A Systematic Study on Instance-Level Bridge Point Cloud Segmentation under Cross-Type and Synthetic–Real Training Conditions

Jeong Kyu Lee[1], Ui Chan Lee[1], Jong Won Ma[1]
[1] School of Civil and Environmental Engineering, Yonsei University
smile6569@yonsei.ac.kr, air0223@yonsei.ac.kr, jongwon.ma@yonsei.ac.kr

Abstract -

Bridge point cloud instance segmentation is essential for digital twin construction and automated bridge management. However, deep learning–based approaches are limited by the scarcity of instance-level annotated real-world data and the domain gap between synthetic and real point clouds. In practical end-to-end Scan-to-Bridge Information Modeling(BrIM) pipelines, segmentation models must operate directly on raw point clouds without assuming prior bridge-specific domain knowledge or relying on domain-driven preprocessing.

To address these challenges, synthetic bridge point clouds for slab and PSC girder bridges were generated using physics-based LiDAR simulation, with a unified component definition applied to both synthetic and real data to construct consistent instance-level annotations. Background elements, including terrain and vegetation, were incorporated into the synthetic scenes to reduce the gap between realistic conditions. Experimental results show that synthetic-only training yields the lowest performance, followed by real-only training, while the best performance is achieved when synthetic and real data are jointly used.

For the Japan dataset, mIoU and mAP improved from 0.253 and 0.131 to 0.531 and 0.638, respectively. For the Cambridge dataset, although mIoU slightly decreased from 0.556 to 0.549, mAP increased from 0.617 to 0.688, indicating improved instance-level performance.

Instance-level segmentation is particularly important in bridge applications, as maintenance and long-term monitoring require component-wise analysis and consistent tracking of *maintenance-relevant structural elements or structural systems*. Given the limited visibility of abutments in real-world and their frequent exclusion in prior studies, we further evaluated performance under scenarios with deck and background components to reflect realistic end-to-end deployment conditions.

Keywords –

Scan-to-BrIM; Instance segmentation; Synthetic point clouds;

1 Introduction

Bridges are critical infrastructure that connects local communities and supports economic activities, making efficient management and maintenance essential. However, conventional bridge management heavily relies on visual inspections and manual operations, which are time-consuming, labor-intensive, and costly [1]. As a result, digital transformation of bridge management has gained increasing attention, with digital twins and Scan-to-Bridge Information Modeling (Scan-to-BrIM) emerging as promising solutions [2].

Within Scan-to-BrIM pipelines, bridge components must be identified from point clouds acquired via laser scanning or photogrammetry and subsequently reconstructed as structured geometric models. Point cloud segmentation therefore plays a central role in enabling downstream modeling and analysis tasks [3]. In particular, instance-level segmentation is required for practical bridge applications, as maintenance, inspection, and long-term monitoring depend on component-wise analysis and consistent tracking of maintenance-relevant structural elements or systems. For this purpose, recent deep learning–based approaches have shown clear advantages over traditional rule-based methods in handling complex bridge geometries and diverse structural configurations [4].

However, the effectiveness of deep learning–based bridge segmentation is fundamentally constrained by the availability and quality of training data. Generating accurate instance-level annotations for real-world bridge point clouds requires substantial domain expertise, manual effort, and time, while large-scale acquisition of real bridge data is often limited by logistical and financial constraints. To address these challenges, many studies have explored training strategies that combine synthetic and real bridge point cloud data, leveraging synthetic data to mitigate labeling costs and data scarcity while incorporating real data to improve realism and generalization.

The synthetic and real data generation pipeline adopted in this study builds upon established workflows in prior literature, while incorporating methodological refinements tailored to bridge point cloud instance segmentation.

Despite recent progress, many existing studies still rely on simplified problem formulations for bridge point

cloud instance segmentation, such as excluding certain structural components (e.g., abutments), merging partially visible elements into dominant instances like decks, or conducting segmentation on pre-filtered point clouds without background points. These simplifications embed strong, often implicit, prior assumptions about bridge geometry and data preprocessing, and therefore depart from realistic end-to-end Scan-to-BrIM workflows, where models must directly process raw, cluttered point clouds acquired from field measurements.

To address this gap, we propose a stepwise experimental framework that explicitly controls and progressively relaxes these assumptions. Rather than treating instance segmentation under a single idealized setting, our framework systematically evaluates model behavior across multiple levels of realism by varying the presence of background points and structurally ambiguous components. This design enables a controlled analysis of how training strategies generalize as the problem formulation transitions from idealized to practically deployable conditions.

