Superpoint Transformer–Based Bridge Component Recognition Using UAV-Mounted LiDAR and Synthetic Point Cloud Generation

Yohan Kim[1]; Jeehoon Kim[2]; Juhyeon Kim, Ph.D.[3]; and Hyoungkwan Kim[4]

Abstract: Given the rapid rise in aging bridges, accurate and efficient inspection and maintenance have become increasingly critical. Point cloud data, rich in geometric details, plays a vital role in this process. Identifying semantic information for each bridge component is essential for effective analysis. This paper presents a novel Superpoint Transformer–Based method for automatically identifying bridge components from unmanned aerial vehicle (UAV)-mounted light detection and ranging (LiDAR) and simultaneous localization and mapping (SLAM)-acquired point cloud data. The method consists of two stages: acquisition and registration of point cloud data, followed by three-dimensional (3D) semantic segmentation using the Superpoint Transformer. Additionally, a new strategy simulates a virtual UAV equipped with LiDAR to generate synthetic data, reducing the domain gap between real and synthetic data. This simulation-based approach outperformed traditional sampling methods, significantly enhancing model performance. By fine-tuning the Superpoint Transformer for bridge component recognition, the proposed method achieved a mean intersection over union (mIoU) of 86.125%. This approach offers an effective solution for bridge component recognition, facilitating efficient inspection and maintenance of aging bridge infrastructure. **DOI: 10.1061/JCCEE5.CPENG-6383.**

Introduction

The deterioration of bridges poses serious structural safety and socioeconomic risks. In South Korea, 2020 data reveal that 18.6% (5,926) of 31,806 bridges had surpassed a 30-year lifespan, with projections indicating that this figure will increase to 51.3% by 2030 (CERIK 2023). The growing number of aging bridges raises concerns over safety incidents. Similarly, in the United States, 42% of the 617,000 bridges are older than 50 years, and 7.5% are structurally deficient according to the American Society of Civil Engineers' (ASCE) 2021 infrastructure report card. Historical records show 503 bridge collapses between 1989 and 2000, with the 2007 Minneapolis bridge collapse causing 158 casualties (ASCE 2021). These statistics underscore the pressing need for regular and effective bridge maintenance.

Maintaining aging bridges enhances safety and provides significant long-term economic benefits. For instance, rehabilitating the Ed Koch Queensboro Bridge, built in 1909, cost $1.0615 billion by 2019—just 23%–24% of the $4.2714 billion estimated for reconstruction (NYCDOT 2019). Effective maintenance relies on regular, detailed inspections to assess bridge conditions accurately. These inspections identify risks early, enable preventive measures, lower long-term costs, and preserve bridge integrity.

Traditional inspection systems, reliant on labor-intensive and time-consuming tasks, are increasingly inadequate for managing expanding traffic networks and aging infrastructure (Costin et al. 2019). Building Information Modeling (BIM) is a key technology for modernizing infrastructure management, offering enhanced reliability, sustainability, and safety while reducing maintenance costs and risks (Eastman et al. 2011). BIM's application to bridges, known as Bridge Information Modeling (BrIM), has proven highly effective for bridge management. Interest in as-built BrIM technologies is growing, particularly for digitizing bridges without prior digital models (Shim et al. 2011; Marzouk and Hisham 2012; Rashidi and Karan 2018).

Accurately modeling the current state of a bridge requires precise measurement and digitization of real-world attributes, such as volume and dimensions. To reduce measurement time and ensure accuracy, 3D as-built data acquisition technologies—such as photogrammetry, terrestrial laser scanning (TLS), and simultaneous localization and mapping (SLAM)—are commonly used, typically providing data in point cloud format. Recently, mobile LiDAR and SLAM-based methods, originally developed for autonomous driving, have gained popularity in bridge applications. These methods enable real-time data capture and registration, streamlining the creation of 3D structural models. Mobile light detection and ranging (LiDAR) systems, miniaturized for mounting on unmanned aerial vehicles (UAVs), further enhance efficiency in capturing 3D point cloud data for large structures such as bridges. However, although these technologies offer detailed and accurate geometric data, they lack semantic information (Esfahani et al. 2021; Hajjan and Becerik-Gerber 2010).

Deep learning has recently gained traction in the Architecture, Engineering, and Construction (AEC) field for processing large data volumes. Its performance largely depends on extensive training data (Gao et al. 2021). However, collecting high-quality point cloud data remains a major challenge. Transfer learning offers

[1]Ph.D. Student, School of Civil and Environmental Engineering, Yonsei Univ., Seoul 03722, Republic of Korea. Email: homez815@yonsei.ac.kr

[2]Graduate Research Assistant, Fariborz Maseeh Dept. of Civil, Architectural, and Environmental Engineering, Univ. of Texas at Austin, Austin, TX 78712. Email: jeehoon3649@utexas.edu

[3]Postdoctoral Researcher, School of Civil and Environmental Engineering, Yonsei Univ., Seoul 03722, Republic of Korea. ORCID: https://orcid.org/0009-0006-9441-3480. Email: kah5125@yonsei.ac.kr

[4]Professor, School of Civil and Environmental Engineering, Yonsei Univ., Seoul 03722, Republic of Korea (corresponding author). Email: hyoungkwan@yonsei.ac.kr

Note. This manuscript was submitted on August 14, 2024; approved on January 14, 2025; published online on April 23, 2025. Discussion period open until September 23, 2025; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Computing in Civil Engineering***, © ASCE, ISSN 0887-3801.

a solution by utilizing large-scale point cloud data sets, such as S3DIS (Armeni et al. 2016), SemanticKITTI (Behley et al. 2019), Semantic3D (Hackel et al. 2017), and DALES (Varney et al. 2020). This technique transfers knowledge from a model pretrained on a source domain to a target domain. In the AEC domain, in which data collection is often limited, transfer learning proves particularly effective (Kim et al. 2022).

Synthetic data provide another solution to data scarcity. Combining real point cloud data with synthetic data improves deep learning model performance compared to using real data alone (Wu et al. 2018, 2019). In the 3D semantic segmentation of bridges, synthetic point cloud data have proven effective in mitigating data shortages (Jing et al. 2022; Lamas et al. 2024). Generating synthetic point cloud data for bridges typically involves two methods: sampling-based and simulator-based. The sampling-based method samples points from 3D mesh models, enabling rapid and straightforward generation of large data sets, although significant differences often exist between generated and real data. In contrast, the simulator-based method utilizes virtual environments and sensors to create synthetic data closely resembling real scanner outputs (Korus et al. 2023). Despite its potential, research on generating synthetic bridge data using UAV-mounted LiDAR and SLAM remains scarce.

This paper introduces a deep learning-based method for bridge component recognition using point cloud data acquired by a UAV-mounted LiDAR and SLAM. The approach comprises two stages: (1) point cloud acquisition and registration using a UAV-mounted LiDAR and SLAM, and (2) deep learning-based bridge component recognition. In the first stage, data sets of real and synthetic bridge point cloud data are generated for analysis. Synthetic data are created using a simulator-based strategy designed to minimize the domain gap between synthetic and real data. Virtual bridge environments, UAV, and LiDAR are modeled within the simulator, where raw data from the virtual UAV-mounted LiDAR is processed using a SLAM algorithm to produce dense point clouds, similar to real data acquisition.

In the second stage, the Superpoint Transformer (Robert et al. 2023), an advanced deep learning model for 3D semantic segmentation, is employed for bridge component recognition. This model integrates a superpoint graph (Landrieu and Simonovsky 2018) with a transformer network (Vaswani et al. 2017), delivering superior performance in 3D semantic segmentation. The superpoint approach overcomes limitations to conventional models that rely on regular grids or point sampling, which often fail to handle the complexity of 3D data while also reducing memory usage. This ensures efficient processing even for complex structures. The superpoint graph captures relationships between bridge components, whereas the transformer network learns these relationships in depth, enabling precise recognition. Furthermore, the transformer's self-attention mechanism identifies intricate correlations and learns detailed structural features, significantly enhancing recognition accuracy.

This paper presents a comprehensive framework for bridge component recognition, combining synthetic data generation using the Superpoint Transformer specifically optimized for processing point cloud data from a UAV-mounted LiDAR and SLAM. The primary objectives are as follows:

- Employ UAV-mounted LiDAR and SLAM for efficient and accurate 3D modeling of bridges,
- Reduce the domain gap between synthetic and real data using a novel synthetic data generation strategy, and
- Improve the accuracy and efficiency of bridge component recognition for maintenance and inspection.

Literature Review

As-Built 3D Data Acquisition Technologies for Bridge

Advancements in data acquisition technologies have transformed traditional, labor-intensive methods. UAVs combined with photogrammetry now enable efficient and cost-effective point cloud data acquisition for bridges. Lin et al. (2021) introduced an end-to-end aerial robot system for bridge inspection, automating processes from visual data capture to 3D mapping, and defect detection, analysis, and reporting. Their approach enhanced inspection efficiency and accuracy through high-resolution data, automated defect detection, and report generation. Hackl et al. (2018) explored UAVs and photogrammetry for topographical data collection in bridge risk assessments. Experiments on real bridges demonstrated sufficient accuracy for simulating complex flow scenarios with reduced effort compared to traditional surveying. Mehranfar et al. (2018) proposed a projection-based algorithm for generating 3D models of bridge elements, simplifying 3D modeling and improving element merging accuracy. Wang et al. (2023) developed a systematic UAV photogrammetry and 3D reconstruction method for seismic risk assessment, demonstrating its cost-effectiveness and efficiency in bridge network management through a case study. Perry et al. (2020) introduced a UAV and machine learning-based system for bridge inspection, offering automated defect detection, growth tracking, and location mapping. This system generated accurate damage information and 3D visualizations, addressing challenges of traditional inspections and quantifying various damage types. Hu et al. (2021) presented a deep learning-based method to create structure-aware 3D semantic models of cable-stayed bridges using UAV images and point clouds. Their approach emphasized high-level structural relationships, improving robustness against noise and partial scans while outperforming existing reconstruction methods. Xu and Zhang (2022) proposed a UAV-based pipeline for accurate and efficient bridge geometry measurements. Their method achieved greater accuracy and shorter reconstruction times on laboratory arch and suspension bridges than those of conventional techniques. Khaloo et al. (2018) introduced a UAV and 3D reconstruction method for inspecting large-scale infrastructure, overcoming accessibility challenges and achieving high accuracy and completeness while reducing on-site time and safety risks. Mandirola et al. (2022) proposed an unmanned aerial system (UAS)-based method for large-scale bridge inspections, emphasizing safety and cost-effectiveness. Repeated UAS surveys supported long-term structural monitoring through aerial imaging and 3D reconstruction, providing valuable insights for damage assessment. Narazaki et al. (2022) developed a vision-based UAV navigation framework for inspecting concrete railway viaducts postearthquake. Their method demonstrated high efficiency in synthetic environments, achieving centimeter-level column detection accuracy and capturing high-quality images for postearthquake inspection.

The integration of UAVs and photogrammetry enables data acquisition from diverse angles and positions, making it well-suited for monitoring bridges with complex shapes and structures. This approach is also advantageous for 3D reconstruction because it allows image capture from inaccessible locations and multiple perspectives. However, high-quality images require adequate lighting and favorable weather conditions. Moreover, images with indistinct features (e.g., defect-free concrete) or repetitive patterns (e.g., girders) can introduce reconstruction errors, necessitating postprocessing to resolve these issues (Huthwohl et al. 2016).

