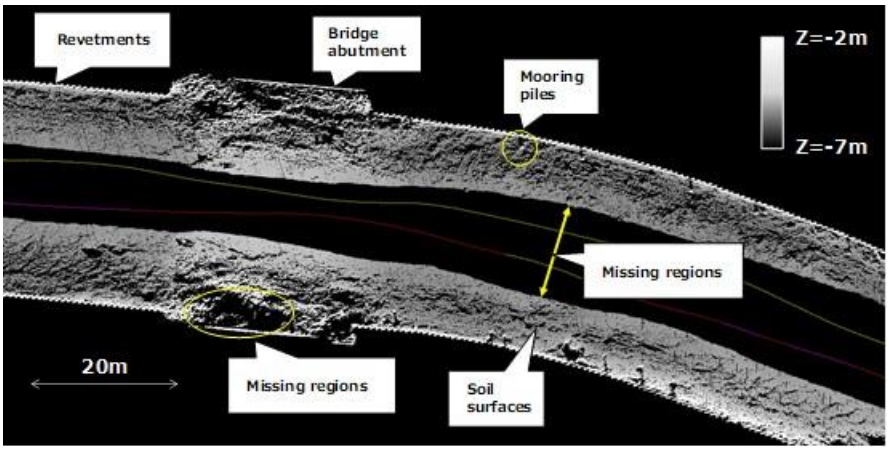

Figure 16: DSM generated from underwater point clouds.

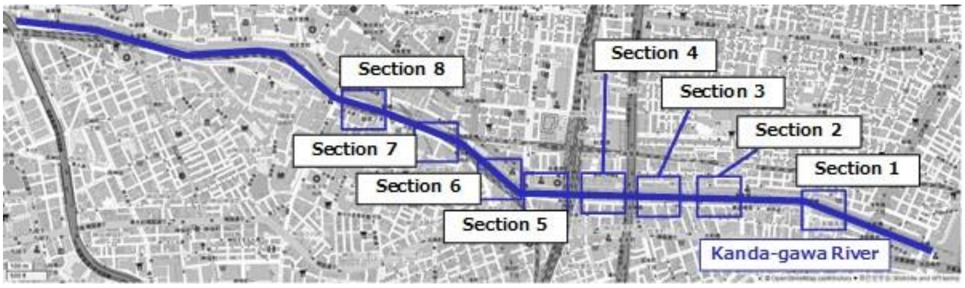

Figure 17: Selected sections for underwater point cloud processing.

Results and Discussion

The processing time for GNSS/LiDAR-SLAM was approximately 129 seconds (0.1095 [s/frame]), and that for segmentation was about 16 seconds (0.0138 [sec/frame]) (all processing environment: CPU: Intel Core i7-1255U 1.70 GHz, RAM: 32 GBytes) for the generated point clouds as shown in Figure 18.

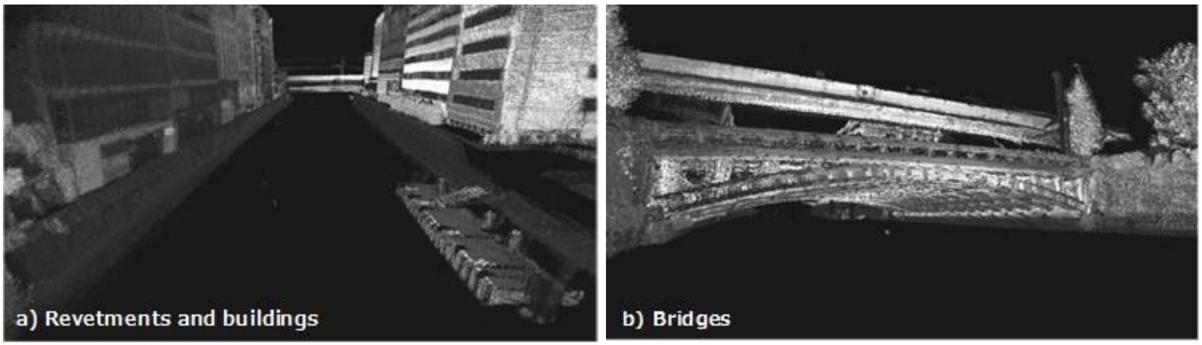

Figure 18: GNSS/LiDAR-SLAM processing results.

Figure 19 and Figure 20 show the segmentation results of the streaming point clouds. Figure 19 and Figure 20 also indicate that revetments, buildings, and bridges of point clouds were managed with layers.