2 Related work

2.1 Semantic Segmentation of Bridge Point Clouds

Bridge point cloud segmentation is a fundamental technology for Scan-to-BIM, automated bridge inspection, and digital twin construction, as it enables the identification and separation of structural components from raw 3D point clouds at both semantic and instance levels. Accurate segmentation is a prerequisite for downstream tasks such as structural assessment, deformation analysis, and maintenance planning.

Early studies on bridge point cloud segmentation primarily relied on handcrafted geometric features and rule-based heuristics, including curvature, normal vectors, region growing, and threshold-based clustering [5]. While these approaches achieved reasonable performance in controlled environments, they exhibited strong dependence on manually tuned parameters and limited robustness when applied to complex real-world scenes with occlusions, noise, and varying point densities.

With the rapid advancement of deep learning, data-driven approaches have become dominant in point cloud segmentation. Methods based on PointNet, PointNet++ [6-7], and sparse convolutional networks significantly improved segmentation accuracy by learning discriminative features directly from point cloud data. More recently, transformer-based instance segmentation frameworks have attracted increasing attention due to their ability to model long-range dependencies and global contextual relationships, leading to improved performance in complex and cluttered scenes [8].

Despite these advances, applying deep learning–based segmentation methods to bridge point clouds remains challenging. Bridge datasets often suffer from severe class imbalance, where small components such as abutments or parapets contain significantly fewer points than major components like decks or girders. In addition, acquiring large-scale, labeled bridge point clouds is costly and time-consuming, limiting the availability and diversity of training data [9]. These challenges restrict the generalization capability of existing models when deployed in real-world bridge environments.

2.2 Synthetic Bridge Point Cloud Generation

Deep learning–based bridge point cloud segmentation requires large amounts of labelled data. However, constructing component-level ground-truth annotations for real bridges is extremely costly and labor-intensive. To address this limitation, extensive research has explored the use of synthetic bridge point cloud data.

Early synthetic data generation methods typically sampled point clouds directly from 3D CAD or mesh-based bridge models [10]. Although these approaches enabled efficient data generation and precise labelling, they failed to capture critical characteristics of real-world point clouds, such as sensor noise, non-uniform sampling density, occlusion effects, and background clutter. Consequently, models trained on such data often exhibited limited generalization to real-world scenarios.

To improve realism, subsequent studies adopted physics-based LiDAR simulation or UAV–SfM pipelines to generate synthetic point clouds that more closely resemble real sensor measurements [11-12]. These methods incorporated factors such as scanning geometry, measurement noise, and partial occlusion, leading to improved data fidelity. Nevertheless, many existing approaches remain limited to a small number of bridge types or simplified environmental settings, often neglecting surrounding terrain, vegetation, and background structures.

As a result, a persistent domain gap between synthetic and real bridge point clouds has been widely reported and recognized as a major factor degrading model performance in real-world applications [13]. This limitation has motivated recent research to investigate training strategies that combine synthetic and real data or progressively adapt models across domains. However, systematic comparisons of different training data compositions under controlled experimental settings remain limited, particularly in the context of instance-level bridge segmentation.

3 Methodology

To ground our methodology in existing research

practice, this study first examines six representative prior works that utilize the same real-world bridge point cloud dataset [14-15-16-17-18-19]. From a training-strategy perspective, most previous studies focus on improving segmentation performance by combining synthetic and real data, applying data augmentation to real data, or leveraging pseudo-labeling techniques to exploit unlabeled real-world point clouds [15-16-17]. Other approaches emphasize the generation of synthetic bridge models designed to closely resemble real bridges in order to enhance training effectiveness [8].

Meanwhile, several studies effectively simplify the segmentation task by constraining scene composition or component definitions at the dataset level. For example, in Xia et al. [14], only pier, slab, and background classes are considered, resulting in a highly simplified scene configuration. In the Yang et al. [15] and Yang et al. [17], background and abutment components are excluded altogether, meaning that the input point clouds do not represent raw scanning conditions. Although the Xu et al. [16] partially includes background and abutments, most bridges are truncated around the mid-span region, thereby avoiding complex boundary conditions near abutments.

In the Lin et al. [18], loss functions are strongly tuned based on bridge-specific structural priors, introducing explicit domain assumptions into the learning process. Furthermore, in the Chen et al. [19], abutments and piers are merged into a single category, further reducing component-level granularity and simplifying the segmentation task.