TLS is a widely used technology for acquiring precise and dense 3D bridge data. Cha et al. (2019) used TLS to detect shape deformations in bridge structures, converting large-scale point cloud data into an octree data structure for efficient processing. Their approach accurately estimated deflections over 4 mm and achieved high scan data compression, demonstrating its effectiveness in managing large-scale scan data for maintenance applications related to deflection and deformation. Yan and Hajjar (2022) developed an automated method to model superstructure elements in steel girder bridges. Their approach partitioned cross-frames into subelements and minimized occlusion impacts using a 3D occlusion labeling algorithm. Unlike existing methods, it generated geometric models suitable for as-built modeling by reflecting actual section properties rather than nominal ones. Yan et al. (2017) introduced a method to automatically generate all-hexahedral finite element meshes from bridge point cloud data. This method produced high-quality finite element meshes for key structural components, including piers, pier caps, girders, and decks. Trias et al. (2022) evaluated TLS as a tool for load rating steel girder bridges and found its accuracy reliable for simulating member-level requirements in load rating evaluations. Liu and Chen (2013) examined bridge assessment methodologies using TLS and developed the LiDAR Bridge Evaluation (LiBE) system that demonstrated high accuracy in detecting and quantifying bridge damage. Their study also identified key limitations of TLS, including scan range, angle, and surface reflectivity, and proposed strategies to address these challenges. Truong-Hong and Lindenbergh (2020) proposed an approach to automatically extract components from TLS point clouds of slab and box beam bridges. Their method successfully identified all component surfaces by utilizing spatial and contextual information, showing potential to replace current workflows based on commercial software. Lu et al. (2019) presented a top-down approach for detecting components in TLS-acquired point clouds of reinforced concrete bridges. Their method handled challenges such as complex structures, occlusions, and sparse data density, significantly reducing computational costs while maintaining accuracy. Riveiro et al. (2016) introduced a fully automated semantic segmentation method for point clouds of masonry arch bridges. By combining heuristic approaches with image processing tools adapted to voxel structures, they classified bridge components, providing critical data for structural health monitoring based on geometric anomalies.

TLS provides high precision and can capture extremely dense point cloud data, making it ideal for detecting shape deformations and defects in infrastructure (Acikgoz et al. 2017; Stałowska et al. 2022; Jing et al. 2024b). However, its high equipment costs, manual sensor positioning, and strict safety requirements for setup make TLS labor-intensive and time-consuming, particularly for large bridges (Huthwohl et al. 2016).

Mobile LiDAR and SLAM have recently gained popularity for real-time data acquisition and rapid 3D modeling of bridges. Yan et al. (2021) proposed a UAV-based method integrating RGB images with mobile LiDAR data for detecting and quantifying concrete cracks. Their approach reduced manual effort by using LiDAR data to automatically identify regions of interest in images and extract pixel sizes without reference objects, improving efficiency and accuracy in crack recognition. Charron et al. (2019) developed a bridge inspection platform using an unmanned ground vehicle (UGV) equipped with calibrated sensors, including mobile LiDAR. The platform generated high-quality 3D point cloud maps of bridge undersides using SLAM, with only a 1.3% difference compared to that of TLS data, significantly reducing inspection time. Gargoum et al. (2018) introduced an algorithm leveraging mobile LiDAR to efficiently assess vertical clearances of highway bridges. The algorithm automated overhead object inventory on highway networks and safely evaluated clearance conditions, proving valuable for bridge management. Yan and Hajjar (2021) presented a heuristic method to segment laser point clouds for identifying key structural elements of steel girder bridges. They validated their method using point clouds from both UAV-mounted mobile LiDAR and TLS.

LiDAR SLAM technologies offer rapid data acquisition and registration, making them highly advantageous for bridge inspection and monitoring. Integrating mobile LiDAR with unmanned vehicles has enabled efficient data collection, with scale error and noise levels deemed acceptable for these tasks (Charron et al. 2019). The maneuverability of UAVs further enhances their utility, allowing for quick acquisition of high-accuracy point cloud data and access to areas difficult for humans to reach. These advancements underscore the need for further research and development to fully realize the potential of these technologies.

Deep Learning-Based 3D Semantic Segmentation of Bridge Components

Three-dimensional (3D) semantic segmentation extracts semantic information from point cloud data, enabling accurate classification and recognition of bridge components. Key approaches to 3D semantic segmentation include heuristic methods (Yan and Hajjar 2021), machine learning (Xia et al. 2022), and deep learning. Among these, deep learning has gained prominence for its robustness and high accuracy in automatically classifying bridge components, eliminating subjective human judgment. This method significantly improves the reliability and efficiency of structural analysis and monitoring in bridge management.

Deep learning for 3D semantic segmentation has progressed from foundational models such as VoxNet (Maturana and Scherer 2015) and PointNet (Qi et al. 2017) to advanced models such as RandLA-Net (Hu et al. 2020) and Point Transformer (Zhao et al. 2021). These approaches excel at recognizing complex patterns in 3D data, making them indispensable for high-precision tasks such as bridge analysis. Kim et al. (2020) proposed a method combining deep learning and subspace partitioning to automatically recognize bridge components from 3D point clouds of full-scale bridges. Their method proved robust, maintaining accuracy even for curved bridges and those with varying pier heights. Lee et al. (2021) developed a hierarchical graph-based convolutional neural network (HGCNN) model for railway bridge segmentation that achieved accuracy comparable to existing models while improving precision for tall components, such as electric poles. Jing et al. (2022) introduced BridgeNet, a 3D deep learning network for subdividing components of masonry arch bridges and extracting key geometric parameters. BridgeNet demonstrated remarkable accuracy, significantly outperforming other networks, with derived geometric parameters closely matching ground truth values. Yang et al. (2022) proposed the weighted superpoint graph (WSPG) method for component segmentation of bridge point clouds. They assigned weights based on class point counts and applied a novel loss function to improve segmentation accuracy and to address the class imbalance issue inherent in bridge data. Xiao et al. (2024) introduced RandLA-BridgeNet, based on RandLA-Net (Hu et al. 2020), to enhance UAV-based bridge inspections through deep learning-based image extraction and crack detection. Their method projected 3D ROIs into 2D images, enabling crack identification while effectively handling complex backgrounds. Jing et al. (2024a) proposed the Geo-Transformer, a lightweight transformer–based module for 3D semantic segmentation of masonry bridges. By combining local attention mechanisms to reduce information loss and global contextual correlations, the Geo-Transformer outperformed existing models across all metrics, highlighting its effectiveness in processing real masonry bridge point clouds.

Deep learning has demonstrated significant success in the semantic segmentation of bridge components, enabling accurate and efficient recognition of complex structures. Among recent advancements, superpoint- and transformer–based models have shown exceptional performance in point cloud semantic segmentation, including applications to bridges. However, existing studies have predominantly focused on data acquired through photogrammetry, TLS, or UGV-mounted LiDAR, which are characterized by stable acquisition conditions and consistent data quality. In contrast, UAV-mounted LiDAR data present unique challenges due to vibrations, movement, and instability during flight, resulting in differences in density, distribution, and noise compared to TLS or UGV data. These characteristics hinder the direct application of existing deep learning models to UAV data. To address these challenges, this paper introduces a novel approach tailored for UAV-mounted LiDAR and SLAM-acquired bridge data, providing new insights into optimizing 3D semantic segmentation workflows for UAV-based bridge inspection.

Synthetic Point Cloud of Bridge

Synthetic point clouds complement real-world data, enhancing the accuracy and efficiency of bridge component classification and analysis, particularly in data-limited scenarios. Jing et al. (2022, 2024a) developed synthetic data sets to train BridgeNet and BridgeNetv2, creating synthetic bridge geometries by combining basic shapes such as rectangles and partial cylinders. These point clouds were intentionally damaged to simulate geometric distortions and laser scanning errors. Lamas et al. (2024) developed a method for generating synthetic point clouds of truss bridges, enabling customizable bridge dimensions, components, and topologies while simulating realistic scanning conditions, including occlusions. Yang et al. (2023) introduced two synthetic data augmentation strategies to improve the weighted superpoint graph (WSPG) model. One approach augmented real bridge data sets with synthetic point clouds, and the other merged synthetic and real superpoints to enhance model training. Korus et al. (2023) introduced DynamoPCSim, a visual programming-based laser scanning simulator, to generate synthetic points for machine learning training. This simulator mimics the operation of scanning lasers, producing synthetic data that closely resemble real-world characteristics. Validation results showed that deep learning models trained on synthetic data achieved comparable performance on both synthetic and real validation data sets. Yang et al. (2024) proposed an enhanced scan-to-BIM framework that integrates simulator-generated synthetic data with real data to improve the SPG model's performance in 3D semantic segmentation. This approach highlights the value of synthetic data in bridging gaps between real and simulated environments.

Although previous studies have demonstrated the effectiveness of synthetic data generation using sampling-based methods (Jing et al. 2022; Yang et al. 2022; Lamas et al. 2024; Yang et al. 2023), these approaches often exhibit a significant domain gap when compared to real-world data. Simulator-based methods (Korus et al. 2023; Yang et al. 2024) have attempted to address this issue by generating synthetic data with more realistic characteristics. However, they do not fully account for the specific challenges of UAV-mounted LiDAR data, such as point cloud instability and noise introduced during flight. To overcome these limitations, this paper proposes a new synthetic data generation strategy tailored to the distinct properties of UAV-mounted LiDAR and SLAM data, enhancing its applicability to UAV-based workflows.

In summary, this paper is distinguished from previous studies in the following ways.

- This paper focuses on the accurate detection of bridge components using UAV-mounted LiDAR data processed with a SLAM algorithm. Although methods such as UAV photogrammetry, TLS, and UGV-mounted LiDAR have proven effective and reliable in many applications, the integration of UAV-mounted LiDAR and SLAM offers unique advantages. These include enhanced capabilities in dynamic and hard-to-reach environments, unlocking new possibilities for UAV-based bridge management and extending its applicability to more challenging scenarios.
- This paper introduces a novel synthetic data generation strategy that explicitly accounts for the unique characteristics of UAV-mounted LiDAR, reducing the domain gap between real and synthetic data sets. This approach provides new insights into improving UAV-based workflows for bridge inspection and monitoring, particularly under challenging conditions.

Methodology

As shown in Fig. 1, the proposed method consists of two stages. In the first stage, point cloud acquisition and registration, a UAV-mounted LiDAR collects raw point cloud data from bridges in both real and virtual environments. These data are then processed using a SLAM algorithm to generate mapped point clouds. The real point

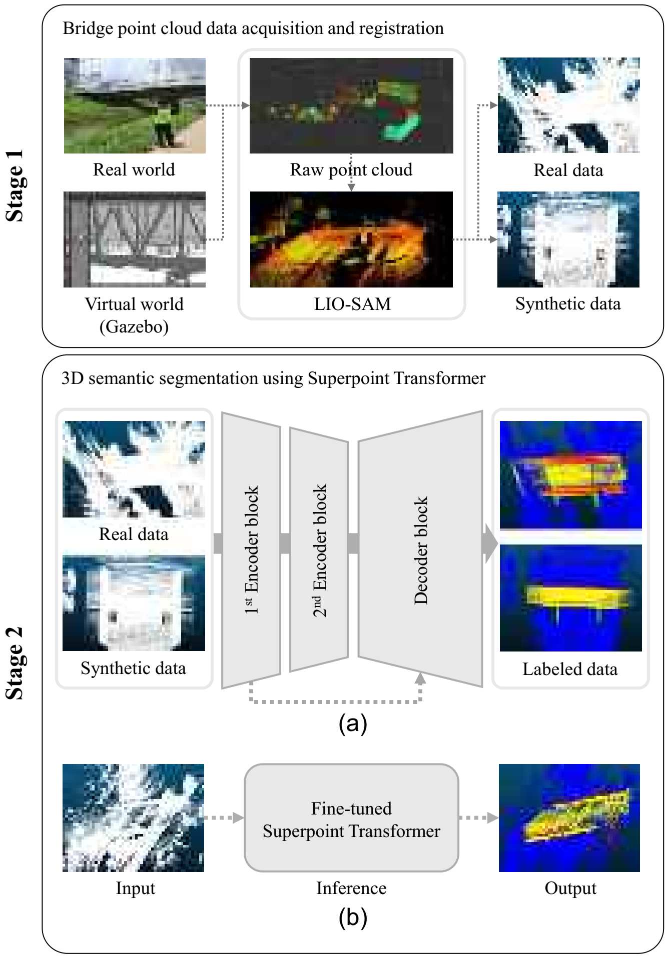

Fig. 1. Overview of proposed method for bridge component recognition: (a) training and validation; and (b) testing. (Images by authors.)