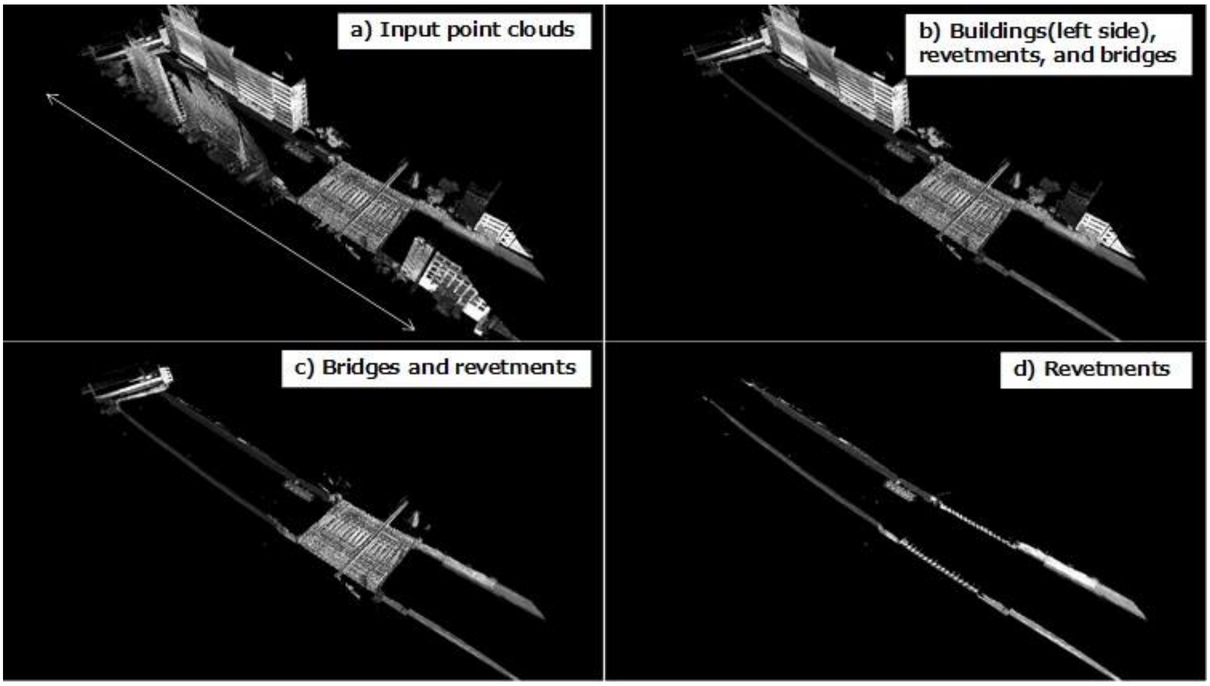
a) Input point clouds
b) Buildings(left side), revetments, and bridges
c) Bridges and revetments
d) Revetments

Figure 19: Segmentation results (bridges, revetments, and buildings).

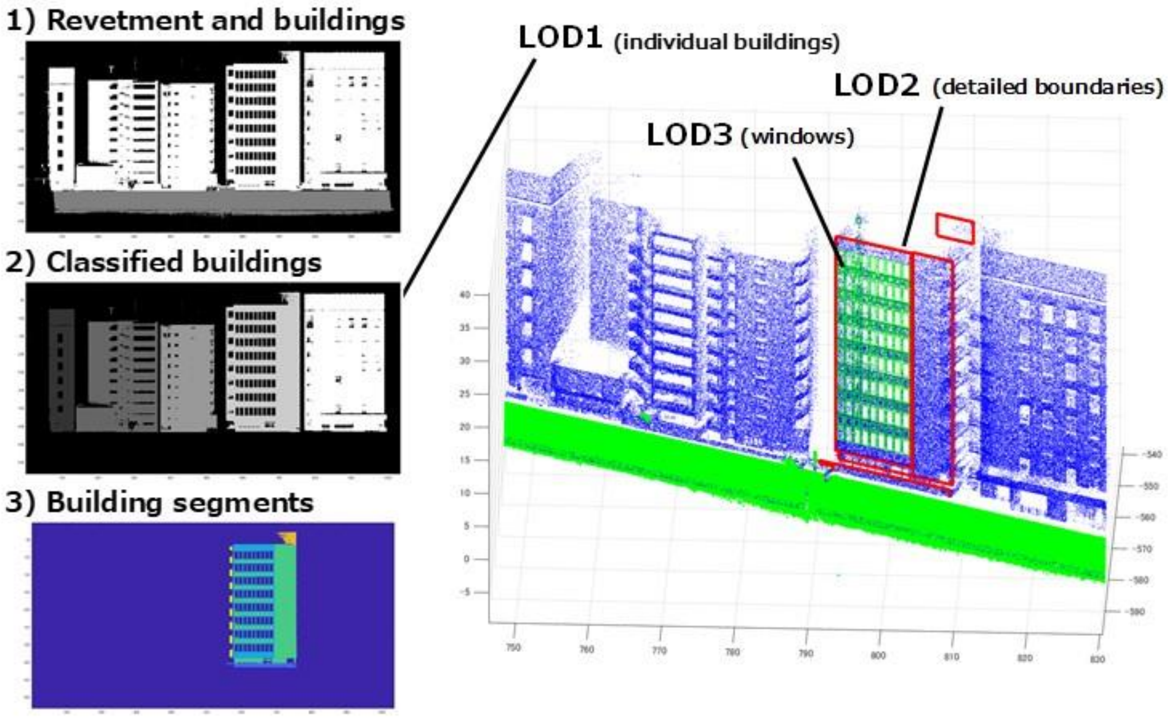
1) Revetment and buildings
2) Classified buildings
3) Building segments
LOD1 (individual buildings)
LOD2 (detailed boundaries)
LOD3 (windows)