In contrast, practical end-to-end Scan-to-BrIM pipelines operate directly on raw point clouds acquired from field measurements, where background clutter and partially occluded components are inherently present and cannot be reliably pre-filtered. To reflect this practical constraint, this study explicitly distinguishes three training scenarios—synthetic data only, real data only, and combined synthetic–real data—and conducts a comprehensive, stepwise performance comparison using raw point cloud inputs under a unified model architecture and evaluation protocol. Through this analysis, we systematically investigate how training data composition influences bridge point cloud instance segmentation performance in realistic, domain-agnostic settings. A summary of performance comparisons reported in previous studies is presented in Table 1.

Table 1. Segmentation of performance comparisons reported in prior studies

Dataset	mIoU (%)	mAP (%)	OA (%)	Acc (%)
Xia et al. [14]	94.72	-	-	-
Yang et al. [15]	91.84	-	95.56	96.74
Xu et al. [16]	41.4	63.3		
Yang et al. [17]	89.83	-	95.34	94.31
Lin et al. [18]	74.42	-	87.67	86.65
Chen et al. [19]	81.0	-	-	92

3.1 Datasets

To evaluate the performance of the proposed method, two real-world bridge point cloud datasets from Japan and Cambridge were used in this study. By leveraging datasets constructed in different countries, environments, and scanning conditions, we aim to comprehensively assess the generalization capability and practical applicability of the proposed framework. Examples of the real bridge point clouds are shown in Fig. 1.

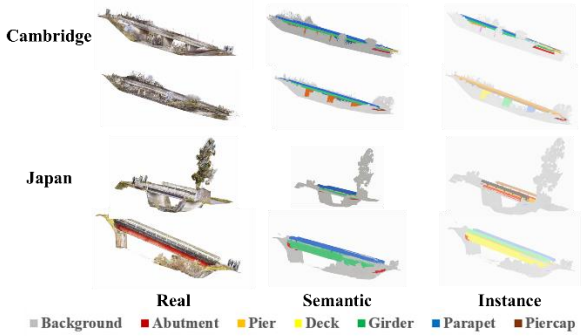

Figure 1. Representative bridge point clouds from the Japan and Cambridge datasets.

3.1.1 Japan Dataset

Japan Dataset is a proprietary dataset collected in December 2023 from nine reinforced concrete bridges located in rural areas of Japan using a Matterport Pro3 terrestrial laser scanner[20]. The scanner provides a 360° horizontal field of view and a 295° vertical field of view, with a measurement accuracy of ±20 mm at a distance of 10 m and a maximum acquisition rate of 100,000 points per second.

All points were manually annotated into five structural component classes: Abutment, Girder, Deck, Parapet, and Background. For abutment components, parts of the frontal regions were reclassified as Background to account for geometric complexity and severe class imbalance.

For this dataset, following the original bridge IDs provided in the dataset, Bridges 2, 4, 5, 6, 8, and 9 were used for training, while Bridges 1, 3, and 8 were used for testing.

3.1.2 Cambridge Dataset

Cambridge Dataset is a publicly available benchmark dataset, consisting of highway bridge point clouds collected in the Cambridgeshire region of the United Kingdom using a FARO Focus 3D X330 terrestrial laser

scanner[21]. To ensure consistency number in Japan Dataset, a total of nine bridges (Bridges 2–10) were used, excluding Bridge 1 from the original set of ten bridges.

The bridges have total lengths ranging from 55 to 91 m, span lengths of 13 to 35 m, and heights between 5 and 8 m. Structurally, they include multiple piers and horizontally curved deck geometries. All points were annotated into six structural component classes: Abutment, Pier, Girder, Deck, Parapet, and Background.

For this dataset, following the original bridge IDs provided in the dataset. Bridges 3, 4, 5, 7, 8, and 10 were used for training, while Bridges 2, 6, and 9 were used for testing.

3.2 Synthetic and Real Data Generation Pipeline

This study proposes an integrated data generation and annotation pipeline that combines synthetic and real-world data to construct training datasets for 3D instance segmentation of bridge components. An overview of the proposed pipeline is illustrated in Fig. 2.