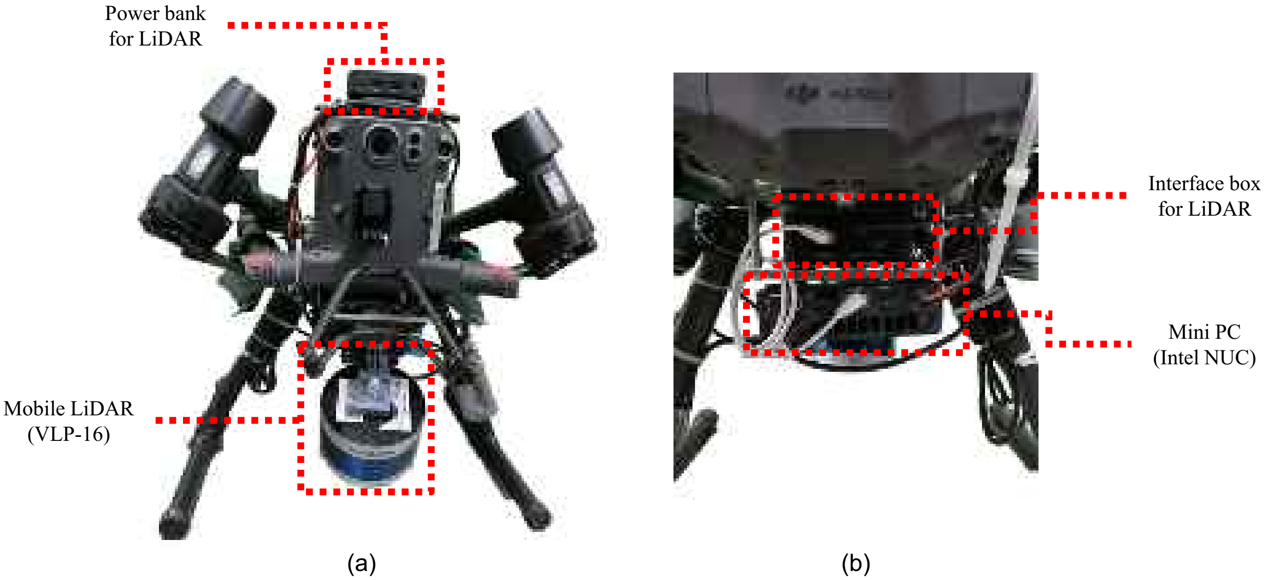

Fig. 2. UAV hardware configuration: (a) front; and (b) rear. (Images by authors.)

cloud data are divided into training, validation, and testing sets, whereas all synthetic data are used exclusively for training. In the second stage, the training and validation data are used to fine-tune a Superpoint Transformer pretrained on the DALES data set (Varney et al. 2020) for bridge component segmentation. The Superpoint Transformer segments the input point cloud into multiple superpoints based on varying roughness levels. Partition, the entire set of the superpoints, is then processed through a transformer network to segment bridge components. Detailed explanations of each stage are provided in the following paragraphs.

Data Acquisition Using UAV-Mounted LiDAR with SLAM Algorithm

LiDAR generates raw point clouds by providing accurate distance and position data, whereas SLAM utilizes these data to estimate the UAV's position and align raw point clouds collected during flight, producing continuous maps. In this paper, LiDAR was used to collect raw point cloud data of the physical environment, and SLAM minimized the data drift caused by UAV movement, enabling the construction of accurate bridge point cloud maps.

A DJI Matrice 300 RTK UAV (Shenzhen, China) equipped with a Velodyne VLP-16 LiDAR (California, USA) was employed to collect point cloud data from eight bridges. The Velodyne VLP-16 LiDAR has a range of up to 100 m, a vertical field of view of ±15°, and a horizontal resolution of 360°. Fig. 2 illustrates the hardware configuration, and Table 1 details the bridges' specifications and flight times. As shown in Fig. 3, the flight path is planned to ensure complete coverage of the bridge, including an initial pass along one side, a detailed scan of the substructure, and a final pass on the opposite side. The flights were conducted manually at speeds of 2 to 3 m/s, with the UAV maintaining a safe distance of 3 to 4 m from the superstructure and 1 to 2 m from the substructure to ensure detailed scans while avoiding collisions.

The LIO-SAM algorithm (Shan et al. 2020) employs a tightly coupled approach that integrates LiDAR and Inertial Measurement Unit (IMU) data in real-time. Leveraging factor graph optimization minimizes the drift caused by UAV movements and vibrations, which can significantly degrade raw LiDAR data. To further address drift, LIO-SAM utilizes loop closure detection to realign overlapping scans during repeated passes over the same area. This method is particularly effective for structures with repetitive geometries, such as bridge girders and piers. IMU-LiDAR calibration is achieved by determining an extrinsic transformation matrix through the procedure described by Chennuo0125 (2020), ensuring that motion data from the IMU aligns accurately with the LiDAR frame for precise odometry corrections.

This paper introduces a new strategy to minimize the domain gap between synthetic and real bridge point cloud data. First, 3D mesh models of eight different bridges were collected and used to construct a virtual bridge environment within the Gazebo simulator, part of the Robot Operating System. Second, a UAV flight simulator (Alqutami 2020) and a LiDAR simulator (Dataspeed 2015) were integrated with the virtual bridge environment to

Table 1. Real-world bridge specifications and flight information

Bridge	Location	Type	Dimensions (length × width × height)	Flight time (s)
(a)	Korea	Prestressed concrete	41 × 15 × 5 m	608
(b)	Korea	Prestressed concrete	49 × 16 × 5 m	655
(c)	Korea	Prestressed concrete	55 × 31 × 4 m	782
(d)	Korea	Prestressed concrete	74 × 36 × 4 m	273
(e)	Korea	Prestressed concrete	41 × 16 × 5 m	494
(f)	Korea	Prestressed concrete	38 × 11.5 × 5 m	475
(g)	Korea	Prestressed concrete	57 × 21 × 5 m	586
(h)	Korea	Prestressed concrete	40 × 15 × 5 m	670

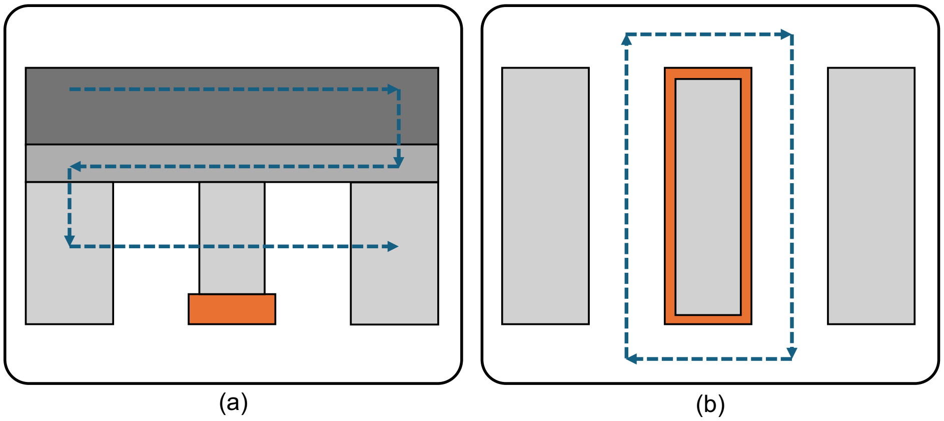

Fig. 3. Flight path: (a) side of bridge; and (b) substructure (bird-eye-view).

acquire raw point cloud data. The UAV flight simulator employed the DJI Matrice 600 model to capture flight data, including IMU readings, whereas the LiDAR simulator used the Velodyne VLP-16 model with Gaussian noise applied to replicate real-world sensor noise (mean: 0.000, standard deviation: 0.008). Various physical effects were also modeled, such as UAV and sensor tilting during flight, vibrations from collisions, and laser blockage by objects, to enhance the realism of the synthetic data. Finally, LIO-SAM was used to process synthetic point cloud data, mirroring the acquisition process of real data sets.

3D Semantic Segmentation of Bridge Components Using the Superpoint Transformer

The superpoint transformer efficiently processes point cloud data by combining a superpoint graph with a transformer network. Superpoints, which group points with similar geometric and positional features, are arranged hierarchically to model relationships. The transformer network leverages this hierarchical structure to analyze interactions between superpoints, enabling fast and accurate processing of complex point clouds. This approach reduces the computational complexity of large data sets, such as bridge point clouds, and preserves essential structural details.

In the first step, geometric features f_c are extracted for each point in the cloud. These include linearity, planarity, scattering, and verticality, which are calculated using principal component analysis (PCA), and elevation, which is determined using the random sample consensus (RANSAC) algorithm. Together, these descriptors capture the geometric properties of the points and are essential for identifying relationships between neighboring points in subsequent steps. Fig. 4 visualizes five geometric features of the bridge point cloud, with brighter colors indicating higher feature values.

Next, the k-nearest neighbors (KNN) algorithm is used to construct an initial graph $G \coloneqq (C, \mathcal{E}, \omega)$, where each point is a node, and the extracted features f_c serve as signals $\boldsymbol{f}$ on the nodes. Edges $\mathcal{E}$ represent connections between neighboring points, while the edge weights ω reflect feature similarity or spatial proximity.

The graph structure is then optimized to group points into superpoints by solving an energy minimization problem defined in Eq. (1):

$$J(\boldsymbol{e};\boldsymbol{f}, G, \lambda) = \|\boldsymbol{e} - \boldsymbol{f}\|^2 + \lambda \sum_{(u,v)\in\mathcal{E}} \omega_{uv}[e_u \neq e_v] \tag{1}$$

where $\boldsymbol{e}$ = approximation of $\boldsymbol{f}$; λ = regularization strength; u and v = neighbor points; and the function $[a \neq b]$ equals 0 if a and b are the same and 1 otherwise. This energy function measures how well the signal e represents the input point cloud's features f in $\|e - f\|^2$ and measures the complexity of the partition in $\lambda\sum_{(u,v)\in E}\omega_{uv}[e_u \neq e_v]$.