Figure 20: Segmentation results (buildings).

Although the accumulated error in LiDAR-SLAM processing without GNSS rectification was 4.838 m for 1,180 frames (Table 2), the accumulated error in LiDAR-SLAM processing was 0.0041[m] per frame. We can mention that our methodology rectified the accumulated errors using GNSS within the CLAS positioning accuracy and our methodology achieved higher accuracy than the LiDAR ranging accuracy of 0.03[m] (catalog value). Moreover, the GNSS/LiDAR-SLAM can estimate positions where GNSS positioning is not available with sufficient accuracy. We confirmed that the GNSS/LiDAR-SLAM can process in 0.1095 [s/frame]. We also confirmed that the segmentation (including the classification of bridges, revetments, and buildings) and the classification of each building can be processed in 0.0138 [s/frame]. Thus, we confirmed that our proposed methodology can process simultaneous 3D mapping with point cloud acquisition and annotation generation with simultaneous processing with LiDAR-SLAM.

Table 2: Pose estimation error evaluation.

Section ID	Elapsed time [sec]	Positioning solution	Start frame	End frame	Baseline length [m]	SLAM error[m]	SLAM error [m] / Baseline length [m]	Notes
--	15.8	FIX	1	159	35.937	2.026	0.056	79 adjusted data
1	21.6	No FIX	161	217	13.262	0.041	0.003	
--	23.8	FIX	219	239	5.182	0.205	0.040	11 adjusted data
2	103.0	No FIX	241	1031	182.110	0.662	0.004	
--	109.8	FIX	1033	1099	15.170	1.159	0.076	34 adjusted data
3	116.0	No FIX	1159	1161	14.272	0.517	0.036	
--	118.0	FIX	1163	1181	4.475	0.228	0.051	10 adjusted data
				Total	270.408	4.838		
						SLAM error (overall) :	0.0041[m/frame]	

Figure 21 shows a part of the underwater point cloud segmentation results. The processing time for each of the eight sections (CPU: Intel Core i7-1255U 1.70 GHz) is shown in Table 3. We proposed a methodology for classifying point clouds acquired by multi-beam scanning sonar into two categories based on knowledge-based classification: river structures such as revetments and landforms such as sediments. In terms of classification performance, we confirmed the usefulness of a simple reproduction of multi-beam sonar scanning from random point clouds using GNSS positioning data.

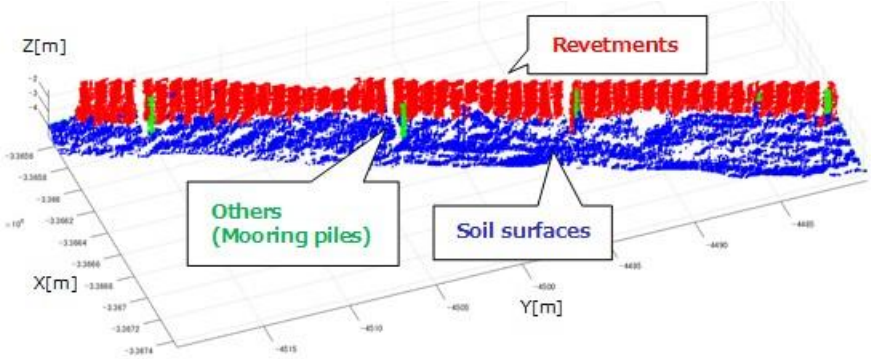

Figure 21: Segmentation results (underwater).