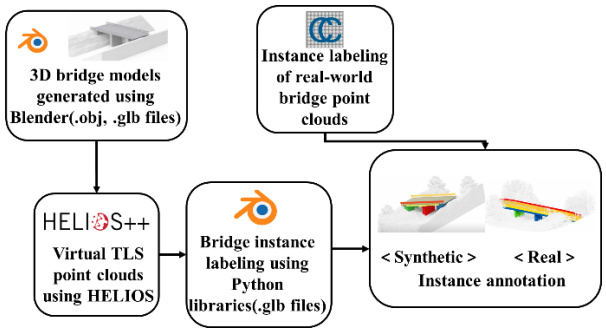

Figure 2. Overview of the proposed data generation and annotation pipeline for bridge component instance segmentation.

Bridge models of slab bridges and PSC girder bridges were first collected from the online repository www.bigdata-eng.com in IFC format and subsequently converted into OBJ files using Blender. Additional structural elements such as piers and surrounding terrain were manually modeled to construct complete 3D bridge scenes in Blender. The synthetic bridge dataset consists of 48 slab bridges and 9 PSC girder bridges, reflecting structural diversity in terms of bridge length (10–45 m for slab bridges and 30–50 m for girder bridges) and pier geometry (rectangular, square, and cylindrical). In addition, terrain-inclusive versions were generated for all bridges to account for realistic topographic conditions. Representative examples of slab and girder bridges are shown in Fig. 3. Virtual terrestrial laser scanning (TLS) was then performed on the generated 3D bridge models using HELIOS++ to produce synthetic bridge point clouds.

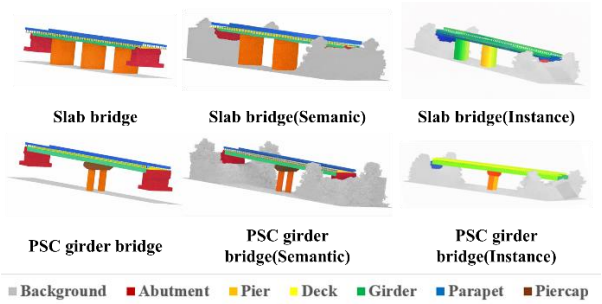

Figure 3. Representative synthetic bridge models used in this study: slab bridges and PSC girder bridges.

To emulate real-world TLS acquisition conditions, scanner placement, field of view, ray distribution, and range attenuation effects were explicitly modeled, and scanner configurations were defined based on the specifications of the RIEGL VZ-400. In particular, TLS scanners were automatically deployed at regular intervals on multiple horizontal planes derived from the bridge bounding box, including planes located above and below the bridge deck in the XY plane and an additional plane positioned 2 m below the structural center. This strategy ensured sufficient coverage of bridge geometries from multiple viewpoints. Scanners that were placed inside the bridge mesh were automatically excluded to prevent unrealistic scanning artifacts. As a result, synthetic point clouds with spatial distributions and point densities comparable to those obtained from real TLS surveys were efficiently generated. The scanner spacing was set to 10 m or 15 m depending on whether background elements were included. Terrain and vegetation were automatically placed to replicate realistic bridge environments. The synthetic dataset was ultimately divided into two variants, with and without vegetation and surrounding hills, while ground surfaces were included in both variants. Detailed HELIOS++ scanner settings are summarized in Table 2.

Following point cloud generation, a Python-based post-processing pipeline was applied to automatically assign component-level instance labels to the synthetic point clouds. Structural component information defined during the Blender modeling stage was preserved in GLB files, and a nearest-neighbor matching process between mesh surfaces and point clouds was used to consistently assign semantic labels and instance IDs to individual points. In contrast, real-world bridge TLS point clouds were manually annotated using CloudCompare.

Table 2. HELIOS++ TLS simulation settings used for synthetic bridge point cloud generation.

Parameter	Value
Pulse frequency	100,000 Hz
Scan frequency	120 Hz
Scan angle	100°

Head rotation speed	10°/s
Vertical FOV	−40° to 60°
Horizontal FOV	−180° to 180°
Trajectory interval	1.0 s

For all point cloud data, RGB values were unified to grayscale to minimize the influence of color information. A voxel down-sampling with a resolution of 0.05 m was applied, followed by spatial cropping into 10 m-sized volumes using a sliding-window strategy. Here, the overlap ratio is defined as the percentage of spatial overlap between adjacent cropped volumes. Overlap ratios of 0%, 12.5%, and 25% were applied, resulting in approximately 100k–300k points per scene. The final dataset includes both synthetic and real bridge point clouds and shares consistent class definitions and instance labeling schemes, enabling direct use for training and evaluating 3D point cloud instance segmentation models.