Since the problem is nonconvex, the ℓ_0-cut pursuit algorithm is used for approximate optimization, iteratively cutting graph edges

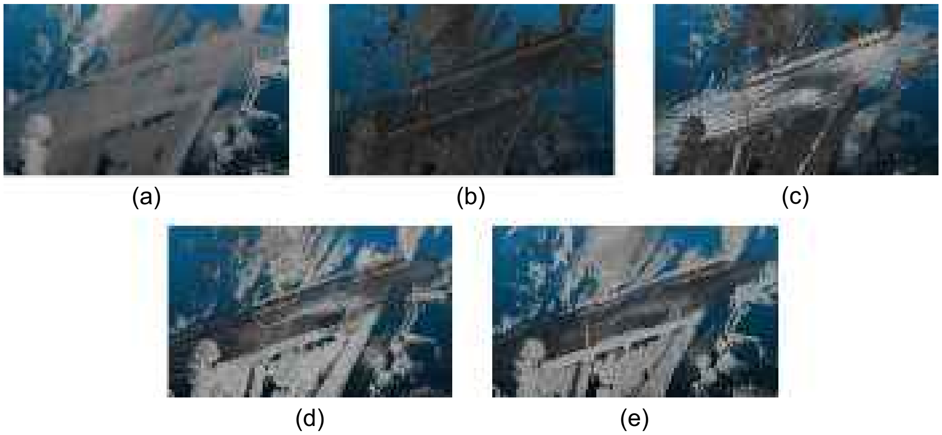

Fig. 4. Visualization of geometric features: (a) elevation; (b) linearity; (c) planarity; (d) scattering; and (e) verticality.

based on the edge weights ω and recalculating the energy costs. This process identifies optimal partitions, forming sets of superpoints. To further improve computational efficiency, parallel graph cuts are employed, allowing multiple graph segments to be processed simultaneously and significantly accelerating the partitioning process for large-scale data sets.

The superpoint transformer organizes point cloud data into a hierarchical partitioning structure by adjusting the partition size across multiple scales using the regularization parameter λ. Partitioning starts at the finest level, forming smaller and more detailed superpoints, and progressively becomes coarser as λ increases, grouping smaller superpoints into larger ones. This process creates a multilevel hierarchy, typically comprising three levels (from level 0 to level 2), where the size of the superpoints increases with each level. At each level, the energy function is optimized to balance precision and computational efficiency by grouping similar points into the same superpoint while minimizing the difference between the original point features and their approximations.

For each superpoint $p \in P_i$ (where $i = 0,\ 1,\ 2$), lower level superpoints are subsets of higher level superpoints. The position information x_p^i of a superpoint p is defined relative to the centroid of higher level superpoints (parent). Similarly, the position of lower level superpoints (children) is defined relative to x_p^i. At the highest level (level 2), the positions of the superpoints are calculated based on the centroid of the entire point cloud.

The multiscale partitions are then passed into the transformer–based network, which learns the relationships within and between the superpoints across different hierarchical levels. This allows the network to capture contextual information about the structural relationships between components, such as those found in complex bridge data.

The transformer–based network of the superpoint transformer takes multiscale hierarchical partitions as input to classify bridge components. The core of the network is a self-attention mechanism, which enables the model to learn the feature information of each superpoint and the contextual information between superpoints. As shown in Fig. 5, the transformer network has an encoder-decoder architecture consisting of two transformer encoder blocks [$\mathrm{Trans}_{\mathrm{enc}}^{i}(i \geq 1)$] and one decoder block [$\mathrm{Trans}_{\mathrm{dec}}^{i}(i = 1)$].

The features of the superpoints generated in the hierarchical partitioning process pass through a multilayer perceptron (MLP) before entering the transformer blocks:

1. $\mathrm{MLP}_{\mathrm{enc}}^{i}(i \geq 0)$ and $\mathrm{MLP}_{\mathrm{dec}}^{i}(i = 1)$ combine the relative positions and handcrafted features of the superpoints.
2. $\mathrm{MLP}_{\mathrm{adj}}^{i}(i \geq 1)$ encodes the adjacency feature, representing the relationship between two superpoints p and q in partition P_i.

For each level of the encoder and decoder, each superpoint's key, query, and value vectors $\boldsymbol{K_p}$, $\boldsymbol{Q_p}$, $\boldsymbol{V_p}$ of dimensions D_{key}, D_{key}, and D_{val}, respectively, are generated. The relational features between two superpoints p and q, denoted as $\boldsymbol{a_{p,q}^{key}}$, $\boldsymbol{a_{p,q}^{que}}$, $\boldsymbol{a_{p,q}^{val}}$ of dimensions D_{key}, D_{key}, and D_{val} are stacked to form the matrices $\boldsymbol{A_p^{key}}$, $\boldsymbol{A_p^{que}}$, and $\boldsymbol{A_p^{val}}$ of size $|N(p)| \times D_{\mathrm{key}}$. The modules $\mathrm{Trans}_{\mathrm{enc}}^{i}$ and $\mathrm{Trans}_{\mathrm{dec}}^{i}$ integrate these vectors through Eqs. (2) and (3) to obtain contextual information:

$$\mathrm{att}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \boldsymbol{V}^{T}\mathrm{softmax}\left(\frac{\boldsymbol{Q}\odot\boldsymbol{K}}{\sqrt{N(p)}}\right) \tag{2}$$

$$[\mathrm{Trans}(m)]_{p} + = att(\boldsymbol{Q_p^T} \oplus \boldsymbol{A_p^{que}}, \boldsymbol{K_{N(p)}} + \boldsymbol{A_p^{key}}, \boldsymbol{V_{N(p)}} + \boldsymbol{A_p^{val}}) \tag{3}$$

where $\odot$ = Hadamard product, which performs element-wise multiplication for two matrices of the same dimensions; m = feature map of superpoint p; $+$ = indicates a residual connection; $\oplus$ = addition operator with broadcasting on the first dimension; and $\boldsymbol{K_{N(p)}}$ and $\boldsymbol{V_{N(p)}}$ = matrices of stacked vectors $\boldsymbol{K_q}$ and $\boldsymbol{V_p}$ for $q \in N(p)$.

The transformer–based network learns both the individual features of superpoints and the relational features between them. The attention mechanism calculates attention weights by considering

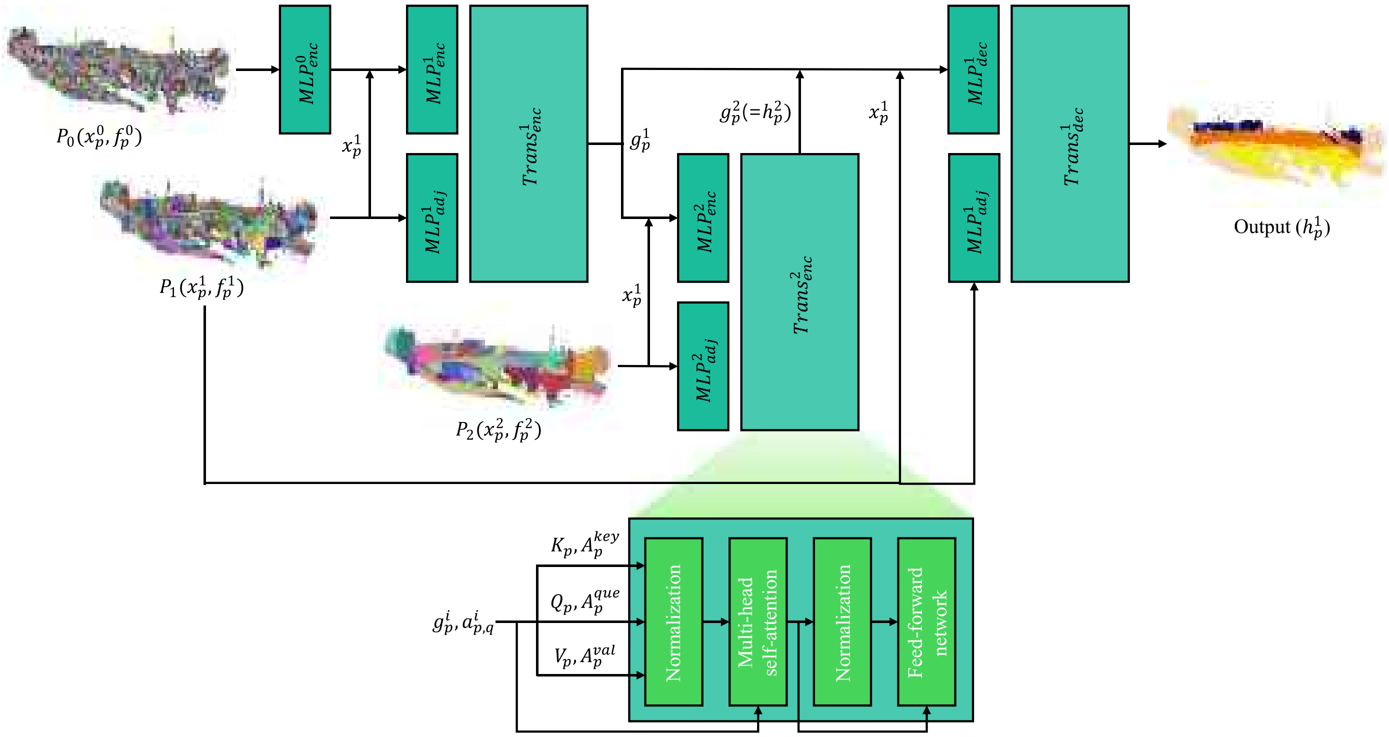

Fig. 5. The architecture of Superpoint Transformer network.

the interactions between individual and relational features. The Hadamard product is used to evaluate these interactions independently. Through this process, the Superpoint Transformer can identify the features that are important and the superpoints that require more attention when classifying bridge components.

The loss function of the Superpoint Transformer is based on cross-entropy loss, defined as in Eq. (4):

$$\mathcal{L} = \sum_{p\in P_1} \frac{-N_p^1}{|C|} H(\hat{y}_p^1, z_p^1) + \sum_{i=2}^{I} \sum_{p\in P_i} \frac{\mu^i N_p^i}{|C|} H(y_p^i, z_p^i) \qquad (4)$$

where N_p^i = number of points in superpoint p within partition P_i; $|C|$ = total number of points in the point cloud; $H(y,z) = -\sum_{k\in K} y_k \log(z_k)$ represents the cross-entropy between the actual label distribution y and the predicted label distribution z; K = set of classes; μ^i = positive weight for each layer from $i = 2$ to I, and since the model has two levels of partitions, $I = 2$ in this paper.

This loss function performs the following two essential tasks: (1) predicting the most frequent label for each finest-level superpoint, and (2) predicting the distribution of labels for each coarser-level superpoint. This approach enables the model to effectively learn semantic information at each level of the hierarchical structure, achieving accurate semantic segmentation of the entire point cloud.

Bridge point cloud data are characterized by diverse components, each with unique geometric, radiometric, and spatial properties. The hierarchical partitioning of bridge point cloud data provides a better representation of the geometric features and spatial relationships of these components. The transformer–based network learns these features and relationships, allowing it to understand the structural patterns and connections of the entire bridge. Through this process, the Superpoint Transformer can accurately classify bridge components.

Experiment

Point Cloud Data Set Preparation and Annotation

Fig. 6 and Table 2 present the real point cloud data sets and their descriptions. Annotation was performed manually using CloudCompare (Girardeau-Montaut 2016) to classify the point clouds into four categories: background, superstructure, substructure, and parapet. Clear geometric and spatial criteria were applied during annotation to ensure consistency. Background includes nonbridge elements such as vegetation and building, superstructure covers girders and decks, substructure comprises piers, pier caps and abutments, and parapet refers to protective barriers. Table 2 shows that real bridges (a), (b), and (c) are used as training data, (d) and (e) as validation data, and (f), (g), and (h) as testing data.

