Knowledge-enhanced Point Cloud Classification for Bridge Components

Tao Yang[1], Yang Zou[1,2], Enrique del Rey Castillo[1], Weiwei Chen[3] and Ajay Kumar Agrawal[1]

[1]Department of Civil and Environmental Engineering, University of Auckland, Auckland 1010, New Zealand
[2]School of Architecture, Building and Civil Engineering, Loughborough University, Loughborough LE11 3TU, UK
[3]Bartlett School of Sustainable Construction, University College London, UK
tyan632@aucklanduni.ac.nz, y.zou2@lboro.ac.uk, e.delrey@auckland.ac.nz, weiwei.chen@ucl.ac.uk, aagr657@aucklanduni.ac.nz

Abstract –

Classification of three-dimensional (3D) point clouds is fundamental to semantic labelling and scene understanding in bridge inspection and management. However, accurate classification of bridge components remains challenging due to their complex geometries and high inter-class geometric similarity. To address these challenges, this study explores the potential of incorporating prior knowledge into a bridge point cloud classification model. Motivated by the distinctive spatial distribution of bridge components, we introduce a bridge-wise batching training strategy that treats all clusters within each bridge as a single training unit. The distribution of various components in the vertical direction is designed as a hard constraint to guide both training and inference. The proposed approach was validated on a bridge dataset covering four representative Reinforced Concrete (RC) girder bridge types. The results show that integrating prior knowledge consistently outperforms the PointNet baselines, yielding higher classification accuracy and more stable convergence during training. Overall, the findings highlight the effectiveness and promise of integrating prior knowledge into the point cloud classification task of bridge components.

Keywords –

Bridge point cloud; Prior knowledge; Point cloud classification; Physics-informed; Structural knowledge

1 Introduction

Point cloud data has been increasingly adopted in bridge engineering due to its superiority in capturing as-is conditions with high spatial fidelity and resolution [1]. Generated by Light Detection and Ranging (LiDAR) or photogrammetry, point clouds preserve fine-scale shape, enable precise measurement without meshing, and naturally support change detection through repeat acquisitions [2]. Thus, point clouds have been widely utilised for various tasks across the bridge lifecycle, including Quality Assurance (QA), construction progress monitoring, and damage assessment [3].

A variety of learning-based tasks have been proposed to facilitate 3D understanding of point clouds, including point cloud semantic segmentation, instance segmentation, classification, and point cloud completion [1,4,5]. Among them, point cloud classification has been widely used to assign semantic meaning to 3D data and enable higher-level scene understanding. By mapping unstructured point sets to semantic categories, classification facilitates a wide range of downstream tasks across multiple domains, including urban mapping, Building Information Modelling (BIM), industrial Facility Management (FM), and autonomous inspection.

Point cloud classification is particularly critical for semantic labelling in bridge Scan-to-BIM. Specifically, traditional geometry-based methods often rely on point cloud clustering or region growing to partition raw scans, followed by heuristic or feature-based classification to identify structural elements [6,7]. More recently, emerging foundation models and vision language models, such as the Segment Anything Model (SAM), have been explored for bridge point cloud pre-segmentation by operating on projected images or videos [8]; however, these approaches typically generate over-segmented or unlabelled point cloud clusters and still require reliable point cloud classification for semantic assignment. Although previous research has developed classification methods for a small number of component types, these methods tend to introduce higher uncertainty when applied to multi-component classification. No existing efforts have been found that could assign semantic labels reliably for bridges with diverse component types.

To address this challenge, this study proposes a prior knowledge-enhanced point cloud classification strategy for RC girder bridges with various component types. Building on existing deep learning architectures, we introduce a bridge-wise batching training strategy that treats all clusters within each bridge as a single training unit, rather than a collection of independent patches from different bridges. Based on bridge-wise batching, prior knowledge is incorporated as explicit constraints to guide both model training and prediction. After embedding prior knowledge into the classification model, it can produce highly accurate classifications of multi-component types, thereby facilitating downstream tasks.

2 Literature review

2.1 Point cloud classification

Point cloud classification aims to assign semantic labels to 3D samples represented as unstructured point sets, enabling 3D scene understanding in Euclidean space [9]. Compared with images defined on regular pixel grids, point clouds are irregular and unordered, which makes feature extraction more challenging. In recent years, deep learning has become the dominant method for point cloud classification. Deep learning-based approaches can be broadly grouped into (i) voxel-based methods that rasterise points onto sparse 3D grids for 3D convolutions, (ii) multi-view methods that render point clouds into 2D images for Convolutional Neural Network (CNN) processing, and (iii) point-based methods that learn features directly from raw point sets [10]. Although voxel- and projection-based strategies benefit from mature convolutional operators, converting from 3D to grid/image representations may result in the loss of critical 3D spatial information. Consequently, point-based methods have become the mainstream strategy for point cloud segmentation and classification.

Representative point-based architectures include PointNet [11], along with subsequent advances such as graph-based message passing [15] and attention mechanisms [12]. With the growing adoption of point cloud data in infrastructure scenes, learning-based point cloud classification has been widely applied to label a variety of infrastructure objects, such as bridge damage [13], road components [14] and building materials [15]. Accurate point cloud classification, therefore, plays a crucial role in infrastructure-level 3D scene understanding, facilitating downstream applications such as Scan-to-BIM integration and infrastructure condition assessment.