3.3 Experimental Setting / Evaluation Metrics

All experiments were conducted on a workstation equipped with an AMD Ryzen 7 5800X CPU, an NVIDIA RTX 3090 GPU with 24 GB VRAM, and 32 GB of system memory. The experimental environment was configured on Ubuntu 20.04, using Python 3.8 and CUDA 11.3.

In this study, Relation3D[22] was adopted as the baseline instance segmentation model, and all models were trained for 200 epochs. The training configuration of Relation3D followed the official settings for the ScanNetV2 dataset, except that the batch size was set to 10 and the number of data loader workers was set to 5. In addition, different overlap ratios (0%, 12.5%, and 25%) were applied during data preparation.

Intersection over Union (IoU) measures the degree of overlap between the predicted and ground-truth regions for a given class, indicating how accurately the predicted area matches the reference annotation. Based on IoU, Average Precision (AP) evaluates instance-level segmentation performance by summarizing the precision–recall relationship obtained from IoU-based matching between predicted and ground-truth instances. Mean Intersection over Union (mIoU) is then computed by averaging the IoU values across all classes, providing an overall measure of semantic segmentation accuracy. Finally, mean Average Precision (mAP) is obtained by averaging the AP values over all classes, reflecting the overall instance segmentation performance.

4 Experimental Results

The Japan and Cambridge datasets were used as training sets, while the synthetic dataset consisted of 48 slab bridges and 9 PSC girder bridges. The synthetic dataset was further divided into two variants depending on the presence of surrounding Background. Overlap ratios of 0%, 12.5%, and 25% were applied during data preparation to represent low, moderate, and high levels of spatial redundancy between adjacent subspaces. Specifically, 0% corresponds to a non-overlapping baseline, while 12.5% and 25% introduce progressively increasing overlap within a practical range before excessive redundancy occurs. This design enables a systematic analysis of the trade-off between computational cost and segmentation accuracy. For evaluation, the respective test sets of the Japan and Cambridge datasets were used.

From this perspective, this study compared three training scenarios based on instance segmentation that include vegetation and abutments: training with synthetic data only, training with real data only, and training with a combination of synthetic and real data. Experimental results show that training with synthetic data alone consistently yielded the lowest performance. For the Japan dataset (Table 3), under background-included settings, AP values for overlap ratios of 0%, 12.5%, and 25% were 0.057, 0.088, and 0.078, respectively. When background points were excluded, AP further decreased to 0.045, 0.040, and 0.032 under the same overlap conditions. In contrast, mIoU values were relatively higher under background-excluded settings, reaching 0.291, 0.279, and 0.242, respectively.

Table 3. Accuracies on the Japan dataset for synthetic-only training across overlap ratios (with/without background)

Background\Overlap	0%	12.5%	25%
	mIoU / mAP	mIoU / mAP	mIoU / mAP
Background(0)	0.257 /0.057	0.206 /0.088	0.210 /0.078
Background(x)	0.291 /0.045	0.279 /0.040	0.242 /0.032

Similar trends were observed in the Cambridge dataset (Table 4). When background points were included, AP values at overlap ratios of 0%, 12.5%, and 25% were 0.266, 0.279, and 0.266, respectively, whereas background-excluded settings resulted in mAP values of 0.213, 0.207, and 0.231 under the same conditions. The mIoU values were similar across all settings. Since real bridge environments require the separation of individual components under complex scenes where abutments, ground surfaces, terrain, and surrounding structures are mixed, training with realistic scenes that include

background elements is essential for instance segmentation.

Table 4. Accuracies on the Cambridge dataset for synthetic-only training across overlap ratios (with/without background)

Background\Overlap	0%	12.5%	25%
	mIoU / mAP	mIoU / mAP	mIoU / mAP
Background(0)	0.300 /0.266	0.291 /0.279	0.294 /0.266
Background(x)	0.298 /0.213	0.306 /0.207	0.299 /0.231

Training with real data only led to a clear performance improvement. However, the optimal overlap ratio varied across datasets. For the Japan dataset (Table 5), the highest performance was achieved at an overlap ratio of 12.5%, with an mIoU of 0.253 and an mAP of 0.131. In contrast, the Cambridge dataset (Table 6) achieved its best performance at an overlap ratio of 25%, yielding an mIoU of 0.556 and an mAP of 0.617. This difference is closely related to bridge scale and the crop-based data construction strategy. For structurally smaller Japanese bridges, increasing the overlap ratio can limit the model's ability to sufficiently learn component arrangements and instance-level relationships. In contrast, the longer and more complex Cambridge bridges provide richer spatial contexts, which are more favorable for learning instance boundaries.