Fig. 7 and Table 3 describe the synthetic point cloud data sets, annotated using the same method as that used for the real data sets. These synthetic data sets were combined with the real data sets to train the Superpoint Transformer.

Implementation Details

The raw point cloud data acquired from the LiDAR sensor and the IMU data collected from the UAV's onboard sensors were stored in ROS-based bag file format. The registered point clouds, processed using LIO-SAM, were saved in PLY file format. Virtual bridge environments for synthetic data generation and the training of the Superpoint Transformer were conducted on a computer running Ubuntu 18.04 OS and equipped with an Intel Core i7-10700KF CPU and an Nvidia GeForce RTX 3090 GPU.

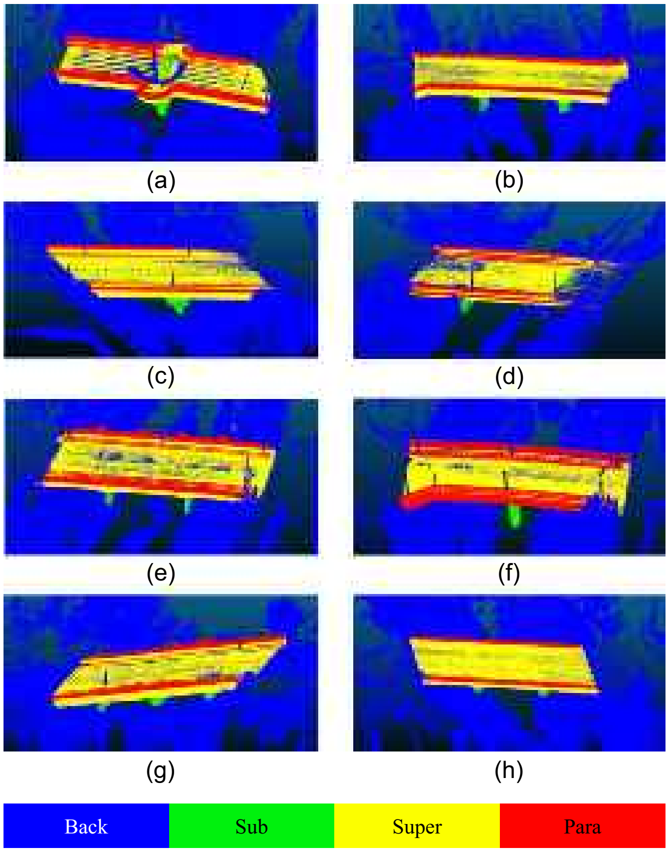

Fig. 6. Visualization of labelled real bridge point cloud data sets.

Table 2. Description for real bridge data sets (number of points)

Bridge	Back	Sub	Super	Para	Total
			Training		
(a)	28,240,853	1,726,171	2,856,554	1,451,746	34,275,324
(b)	35,645,205	2,690,276	4,384,671	1,053,825	43,773,977
(c)	24,196,087	1,143,888	3,020,324	260,617	28,620,916
			Validation		
(d)	6,889,226	977,311	1,662,405	370,546	9,899,488
(e)	13,863,681	1,368,541	1,854,340	507,745	17,594,307
			Testing		
(f)	15,782,580	745,190	1,658,090	613,357	18,799,217
(g)	15,591,982	942,744	1,689,913	559,656	18,784,295
(h)	18,890,343	1,170,230	2,153,429	486,005	22,700,007

Note: Back = background; sub = substructure; super = superstructure; and para = parapet.

To train the Superpoint Transformer, the hyperparameters were set as follows: batch size of 1, learning rate of 0.01, and 400 epochs. A key parameter influencing model performance is the regularization strength, which determines the partitioning roughness. Based on validation experiments, values of 0.1, 0.2, and 0.4 were selected as optimal, as detailed in Section 4.2.

The mean Intersection over Union (mIoU) was used as the primary evaluation metric, which measures the average overlap between predicted and ground truth classifications across all classes. Defined in Eq. (5), mIoU is expressed as follows:

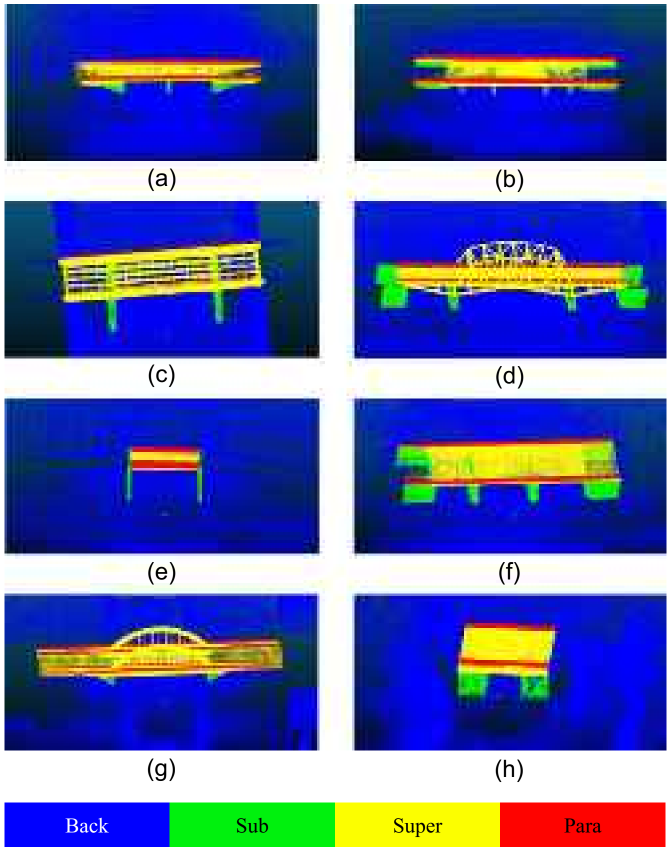

Fig. 7. Visualization of labelled synthetic bridge data sets.

Table 3. Description for synthetic data sets (number of points)

Bridge	Back	Sub	Super	Para	Total
(a)	1,317,613	81,454	278,392	103,083	1,780,542
(b)	627,069	117,011	206,445	135,152	1,085,677
(c)	2,105,742	319,439	563,285	22,836	3,011,302
(d)	3,393,162	340,569	1,151,452	181,709	5,066,892
(e)	1,002,953	88,051	57,765	90,915	1,239,684
(f)	3,285,452	578,232	493,971	154,701	4,512,356
(g)	2,505,678	52,865	646,716	141,058	3,346,317
(h)	2,565,778	156,595	172,238	53,236	2,947,847

$$\text{mIoU} = \frac{1}{M}\sum_{i=1}^{M}\frac{(TP)_i}{(TP)_i + (FP)_i + (FN)_i} \tag{5}$$

where M = total number of classes; TP_i = true positives, indicating the number of points correctly classified as class i; FP_i = false positives, indicating the number of points incorrectly classified as class i; and FN_i = false negatives, indicating the number of points that belong to class i but are classified as another class.

Setting Regularization Strength

The regularization strength directly affects the size of the superpoints, influencing segmentation performance. Lower regularization strengths produce finer partitions that capture intricate details and complex structures, whereas higher values generate coarser partitions, reducing overfitting and improving generalization. However, excessively fine partitions can result in slower training and increased sensitivity to noise or outliers, whereas excessively coarse partitions may lead to the loss of critical details, ultimately degrading performance. A hierarchical graph structure connects the various partition levels, allowing the model to learn relationships between fine patterns at lower levels and general patterns at higher levels. Table 4 presents segmentation results for the validation data sets, highlighting the impact of regularization strength combinations on the IoU of each class. For example, the combination (0.01, 0.1, 0.5) yielded the highest IoU for the substructure class but the lowest for the parapet class. Considering the segmentation results for the substructure, superstructure, and parapet classes, the regularization strength values (0.1, 0.2, 0.4) were selected.

Table 4. Segmentation results according to regularization strength values (IoU, %)

Regularization strength	Back	Sub	Super	Para	mIoU
0.01, 0.1, 0.5	94.037	**78.879**	75.953	72.573	80.361
0.02, 0.05, 0.1	94.298	77.624	75.371	75.464	80.689
0.05, 0.1, 0.4	95.999	55.736	75.665	71.117	74.629
0.1, 0.2, 0.3	93.780	77.564	77.065	76.083	81.123
0.1, 0.2, 0.4	94.247	77.761	**77.260**	**77.131**	**81.600**
0.1, 0.2, 0.6	93.849	77.312	76.250	77.074	81.121
0.1, 0.5, 1.0	**97.652**	67.891	74.340	69.362	77.311
0.2, 0.3, 0.6	97.340	71.998	75.801	72.707	79.461
0.2, 0.4, 0.6	94.073	78.162	75.489	76.600	81.081

Note: Bold values indicate the highest IoU.

Table 5. Segmentation results with and without synthetic bridge point cloud data (IoU, %)

Data	Back	Sub	Super	Para	mIoU
Without synthetic	93.405	70.392	75.114	75.687	78.650
With synthetic	**94.247**	**77.761**	**77.260**	**77.131**	**81.600**

Note: Bold values indicate the highest IoU.

Impact of Synthetic Data on Model Performance

An experiment was conducted using a validation data set to evaluate the impact of synthetic data on model training. As shown in Table 5, including synthetic data in the training data set improved the mIoU from 78.650% to 81.600% compared to that of training without synthetic data. Notably, the IoU for the substructure class increased significantly from 70.392% to 77.761%, and improvements were observed across other classes as well. This improvement is attributed to the higher proportion of bridge component points in the synthetic data. The average point ratios of the substructure, superstructure, and parapet classes in the three real training data sets were 5.06%, 9.63%, and 2.52%, respectively, whereas the corresponding ratios in the eight synthetic data sets were 7.44%, 14.82%, and 4.71%, respectively. These ratios indicate that synthetic data provided a more balanced representation of bridge components, enabling the model to learn more effectively. Furthermore, substructures exhibit significant variability in shape depending on the presence and configuration of piers, pier caps, and abutments. The inclusion of synthetic data allowed the model to learn diverse substructure shapes, which likely contributed to the substantial performance improvements. These results confirm that synthetic point cloud data enhanced the ability of the Superpoint Transformer to generalize across various bridge components.

Impact of Synthetic Data Generation Methods on Model Performance

A comparative experiment was conducted using the traditional sampling-based method to evaluate the effectiveness of the simulation-based strategy for synthetic data generation proposed in this paper. Synthetic point clouds were sampled from 3D mesh models using CloudCompare version 2.11.1, open-source software for 3D point cloud and mesh processing. During the sampling process, CloudCompare generates points by ensuring a uniform distribution across the surface of the mesh model. Users can specify either the approximate total number of points to be sampled or the surface density (e.g., points per unit area). The software calculates the surface area of each triangle in the mesh and allocates points proportionally to each triangle's area. Within each triangle, the points are randomly positioned while ensuring an overall uniform density across the entire mesh surface. For this experiment, the number of points to be sampled for each class was specified to approximately match the class proportions of the simulation-based synthetic data generated in this paper. This adjustment of class proportions ensures fairness and eliminates potential bias in the experiment. The synthetic data generated using the sampling-based method are shown in Fig. 8 and Table 6.