Table 3: Processing time for underwater point clouds.

Section ID	1	2	3	4	5	6	7	8
The number of GNSS epochs	276	349	344	334	358	404	355	333
The number of point clouds [pts]	49696	50721	56670	55347	66268	69235	71748	67307
Processing time [sec]	0.29	0.31	1.81	0.35	0.58	3.49	0.26	0.65
File loading [sec]	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Revetment extraction [sec]	0.19	0.19	0.18	0.26	0.23	0.26	0.22	0.26
Classification of features [sec]	-0.12	-0.13	1.42	-0.20	0.09	2.94	-0.21	0.10

Conclusion

In this research, we focused on the integration of image-based and 3D-based approaches embedded in SLAM and mobile mapping. We proposed a methodology to improve the performance of streaming point cloud processing based on image-based and 3D-based point cloud segmentation for 3D mapping of urban river environments. We also developed a methodology of streaming point cloud segmentation embedded in GNSS/LiDAR-SLAM and multi-beam scanning. Through our experiments using a water-borne mobile mapping system, we confirmed that our methodology can improve the scalability of point cloud processing and achieve high-speed processing and precise classification as well as conventional image-based and 3D-based point cloud segmentation approaches.

References

Yuan Wang, Tianyue Shi, Peng Yun, Lei Tai, Ming Liu, (2018). PointSeg: Real-Time Semantic Segmentation Based on 3D LiDAR Point Cloud. *arXiv:1807.06288, 7 pages.*

Andres Milioto, Ignacio Vizzo, Jens Behley, Cyrill Stachniss, (2019). RangeNet ++: Fast and Accurate LiDAR Semantic Segmentation. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4213-4220.*

Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, Masayoshi Tomizuka, (2020). SqueezeSegV3: Spatially-adaptive Convolution for Efficient Point-cloud Segmentation, *European Conference on Computer Vision, 21 pages.*

Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, Kurt Keutzer, (2017). SqueezeNet: AlexNet-level Accuracy with 50x Fewer Parameters and <0.5MB Model Size, *Proc. International Conference on Learning Representation, 13 pages.*

Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, (2012). ImageNet Classification with Deep Convolutional Neural Networks, *Neural Information Processing Systems, 141, pp.1097-1105.*

Andreas Geiger, Philip Lenz, Raquel Urtasun, (2012). Are we ready for autonomous driving? The KITTI Vision Benchmark Suite, *Proc. CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.3354- 3361.*

Charles R. Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, Leonidas J. Guibas, (2016). Volumetric and Multi-View CNNs for Object Classification on 3D Data, *In Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, pp.5648-5656.*

Charles R. Qi, Hao Su, Kaichun Mo, Leonidas J. Guibas, (2016). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation, *arXiv:1612.00593, 19 pages.*

Charles R Qi, Li Yi, Hao Su, Leonidas J. Guibas, (2017). PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space, *ArXiv:1706.02413, 14 pages.*

Masafumi Nakagawa, Naoto Kimura, Takeshi Komori, Nobuaki Kubo, Etsuro Shimizu, (2023). Point Cloud Acquisition based on GNSS/SLAM with an Autonomous Boat in Urban River Environments, *Asian Conference on Remote Sensing 2023, 10 pages.*

Yang Chen, Gérard Medioni, (1992). Object Modelling by Registration of Multiple Range Images, *Image Vision, Computing. Butterworth-Heinemann. Vol. 10, Issue 3, pp. 145-155.*

Dai Mimura, Katsumasa Miyatake, Yukihiro Kubo, Sueo Sugimoto, (2020). Positioning Accuracy of Single Frequency GNSS PPP by CLAS Comparing with MADOCA Products, *Proceedings of the ISCIE International Symposium on Stochastic Systems Theory and its Applications, pp.43-48.*

Huan Ni, Xiangguo Lin, Xiaogang Ning, Jixian Zhang, (2016). Edge Detection and Feature Line Tracing in 3D-Point Clouds by Analyzing Geometric Properties of Neighborhoods, *Remote Sensing, 8(9):710.*