Table 5. Accuracies on the Japan dataset under real-data-only training with different overlap ratios.

Overlap	mIoU	mAP
0%	0.075	0.055
12.5%	0.253	0.131
25%	0.181	0.112

Table 6. Accuracies on the Cambridge dataset under real-data-only training with different overlap ratios.

Overlap	mIoU	mAP
0%	0.462	0.485
12.5%	0.415	0.382
25%	0.556	0.617

The most significant performance improvements were observed when synthetic and real data were combined for training. For the Cambridge dataset, applying a 25% overlap ratio and a background-excluded setting for synthetic data resulted in an mIoU of 0.549 and an AP of 0.688. Similarly, for the Japan dataset, an overlap ratio of 12.5% with background-excluded synthetic data achieved an mIoU of 0.531 and an AP of 0.638 (Table 7). In both datasets, better performance was achieved when the synthetic data excluded background elements,

Table 7. Optimal overlap ratio selection for synthetic–real combined training and corresponding performance.

Background \Name(Overlap)	Cambridge (25%)	Japan (12.5%)
	mIoU / mAP	mIoU / mAP
Background(0)	0.530 / 0.668	0.488 / 0.543
Background(x)	0.549 / 0.688	0.531 / 0.638

which can be attributed to the fact that background characteristics were learned primarily from real data. Detailed performance results for each component are presented in Figs. 4 and 5.

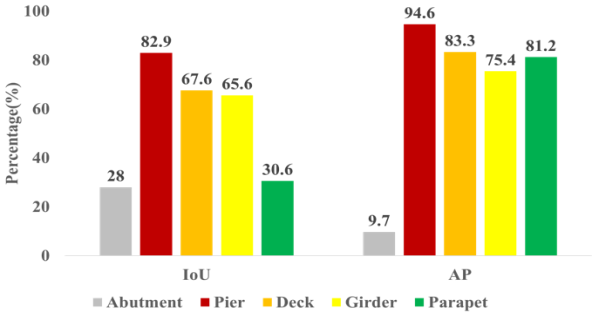

Figure 4. Component-wise instance segmentation performance on the Japan dataset.

These results indicate that synthetic data contributes beyond merely increasing the amount of training data, effectively supporting instance segmentation learning for components that are difficult to sufficiently observe in real TLS data. In particular, abutments, which are adjacent to the ground and structurally complex, often exhibit sparse point distributions and unclear boundaries with background elements, making instance segmentation challenging. Synthetic data provides clear geometric structures and instance boundaries for such components, enabling the model to learn object-level segmentation capabilities required in real end-to-end maintenance pipelines.

Furthermore, when the entire synthetic dataset was combined with real bridge data collected from different regions for training (Table 8), the Japan dataset showed improvements in mAP and mIoU from 0.131 and 0.253 to 0.238 and 0.279, respectively, compared to training with real data only. Similarly, for the Cambridge dataset, mAP and mIoU increased from 0.617 and 0.556 to 0.734 and 0.605. These results outperform the case where training was conducted using real data from the same regional domain only (mAP 0.688, mIoU 0.549), demonstrating that combining background-free synthetic data with multi-domain real data is also effective for improving domain generalization.

Table 8. Performance Comparison of Cross-Domain Real-Data Training with Fixed Synthetic Data

Overlap/Configuation	mIoU	mAP	Abutment	Pier	Deck	Girder	Parapet
			IoU/AP	IoU/AP	IoU/AP	IoU/AP	IoU/AP
Japan(12.5%)	0.279	0.238	0.069/0.002	- / -	0.374/0.138	0.460/0.446	0.214/0.365
Cambridge(25%)	0.605	0.734	0.551/0.223	0.726/0.964	0.659 / 0.834	0.689/0.772	0.402/0.876

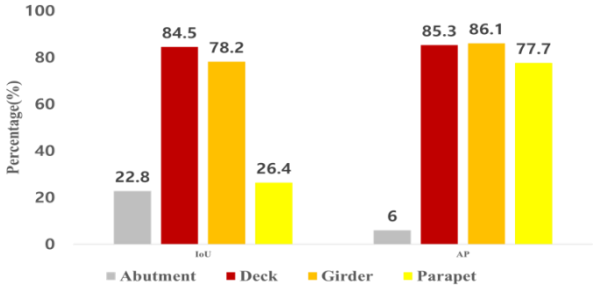

Figure 5. Component-wise instance segmentation performance on the Cambridge dataset.