As shown in Table 7, our method achieved an mIoU of 81.600% on validation data sets, surpassing the sampling-based method, which achieved 77.812%. The improvement, particularly within the superstructure and parapet classes, ranged from 6% to 7%. These results underscore the effectiveness of our strategy in generating synthetic data that align more closely with real-world scenarios. A key advantage of our method lies in its ability to address the limitations of the sampling-based method in terms of occlusion and point distribution. The sampling-based method generates points even in occluded areas that would not be visible during real-world data acquisition. For example, as shown in Fig. 9(a), parts of the abutment embedded in the ground and portions of the girder inside the abutment are visible in the sampled point cloud. In contrast, our simulation-based strategy excludes points in occluded areas, as demonstrated in Fig. 9(b), resulting in synthetic data that more accurately reflect real-world acquisition constraints. Additionally, the point distribution in the sampling-based method is uniform across the mesh surface, ignoring practical data acquisition limitations. The proposed strategy simulates the movement and perspective of a UAV in a virtual environment, producing a point distribution that mirrors the patterns observed in real-world UAV-LiDAR data. For instance, points are denser in regions closer to the UAV's flight path and sparser in areas further away or difficult to access. By operating a UAV in the simulator, our method generates synthetic data with a realistic point distribution and eliminates points in areas that would typically be occluded, contributing to its superior performance in downstream tasks such as segmentation and classification.

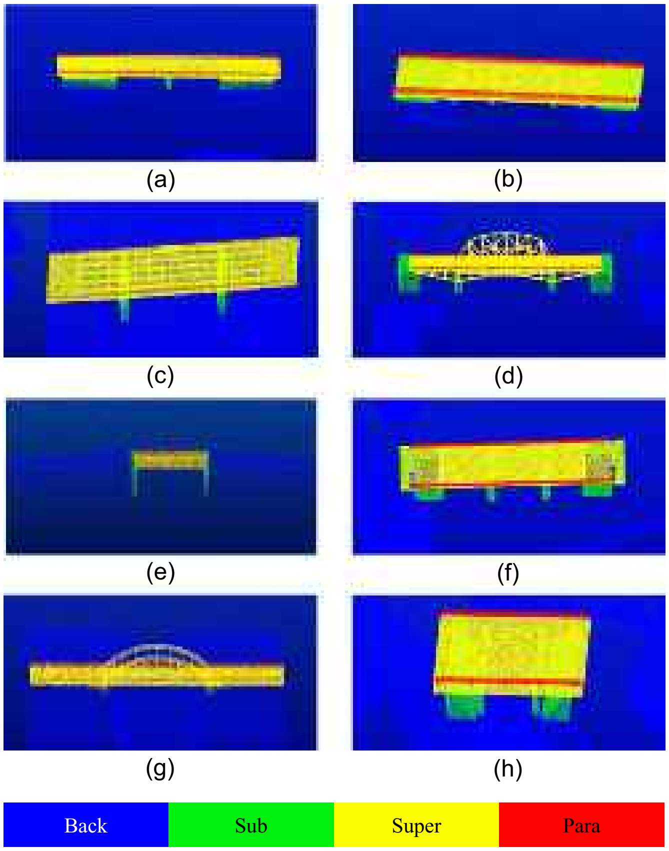

Fig. 8. Visualization of synthetic data generated using sampling-based method.

Experimental Results

The performance of the proposed method on testing data was evaluated using hyperparameters determined through comparative experiments and synthetic data generated based on our strategy. Additionally, the same task was performed using RandLA-Net, which has demonstrated notable semantic segmentation performance on bridge data acquired via TLS (Xiao et al. 2024). Table 8 presents the quantitative results of these tests, and Figs. 10(a–c) show the ground truth, predictions from the Superpoint Transformer, and predictions from RandLA-Net, respectively.

Regarding segmentation performance, the Superpoint Transformer achieved an mIoU of 86.125%. The performance for each class was as follows: 98.145% for the background, 87.674% for the substructure, 84.206% for the superstructure, and 74.475% for the parapet. Although RandLA-Net has previously achieved successful semantic segmentation of TLS bridge data (Xiao et al. 2024), it significantly underperformed on our testing data. RandLA-Net achieved an mIoU of 64.364%, with class-specific IoU values of

Table 6. Description for synthetic data generated using sampling-based method

Data	Back	Sub	Super	Para	Total
Synthetic 1	1,320,030	81,993	279,998	104,969	1,786,990
Synthetic 2	631,078	120,010	224,997	119,998	1,096,083
Synthetic 3	2,112,064	319,995	575,029	22,993	3,030,081
Synthetic 4	3,399,994	347,247	1,160,067	190,020	5,097,328
Synthetic 5	999,990	89,990	59,990	91,024	1,240,994
Synthetic 6	3,299,959	580,004	499,999	154,986	4,534,948
Synthetic 7	2,509,991	54,001	640,020	142,013	3,346,025
Synthetic 8	2,624,283	159,990	175,000	54,005	3,013,278

Table 7. Segmentation results according to synthetic data generation methods (IoU, %)

Method	Back	Sub	Super	Para	mIoU
Sampling-based	92.742	76.397	71.515	70.594	77.812
Ours	**94.247**	**77.761**	**77.260**	**77.131**	**81.600**

Note: Bold values indicate the highest IoU.

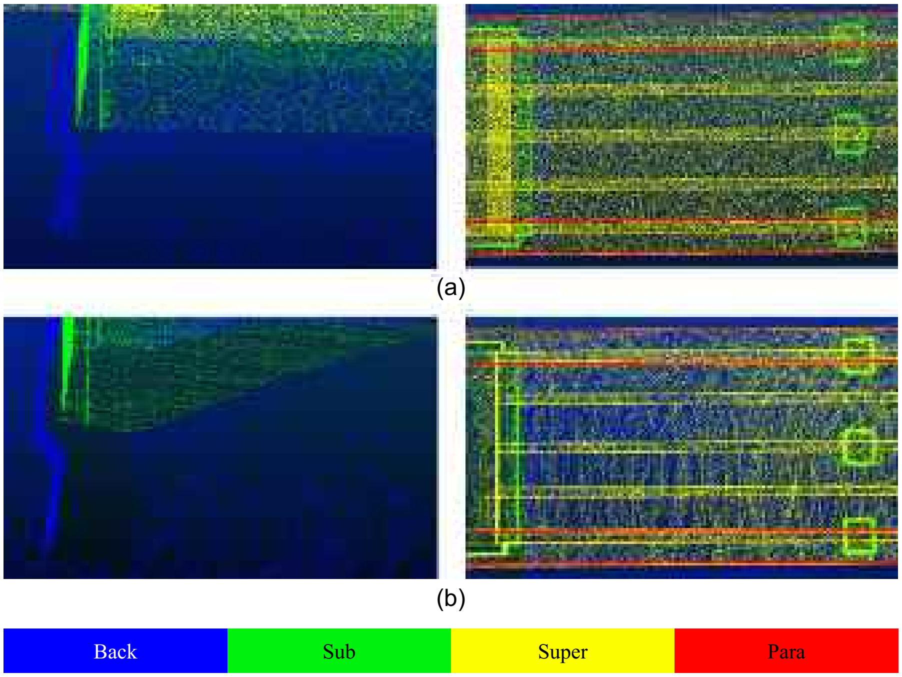

Fig. 9. Differences based on synthetic data generation methods: (a) sampling-based method; and (b) our strategy.

Table 8. Quantitative results for testing data set (IoU, %)

Model	Back	Sub	Super	Parapet	mIoU
RandLA-Net	93.399	54.811	64.506	44.740	64.364
Ours	**98.145**	**87.674**	**84.206**	**74.475**	**86.125**

Note: Bold values indicate the highest IoU.

93.399% for the background, 54.811% for the substructure, 64.506% for the superstructure, and 44.740% for the parapet. As shown in Table 9, the Superpoint Transformer also outperforms RandLA-Net regarding inference time. For the same testing data, RandLA-Net took 100.187 s, whereas the Superpoint Transformer required only 9.294 s.

Discussion

Interpretation of Results

The IoU for the parapet class was relatively low compared to that of other classes, which can be attributed to two main reasons. First, the parapets extend beyond the bridge. As shown in Fig. 11(a), parapets sometimes continue along walkways outside the bridge. In this paper, only the parapets on the bridge were segmented as the parapet class, whereas those extending beyond were segmented as the background class. Technically, no distinction exists between these two types of parapets, which may have led to misclassification. Second, objects adjacent to the parapets can cause segmentation errors. For instance, as shown in Fig. 11(b), streetlamp bases attached to the parapets were sometimes incorrectly classified as part of the parapets. Due to their small and intricate structures, parapets may be difficult to capture accurately with lower-resolution point cloud data, potentially hindering the model's segmentation accuracy.

Compared to previous studies that performed semantic segmentation of bridge components using deep learning, the performance of the proposed method was relatively lower, particularly in experiments using RandLA-Net. This can be attributed to two primary factors. First, UAV-mounted LiDAR data have more noise compared to TLS bridge data. UAV movement, vibrations, and environmental factors can introduce additional noise, making it more challenging for the model to accurately segment the data. Second, class imbalance could impact the model's performance. TLS bridge data published by Lu et al. (2019) showed a background-to-bridge point ratio of approximately 55:45, whereas our data sets exhibit a more pronounced imbalance, with a ratio of 77:23. This imbalance arises because the Velodyne VLP-16 model used for data acquisition has a range of 100 m and collects numerous background points, especially in urban areas.

The Superpoint Transformer can mitigate the impact of noise and address class imbalance issues through superpoint clustering and a hierarchical structure. By grouping points with similar features, the transformer minimizes the influence of noisy points. Rare class points can be clustered with nearby points, leading to a more balanced class representation. The hierarchical structure allows the model to learn both local and global contexts simultaneously, reducing the impact of outliers such as noise. This approach enhances the model's understanding of global relationships between classes, making the Superpoint Transformer particularly suitable for our data.

Limitations and Future Research Directions

The experimental results demonstrate that the proposed method is effective in recognizing bridge components from point clouds. Furthermore, a new strategy for generating synthetic bridge data using a UAV and LiDAR has improved the performance of the Superpoint Transformer in recognizing bridge components by reducing the domain gap between real and synthetic data. Nevertheless, several improvements are needed for the proposed method. First, automated data acquisition using UAV navigation could be developed because this paper relied on manually operated UAVs for data collection. Second, based on the semantic information of bridge components

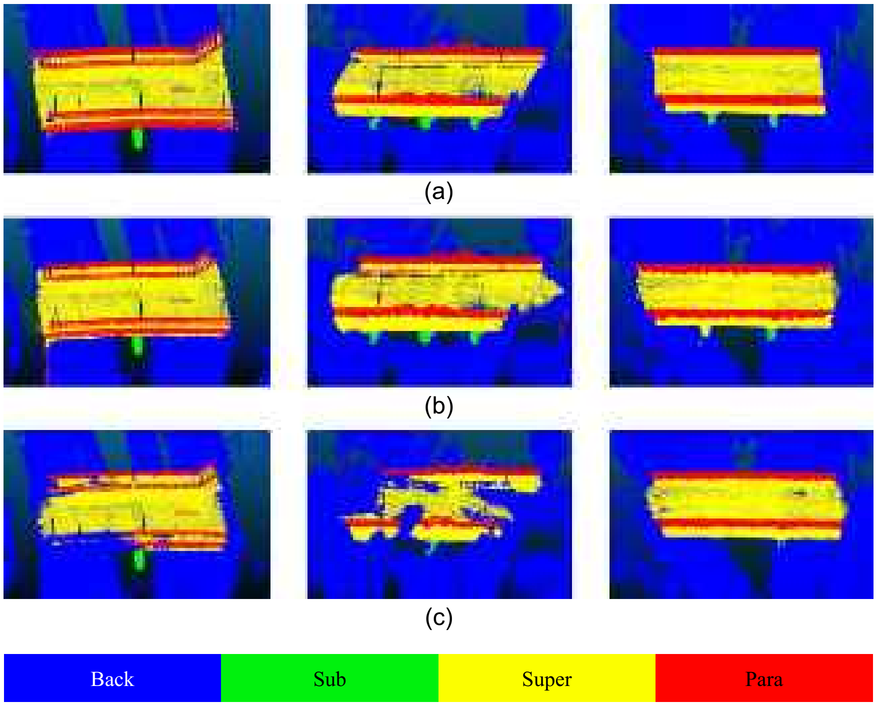

Fig. 10. Visualization of testing results: (a) ground truth; (b) predictions of Superpoint Transformer; and (c) predictions of RandLA-Net.