2.2 Physics-guided neural networks

In recent years, Machine Learning (ML) has shifted from purely data-driven paradigms toward knowledge-data dual-driven methods, in which prior knowledge is explicitly encoded into data-driven models to improve robustness, accuracy, and reliability. These approaches are often referred to as physics-guided neural networks (PgNNs) [16]. By integrating prior knowledge and mathematical models into network training, PgNNs not only minimise data-driven loss but also enforce feasibility and consistency constraints imposed by known physics [17]. When such prior knowledge is formalised as governing differential equations, such as ordinary differential equations (ODEs) or partial differential equations (PDEs), together with associated boundary and initial conditions and embedded into the loss function, the resulting framework is commonly termed physics-informed neural networks (PINNs) [16].

The infrastructure domain offers rich priors and well-validated physical laws; accordingly, PgNNs and PINNs have been increasingly applied across the infrastructure life cycle, such as design, post-design analysis, construction, and maintenance/structural health monitoring. Representative studies span physics-based surrogate modelling for early-stage structural design [18], physics-consistent response inference for real bridge systems using limited measurements [19], and physics-informed damage identification and long-term response prediction for bridges [20].

Overall, existing research suggests that embedding prior knowledge into neural networks can substantially enhance data efficiency, interpretability, and generalisation, enabling learning models to tackle complex and nonlinear engineering problems in the real world.

3 Methodology

3.1 Training strategy

This section presents a knowledge-enhanced training strategy for point cloud classification of bridge components, in which prior knowledge information is encoded as hard, rule-based constraints during both training and inference. Conventional approaches typically obtain batches for each training and testing by randomly sampling instances from different bridges. However, these point cloud instances are sampled from different bridges and randomly contain different bridge component types, posing a challenge for leveraging topology-aware priors in the same bridges.

To address this, we propose bridge-wise training and inference via bridge-wise batching: each optimisation step uses all clusters from one bridge, with an adaptive batch size equal to that bridge's number of clusters. This batching strategy enables these constraints, derived from prior knowledge, to be effectively imposed on these components in the same bridges. To validate the proposed training strategy and the effectiveness of prior knowledge, the pioneering point-based learning model, PointNet [11], is adopted for its established reputation and superior performance in point cloud classification. The main framework for the proposed bridge-wise point cloud classification model with rule-based constraints is illustrated in Figure 1, which includes the strategy of utilising rule-based constraints in both the training and inference stages.

3.1.1 Bridge-wise training

As shown in Figure 1(a), for each iteration, all point cloud clusters from a single bridge are grouped into one batch and passed through PointNet to extract point-wise features. These features are then fed into fully connected layers to produce a score matrix $S \in \mathrm{R}^{C \times K}$, where C is the number of classes and K is the number of clusters in the bridge. Based on the score matrix, prior knowledge is compiled into a hard mask $M \in \{0, 1\}^{C \times K}$, as defined by Equation (1), where zeros mark infeasible prediction results based on the pre-defined prior knowledge (e.g., piers must have a component above). Before computing the loss, infeasible entries are masked by mapping them to -∞ and cross-entropy is evaluated on the masked scores $\widetilde{S}$. This forces gradients to flow only through prior-consistent constraints, preventing the network from spending capacity on structurally impossible labels.

$$\widetilde{S}_{ck} = \begin{cases} S_{ck}, & M_{ck} = 1 \\ -\infty, & M_{ck} = 0 \end{cases} \tag{1}$$

3.1.2 Bridge-wise inference

As shown in Figure 1(b), inference follows the same bridge-wise setting: all clusters of a bridge are processed together. For cluster i, the network outputs raw scores $S_{i\text{-}k}$. Then, the hard masks are applied to the raw scores, marking infeasible predictions as **False**. The final label is chosen from the prior-admissible set, as defined in Equation (2). This two-stage logic yields predictions that are simultaneously data-driven and structurally admissible, preventing violations of the pre-defined priors and propagating topological consistency across all clusters within the same bridge.

$$\hat{y}_{i\text{-}k} = \arg \max_{C:\, M_{ck}=1} \widetilde{S}_{i\text{-}k} \tag{2}$$

3.2 Elevation-based hierarchical priors

Bridge structures exhibit a strong vertical layout characteristic due to their mechanical design [7], as illustrated in Figure 2. For example, components with lower location, such as piers and abutments, transfer gravity and other loads to the ground. Pier caps, if present, connect the piers and girders and transmit loads from the superstructure to the piers. Within the superstructure, the components also follow a consistent bottom-to-top order, from girders to decks to parapets, while diaphragms, if present, are distributed between the girders.

To leverage this observation for point cloud classification, we encode this knowledge as Elevation-based Hierarchical Priors (EHP). For each point cloud cluster, we compute a relative elevation to capture its vertical position. Based on the observed distribution of bridge components along the vertical axis, we then define several elevation bands. For each band, we specify a subset of component types that are admissible at that height. This prior is implemented as a hard mask on the logits, so classes that violate the hierarchy are removed from the feasible set during both training and inference.