Instance segmentation results for Bridge 8 in the Japan dataset and Bridge 6 in the Cambridge dataset are visualized in Fig. 6. In the visualization, background points are shown in gray, incorrectly predicted regions are highlighted in red, and correctly segmented instances are distinguished by different colors.

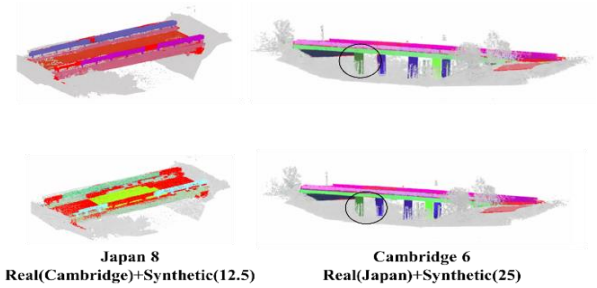

Figure 6. Instance Segmentation Results for Bridge 8 (Japan) and Bridge 6 (Cambridge)

5 Conclusion and Limitation

This study systematically analyzed the performance variations of 3D point cloud–based component-level instance segmentation for bridge maintenance and digital twin construction according to different training data compositions. The fundamental motivation for performing bridge instance segmentation is that, even when components belong to the same category, each component is treated as an independent management unit at an appropriate management level from a maintenance perspective. Multiple piers, parapets, and abutments, which are spatially separable structural units, may be classified into the same semantic class. However, in actual maintenance processes, each component exhibits different damage states, deterioration patterns, inspection histories, and repair records. Therefore, point-wise semantic classification alone is insufficient to satisfy these management requirements, and instance segmentation that identifies individual components as object-level entities is a necessary step in end-to-end Scan-to-BrIM and bridge digital twin pipelines.

Nevertheless, this study has several limitations. First, the experimental scope is limited to slab bridges and PSC girder bridges. Second, although the synthetic dataset reflects variations in component geometry, it is still based on a limited set of component configurations and structural details. Third, some components, such as abutments, still exhibit relatively low performance due to complex geometries and severe class imbalance.

In summary, this study clearly defines bridge instance segmentation as an essential object-level analysis step at the system-management level from a maintenance perspective and demonstrates the performance of training with scenes that include both bridge components and surrounding elements such as background and abutments. The results further show that combining synthetic data with multi-domain real data is an effective strategy for robustly separating individual components in realistic bridge scenes, providing practical guidelines for 3D point cloud–based Scan-to-BrIM and bridge digital twin construction.

Although this study includes both slab-type and girder-type bridges from publicly available datasets in Korea to incorporate structural diversity, the datasets used in this study remain limited. While the effectiveness of the proposed framework is demonstrated, further validation on a broader range of bridge types and real-world datasets is required. In addition, a more comprehensive exploration of training strategies—such as synthetic–real mixing ratios, pretraining versus joint training, and evaluation across multiple models—remains as future work to further enhance generalizability and robustness.

References

[1] J. O. Sobanjo, “Cost Estimating Under Uncertainty: Issues in Bridge Management” *Transportation Research Circular* 498 (2000).

[2] R. F. S. Cerquin, M. Alarcon, and R. M. Delgadillo, “BrIM and Digital Twin Integration for Structural Health Monitoring and Analysis of the Villena Rey Bridge via Laser Scanning,” *Appl. Sci.*, vol. 15, no. 21, Nov. 2025, doi: 10.3390/app152111741.

[3] M. Castellani, M. A., G.-M. E., A. F., and U. F., “UAV photogrammetry and laser scanning of

bridges: a new methodology and its application to a case study," *Procedia Struct. Integr.*, vol. 62, pp. 193–200, Jan. 2024, doi: 10.1016/j.prostr.2024.09.033.

[4] G.-Q. Zhang, B. Wang, J. Li, and Y.-L. Xu, "The application of deep learning in bridge health monitoring: a literature review," *Adv. Bridge Eng.*, vol. 3, no. 1, p. 22, Dec. 2022, doi: 10.1186/s43251-022-00078-7.

[5] C. Perea, J. Alcala, V. Yepes, F. Gonzalez-Vidosa, and A. Hospitaler, "Design of reinforced concrete bridge frames by heuristic optimization," *Adv. Eng. Softw.*, vol. 39, no. 8, pp. 676–688, Aug. 2008, doi: 10.1016/j.advengsoft.2007.07.007.