Table 9. Inference time and GPU memory consumption of models

Model	Inference time (s)	GPU memory consumption (MB)
RandLA-Net	100.187	16,492
Ours	9.294	18,132

extracted in this paper, an advanced system could be developed to automatically generate as-built BIMs for bridges. Third, detecting and locating defects within bridges is crucial for bridge management. Real-time defect detection and localization could be achieved through sensor fusion between mobile LiDAR and cameras, allowing for more efficient bridge management. By addressing these improvements, the proposed method could further enhance the accuracy and efficiency of bridge inspection and maintenance processes.

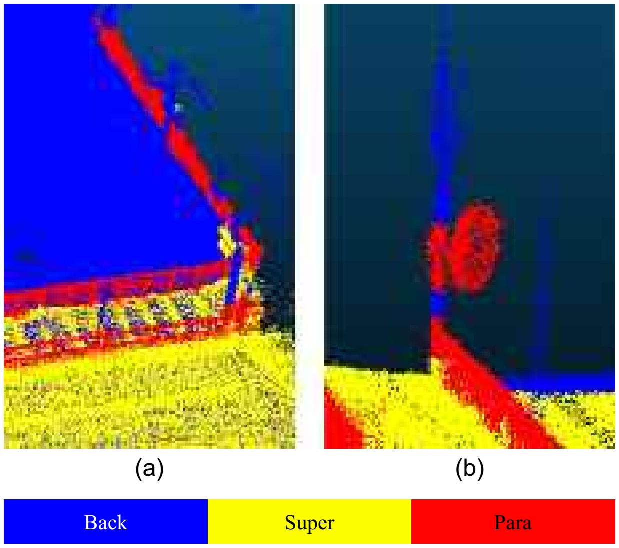

Fig. 11. Examples of incorrect segmentation: (a) parapet extending beyond bridge; and (b) streetlamp misclassified as parapet.

Conclusion

A method for bridge component segmentation from point cloud data acquired by a UAV-mounted LiDAR and SLAM was developed and evaluated. In this process, bridge data sets comprising eight real and eight synthetic bridge point clouds were produced. To generate synthetic bridge point cloud data, a new strategy was introduced using the ROS Gazebo simulator, which simulated virtual bridges, UAV, and LiDAR. Following the same procedure as that of real data acquisition, raw bridge point cloud data were acquired using a UAV-mounted LiDAR, and point clouds were registered using LIO-SAM. By fine-tuning the model with both synthetic and real bridge point cloud data, the Superpoint Transformer model suitable for bridge management using a UAV-mounted LiDAR was developed. The fine-tuned model effectively identified bridge components with an mIoU of 86.125% on testing data sets. The class-specific IoUs were 98.145% for the background, 87.674% for the substructure, 84.206% for the superstructure, and 74.475% for the parapet. Additionally, the model achieved this inference in only 9.294 s. The experimental results suggest that the proposed method could be valuable in scenarios requiring precise bridge management.

The contributions of this paper are fourfold. First, this paper successfully demonstrates that fine-tuning the Superpoint Transformer with both synthetic and real bridge point cloud data, specifically acquired using a UAV-mounted LiDAR and SLAM, can significantly improve the accuracy of bridge component recognition. This approach is highly effective in challenging urban environments, where the presence of vehicles, pedestrians, and surrounding structures complicates accurate segmentation. Despite these complexities, the model achieved an mIoU of 86.125% and an inference time of only 9.294 s without extensive preprocessing, showcasing its robustness for real-world bridge management applications.

Second, this paper introduces a novel simulator-based strategy for generating synthetic bridge point cloud data that closely mirror real-world acquisition using UAV-mounted LiDAR. Unlike previous methods that rely on sampling-based techniques or different sensor platforms, this approach reduces the domain gap between synthetic and real data, significantly improving model performance. This finding underscores the importance of simulating data under realistic conditions, particularly in scenarios in which real data are difficult to obtain.

Third, the integration of UAVs and LiDAR for fast and efficient data acquisition in bridge management is demonstrated. By leveraging mobile LiDAR and SLAM algorithms, the method enables real-time 3D geometric data acquisition, even in areas that are otherwise difficult to access. This approach not only streamlines the data collection process but also enhances the practicality of UAV-based systems for large-scale infrastructure management.

Fourth, a new bridge point cloud data set for semantic segmentation was created. The data set consists of eight real and eight synthetic bridge point clouds, labeled into four classes (background, substructure, superstructure, parapet). Since the data set was generated using UAV-mounted LiDAR and SLAM, it is particularly useful for research on autonomous UAV navigation, making it highly applicable in developing advanced methods for the automated inspection and maintenance of bridges. This provides a valuable resource for future researchers aiming to develop new systems for bridge management.

In conclusion, the findings of this paper offer advancements in the field of bridge management. By addressing the challenges of data acquisition and model training, this paper lays the groundwork for more efficient and accurate bridge management. Although the data in this paper primarily focus on prestressed concrete bridges commonly found in urban areas of Korea, the proposed method is adaptable to various types of bridges. Future work should focus on further refining these techniques and exploring their applications in broader contexts to ensure the safety and longevity of bridge infrastructure.

Data Availability Statement

Some or all data, models, or code that support the findings of this paper are available from the corresponding author upon reasonable request.

Acknowledgments

This research was conducted with the support of the National Research Foundation of Korea (NRF) grant funded by the Ministry of Education (No. 2018R1A6A1A08025348) and the National R&D Project for Smart Construction Technology (No. RS-2020-KA156488) funded by the Korea Agency for Infrastructure Technology Advancement under the Ministry of Land, Infrastructure and Transport, and managed by the Korea Expressway Corporation.

References

Acikgoz, S. K., K. Soga, and J. Woodhams. 2017. "Evaluation of the response of a vaulted masonry structure to differential settlements using point cloud data and limit analyses." *Constr. Build. Mater.* 150 (Sep): 916–931. https://doi.org/10.1016/j.conbuildmat.2017.05.075.

Alqutami, T. 2020. "dji_ros_simulator." Accessed April 18, 2024. https://github.com/TareqAlqutami/dji_ros_simulator.

Armeni, I., O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese. 2016. "3d semantic parsing of large-scale indoor spaces." In *Proc., IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 1534–1543. New York: IEEE.

ASCE. 2021. "2021 Report card for America's infrastructure." Accessed March 13, 2024. https://infrastructurereportcard.org/cat-item/bridges-infrastructure/.

Behley, J., M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall. 2019. "Semantickitti: A dataset for semantic scene understanding of lidar sequences." In *Proc., IEEE/CVF International Conf., Computer Vision*, 9297–9307. New York: IEEE.

CERIK (Construction and Economy Research Institute of Korea). 2023. "CERIK highlight." Accessed March 13, 2024. http://www.cerik.re.kr/report/highlight/detail/2734.

Cha, G., S. Park, and T. Oh. 2019. "A terrestrial LiDAR-based detection of shape deformation for maintenance of bridge structures." *J. Constr. Eng. Manage.* 145 (12): 04019075. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001701.

Charron, N., E. McLaughlin, S. Phillips, K. Goorts, S. Narasimhan, and S. L. Waslander. 2019. "Automated bridge inspection using mobile ground robotics." *J. Struct. Eng.* 145 (11): 04019137. https://doi.org/10.1061/(ASCE)ST.1943-541X.0002404.

chennuo0125. 2020. "lidar_imu_calib." Accessed October 20, 2024. https://github.com/chennuo0125-HIT/lidar_imu_calib.

Costin, A., A. Adibfar, H. Hu, and S. S. Chen. 2019. "Building Information Modeling (BIM) for transportation infrastructure–Literature review, applications, challenges, and recommendations." *Autom. Constr.* 94 (Oct): 257–281. https://doi.org/10.1016/j.autcon.2018.07.001.

Dataspeed. 2015. "velodyne_simulator." Accessed April 18, 2024. https://bitbucket.org/DataspeedInc/velodyne_simulator.

Eastman, C. M., C. Eastman, P. Teicholz, and R. Sacks. 2011. *BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractor.* Chichester, UK: Wiley.

Esfahani, M. E., C. Rausch, M. M. Sharif, Q. Chen, C. Hass, and B. T. Adey. 2021. "Quantitative investigation on the accuracy and precision of Scan-to-BIM under different modelling scenarios." *Autom. Constr.* 126 (Jun): 103686. https://doi.org/10.1016/j.autcon.2021.103686.

Gao, B., Y. Pan, C. Li, S. Geng, and H. Zhao. 2021. "Are we hungry for 3D LiDAR data for semantic segmentation? A survey of datasets and methods." *IEEE Trans. Intell. Transp. Syst.* 23 (7): 6063–6081. https://doi.org/10.1109/TITS.2021.3076844.

Gargoum, S. A., L. Karsten, K. El-Basyouny, and J. C. Koch. 2018. "Automated assessment of vertical clearance on highways scanned using mobile LiDAR technology." *Autom. Constr.* 95 (Nov): 260–274. https://doi.org/10.1016/j.autcon.2018.08.015.

Girardeau-Montaut, D. 2016. "CloudCompare." Accessed April 12, 2024. https://www.cloudcompare.org.

Hackel, T., N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys. 2017. "Semantic3D.net: A new large-scale point cloud classification benchmark." Preprint, submitted April 12, 2017. http://arxiv.org/abs/1704.03847.

Hackl, J., B. T. Adey, M. Woźniak, and O. Schümperlin. 2018. "Use of unmanned aerial vehicle photogrammetry to obtain topographical information to improve bridge risk assessment." *J. Infrastruct. Syst.* 24 (1): 04017041. https://doi.org/10.1061/(ASCE)IS.1943-555X.0000393.

Hajjan, H., and B. Becerik-Gerber. 2010. "Scan to BIM: Factors affecting operational and computational errors and productivity loss." In *Proc., Int. Symp. Automation and Robotics in Construction*, 25–27. Montréal: International Association for Automation and Robotics in Construction. https://doi.org/10.22260/ISARC2010/0028.

Hu, F., J. Zhao, Y. Huang, and H. Li. 2021. "Structure-aware 3D reconstruction for cable-stayed bridges: A learning-based method." *Comput. Aided Civ. Infrastruct. Eng.* 36 (1): 89–108. https://doi.org/10.1111/mice.12568.

Hu, Q., B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, and A. Markham. 2020. "Randla-net: Efficient semantic segmentation of large-scale point clouds." In *Proc., IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 11108–11117. New York: IEEE.

Huthwohl, P., R. Lu, and I. Brilakis. 2016. "Challenges of bridge maintenance inspection." In *Proc., International Conf., Computing in Civil and Building Engineering*, 51–58. Reston, VA: ASCE.