[6] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660. Accessed: Jan. 29, 2026. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2017/html/Qi_PointNet_Deep_Learning_CVPR_2017_paper.html

[7] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2017. Accessed: Jan. 29, 2026. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/d8bf84be3800d12f74d8b05e9b89836f-Abstract.html

[8] A. U. Rahman and V. Hoskere, "Instance segmentation of reinforced concrete bridge point clouds with transformers trained exclusively on synthetic data," *Autom. Constr.*, vol. 173, p. 106067, May 2025, doi: 10.1016/j.autcon.2025.106067.

[9] R. Lu, I. Brilakis, and C. R. Middleton, "Detection of Structural Components in Point Clouds of Existing RC Bridges," *Comput.-Aided Civ. Infrastruct. Eng.*, vol. 34, no. 3, pp. 191–212, 2019, doi: 10.1111/mice.12407.

[10] A. R. Colligan, T. T. Robinson, D. C. Nolan, and Y. Hua, "Point cloud dataset creation for machine learning on CAD models," *Comput.-Aided Des. Appl.*, vol. 18, no. 4, pp. 760–771, Jan. 2021, doi: 10.14733/cadaps.2021.760-771.

[11] L. Winiwarter *et al.*, "Virtual laser scanning with HELIOS++: A novel take on ray tracing-based simulation of topographic full-waveform 3D laser scanning," *Remote Sens. Environ.*, vol. 269, p. 112772, Feb. 2022, doi: 10.1016/j.rse.2021.112772.

[12] G. Lenda, N. Borowiec, and U. Marmol, "Study of the Precise Determination of Pipeline Geometries Using UAV Scanning Compared to Terrestrial Scanning, Aerial Scanning and UAV Photogrammetry," *Sensors*, vol. 23, no. 19, Oct. 2023, doi: 10.3390/s23198257.

[13] N. Duc *et al.*, "Mind the Domain Gap: Measuring the Domain Gap Between Real-World and Synthetic Point Clouds for Automated Driving Development," May 23, 2025, *arXiv*: arXiv:2505.17959. doi: 10.48550/arXiv.2505.17959.

[14] T. Xia, J. Yang, and L. Chen, "Automated semantic segmentation of bridge point cloud based on local descriptor and machine learning," *Autom. Constr.*, vol. 133, p. 103992, Jan. 2022, doi: 10.1016/j.autcon.2021.103992.

[15] X. Yang, E. Del Rey Castillo, Y. Zou, and L. Wotherspoon, "Semantic segmentation of bridge point clouds with a synthetic data augmentation strategy and graph-structured deep metric learning," *Autom. Constr.*, vol. 150, p. 104838, Jun. 2023, doi: 10.1016/j.autcon.2023.104838.

[16] J. Xu *et al.*, "Domain-generalizable point cloud instance segmentation of bridge components using class-balanced dynamic thresholding," *Autom. Constr.*, vol. 181, p. 106631, Jan. 2026, doi: 10.1016/j.autcon.2025.106631.

[17] X. Yang, Y. Fu, and J. Kim, "Synthetic-to-Real Domain Adaptation with Virtual Laser Scanning and Self-Training-Based Category-Aware Cuboid Mixing for Semantic Segmentation of Bridge Point Clouds," 2024, *SSRN*. doi: 10.2139/ssrn.4902362.

[18] C. Lin, S. Abe, S. Zheng, X. Li, and P. Chun, "A structure-oriented loss function for automated semantic segmentation of bridge point clouds," *Comput.-Aided Civ. Infrastruct. Eng.*, vol. 40, no. 6, pp. 801–816, Feb. 2025, doi: 10.1111/mice.13422.

[19] Y. Chen, C. Lin, X. Li, S. Kubo, T. Yamane, and P. Chun, "Automated dimension estimation of bridge components using semantic segmentation and geometric fitting of point cloud data," *Eng. Struct.*, vol. 342, p. 120837, Nov. 2025, doi: 10.1016/j.engstruct.2025.120837.

[20] "Point Cloud Dataset of Reinforced Concrete Bridges Captured with Matterport Pro3 Scanner in Japan." figshare, Mar. 08, 2025. doi: 10.6084/m9.figshare.28091453.v2.

[21] "Detection of Structural Components in Point Clouds of Existing RC Bridges." Accessed: Jan. 30, 2026. [Online]. Available: https://zenodo.org/records/1240534

[22] J. Lu and J. Deng, "Relation3D: Enhancing relation modeling for point cloud instance segmentation," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2025, pp. 8889–8899..