Jing, Y., B. Sheil, and S. Acikgoz. 2022. "Segmentation of large-scale masonry arch bridge point clouds with a synthetic simulator and the BridgeNet neural network." *Autom. Constr.* 142 (Oct): 104459. https://doi.org/10.1016/j.autcon.2022.104459.

Jing, Y., B. Sheil, and S. Acikgoz. 2024a. "A lightweight transformer-based neural network for large-scale masonry arch bridge point cloud segmentation." *Comput.-Aided Civ. Infrastruct. Eng.* 39 (16): 2427–2438. https://doi.org/10.1111/mice.13201.

Jing, Y., J. X. Zhong, B. Sheil, and S. Acikgoz. 2024b. "Anomaly detection of cracks in synthetic masonry arch bridge point clouds using fast point feature histograms and patchcore." *Autom. Constr.* 168 (Dec): 105766. https://doi.org/10.1016/j.autcon.2024.105766.

Khaloo, A., D. Lattanzi, K. Cunningham, R. Dell'Andrea, and M. Riley. 2018. "Unmanned aerial vehicle inspection of the Placer River Trail Bridge through image-based 3D modelling." *Struct. Infrastruct. Eng.* 14 (1): 124–136. https://doi.org/10.1080/15732479.2017.1330891.

Kim, H., J. Yoon, and S. H. Sim. 2020. "Automated bridge component recognition from point clouds using deep learning." *Struct. Control Health Monit.* 27 (9): e2591. https://doi.org/10.1002/stc.2591.

Kim, J., D. Chung, Y. Kim, and H. Kim. 2022. "Deep learning-based 3D reconstruction of scaffolds using a robot dog." *Autom. Constr.* 134 (Feb): 104092. https://doi.org/10.1016/j.autcon.2021.104092.

Korus, K., T. Czerniawski, and M. Salamak. 2023. "Visual programming simulator for producing realistic labeled point clouds from digital infrastructure models." *Autom. Constr.* 156 (Dec): 105126. https://doi.org/10.1016/j.autcon.2023.105126.

Lamas, D., A. Justo, M. Soilan, and B. Riveiro. 2024. "Automated production of synthetic point clouds of truss bridges for semantic and instance segmentation using deep learning models." *Autom. Constr.* 158 (Feb): 105176. https://doi.org/10.1016/j.autcon.2023.105176.

Landrieu, L., and M. Simonovsky. 2018. "Large-scale point cloud semantic segmentation with superpoint graphs." In *Proc., IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 4558–4567. New York: IEEE.

Lee, J. S., J. Park, and Y. M. Ryu. 2021. "Semantic segmentation of bridge components based on hierarchical point cloud model." *Autom. Constr.* 130 (Oct): 103847. https://doi.org/10.1016/j.autcon.2021.103847.

Lin, J. J., A. Ibrahim, S. Sarwade, and M. Golparvar-Fard. 2021. "Bridge inspection with aerial robots: Automating the entire pipeline of visual data capture, 3D mapping, defect detection, analysis, and reporting." *J. Comput. Civ. Eng.* 35 (2): 04020064. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000954.

Liu, W., and S. Chen. 2013. "Reliability analysis of bridge evaluations based on 3D light detection and ranging data." *Struct. Control Health Monit.* 20 (12): 1397–1409. https://doi.org/10.1002/stc.1533.

Lu, R., I. Brilakis, and C. R. Middleton. 2019. "Detection of structural components in point clouds of existing RC bridges." *Comput.-Aided Civ. Infrastruct. Eng.* 34 (3): 191–212. https://doi.org/10.1111/mice.12407.

Mandirola, M., C. Casarotti, S. Peloso, I. Lanese, E. Brunesi, and I. Senaldi. 2022. "Use of UAS for damage inspection and assessment of bridge infrastructures." *Int. J. Disaster Risk Reduct.* 72 (Apr): 102824. https://doi.org/10.1016/j.ijdrr.2022.102824.

Marzouk, M., and M. Hisham. 2012. "Bridge information modeling in sustainable bridge management." In *Proc., Integrating Sustainability Practices in the Construction Industry*, 457–466. Reston, VA: ASCE. https://doi.org/10.1061/41204(426)57.

Maturana, D., and S. Scherer. 2015. "Voxnet: A 3D convolutional neural network for real-time object recognition." In *Proc., IEEE/RSJ Conf. Intelligent Robots and Systems*, 922–928. New York: IEEE. https://doi.org/10.1109/IROS.2015.7353481.

Mehranfar, M., H. Arefi, and F. Alidoost. 2018. "Knowledge-based 3D reconstruction of bridge structures using UAV-based photogrammetric point cloud." *J. Appl. Remote Sens.* 15 (4): 044503. https://doi.org/10.1117/1.JRS.15.044503.

Narazaki, Y., V. Hoskere, G. Chowdhary, and B. F. Spencer Jr. 2022. "Vision-based navigation planning for autonomous post-earthquake inspection of reinforced concrete railway viaducts using unmanned aerial vehicles." *Autom. Constr.* 137 (May): 104214. https://doi.org/10.1016/j.autcon.2022.104214.

NYCDOT (New York City DOT). 2019. "2019 annual bridge and tunnel condition reports." Accessed March 13, 2024. https://www.nyc.gov/html/dot/downloads/pdf/dot_bridgereport19.pdf.

Perry, B. J., Y. Guo, R. Atadero, and J. W. van de Lindt. 2020. "Streamlined bridge inspection system utilizing unmanned aerial vehicles (UAVs) and machine learning." *Measurement* 164 (Nov): 108048. https://doi.org/10.1016/j.measurement.2020.108048.

Qi, C. R., H. Su, K. Mo, and L. J. Guibas. 2017. "Pointnet: Deep learning on point sets for 3D classification and segmentation." In *Proc., IEEE Conf. Computer Vision and Pattern Recognition*, 652–660. New York: IEEE.

Rashidi, A., and E. Karan. 2018. "Video to BrIM: Automated 3D as-built documentation of bridges." *J. Perform. Constr. Facil.* 32 (3): 1–11. https://doi.org/10.1061/(ASCE)CF.1943-5509.0001163.

Riveiro, B., M. J. DeJong, and B. Conde. 2016. "Automated processing of large point clouds for structural health monitoring of masonry arch bridges." *Autom. Constr.* 72 (3): 258–268. https://doi.org/10.1016/j.autcon.2016.02.009.

Robert, D., H. Raguet, and L. Landrieu. 2023. "Efficient 3D semantic segmentation with superpoint transformer." In *Proc., IEEE/CVF Int. Conf., Computer Vision*, 17195–17204. New York: IEEE.

Shan, T., B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus. 2020. "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping." In *Proc., IEEE/RSJ Conf. Intelligent Robots and Systems*, 5135–5142. New York: IEEE. https://doi.org/10.1109/IROS45743.2020.9341176.

Shim, C., N. Yun, and H. Song. 2011. "Application of 3D bridge information modeling to design and construction of bridges." *Procedia Eng.* 14 (Jan): 95–99. https://doi.org/10.1016/j.proeng.2011.07.010.

Stałowska, P., C. Suchocki, and M. Rutkowska. 2022. "Crack detection in building walls based on geometric and radiometric point cloud information." *Autom. Constr.* 134 (Feb): 104065. https://doi.org/10.1016/j.autcon.2021.104065.

Trias, A., Y. Yu, J. Gong, and F. L. Moon. 2022. "Supporting quantitative structural assessment of highway bridges through the use of LiDAR scanning." *Struct. Infrastruct. Eng.* 18 (6): 824–835. https://doi.org/10.1080/15732479.2021.1880446.

Truong-Hong, L., and R. Lindenbergh. 2020. "Extracting bridge components from a laser scanning point cloud." In *Proc., Int. Conf., on Computing in Civil and Building Engineering*, 721–739. Reston, VA: ASCE.

Varney, N., V. K. Asari, and Q. Graehling. 2020. "DALES: A large-scale aerial LiDAR data set for semantic segmentation." In *Proc., IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 186–187. New York: IEEE.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. 2017. "Attention is all you need." *Proc. Adv. Neural Inf. Process. Syst.* 30: 6000–6010.https://doi.org/10.48550/arXiv.1706.03762.

Wang, X., C. Demartino, Y. Narazaki, G. Monti, and B. F. Spencer Jr. 2023. "Rapid seismic risk assessment of bridges using UAV aerial photogrammetry." *Eng. Struct.* 279 (Mar): 115589. https://doi.org/10.1016/j.engstruct.2023.115589.

Wu, B., A. Wan, X. Yue, and K. Keutzer. 2018. "Squeezeseg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D lidar point cloud." In *Proc., IEEE Int. Conf., on Robotics and Automation*, 1887–1893. New York: IEEE.

Wu, B., X. Zhou, S. Zhao, X. Yue, and K. Keutzer. 2019. "Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud." In *Proc., Int. Conf., on Robotics and Automation*, 4376–4382. New York: IEEE. https://doi.org/10.1109/ICRA.2019.8793495.

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

Xiao, J. L., J. S. Fan, Y. F. Liu, B. L. Li, and J. G. Nie. 2024. "Region of interest (ROI) extraction and crack detection for UAV-based bridge inspection using point cloud segmentation and 3D-to-2D projection." *Autom Constr.* 158 (Feb): 105226. https://doi.org/10.1016/j.autcon.2023.105226.

Xu, Y., and J. Zhang. 2022. "UAV-based bridge geometric shape measurement using automatic bridge component detection and distributed multi-view reconstruction." *Autom. Constr.* 140 (Aug): 104376. https://doi.org/10.1016/j.autcon.2022.104376.

Yan, Y., B. Guldur, and J. F. Hajjar. 2017. "Automated structural modelling of bridges from laser scanning." In *Proc., Structures Congress*, 457–468. Reston, VA: ASCE. https://doi.org/10.1061/9780784480403.039.

Yan, Y., and J. F. Hajjar. 2021. "Automated extraction of structural elements in steel girder bridges from laser point clouds." *Autom. Constr.* 125 (May): 103582. https://doi.org/10.1016/j.autcon.2021.103582.

Yan, Y., and J. F. Hajjar. 2022. "Geometric models from laser scanning data for superstructure components of steel girder bridges." *Autom. Constr.* 142 (Oct): 104484. https://doi.org/10.1016/j.autcon.2022.104484.

Yan, Y., Z. Mao, J. Wu, T. Padir, and J. F. Hajjar. 2021. "Towards automated detection and quantification of concrete cracks using integrated images and lidar data from unmanned aerial vehicles." *Struct. Control Health Monit.* 28 (8): e2757. https://doi.org/10.1002/stc.2757.

Yang, L., Y. C. Lin, H. Cai, and A. Habib. 2024. "From scans to parametric BIM: An enhanced framework using synthetic data augmentation and parametric modeling for highway bridges." *J. Comput. Civ. Eng.* 38 (3): 04024008. https://doi.org/10.1061/JCCEE5.CPENG-5640.

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

Yang, X., E. del Rey Castillo, Y. Zou, L. Wotherspoon, and Y. Tan. 2022. "Automated semantic segmentation of bridge components from large-scale point clouds using a weighted superpoint graph." *Autom. Constr.* 142 (Oct): 104519. https://doi.org/10.1016/j.autcon.2022.104519.

Zhao, H., L. Jiang, J. Jia, P. H. Torr, and V. Koltun. 2021. "Point transformer." In *Proc., IEEE/CVF Int. Conf., on Computer Vision*, 16259–16268. New York: IEEE.