To achieve this, for each point cloud instance in raw point coordinates, such as instance i, we first compute a robust vertical bounding box by taking lower and upper quantiles of the z-coordinates (Z_i), $Z^i_{\min}$ and $Z^i_{\max}$, as defined by Equation (3). The instance-level elevation is then represented by the midpoint Z^i_{mid}.

$$\begin{cases} Z^i_{\min} = \mathrm{quantile}\left(Z_i, q_{\mathrm{low}}\right) \\ Z^i_{\max} = \mathrm{quantile}\left(Z_i, q_{\mathrm{high}}\right) \\ Z^i_{\mathrm{mid}} = \dfrac{1}{2}\left(Z^i_{\min} + Z^i_{\max}\right) \end{cases} \tag{3}$$

Because the model is trained with bridge-wise batching, we further normalise elevations at the bridge level. For each bridge, we compute the bottom level $Z_{\min}$ and top level $Z_{\max}$ by taking the minimum of all $Z^i_{\min}$ and the maximum of all $Z^i_{\max}$ across the whole bridge, as defined by Equation (4). Then, the effective vertical range (Z_{r}) of bridges can be calculated.

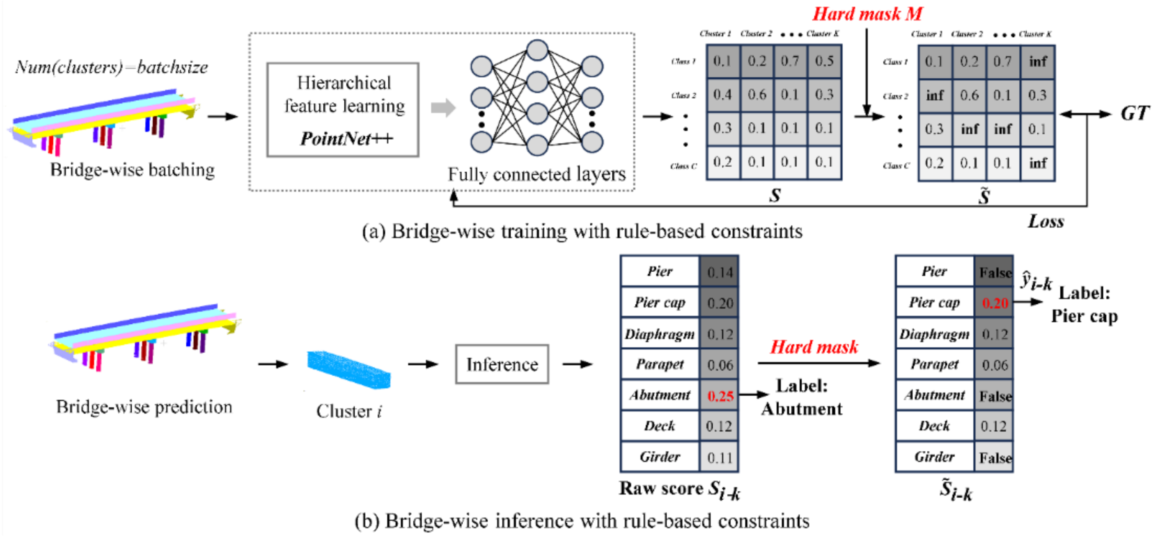

Figure 1. Bridge-wise point cloud classification model with rule-based constraints, (a) bridge-wise training and (b) bridge-wise inference.

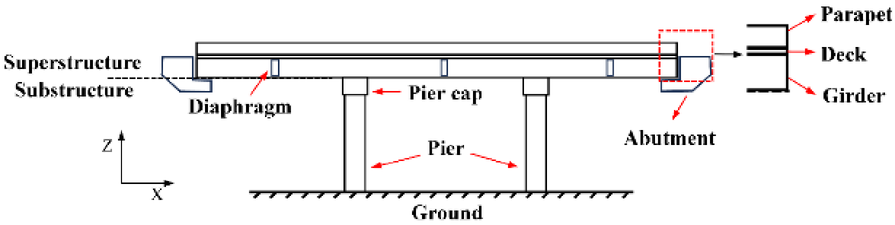

Figure 2. Component distribution of typical RC girder bridges in the vertical direction.

$$\begin{cases} Z_{\min} = \min\limits_{i} Z^{i}_{\min} \\ Z_{\max} = \max\limits_{i} Z^{i}_{\max} \\ Z_{r} = Z_{\max} - Z_{\min} \end{cases} \tag{4}$$

For each cluster i, we define a relative $Z^{i}_{\mathrm{ele}} \in [0,1]$ by normalising the midpoint Z^{i}_{mid} within this bridge, as defined in Equation (5). This normalisation removes absolute elevation and makes the prior comparable across different bridge structures.

$$Z^{i}_{\mathrm{ele}} = \frac{Z^{i}_{\mathrm{mid}} - Z_{\min}}{Z_{r}} \in [0,1] \tag{5}$$

Based on the observed distribution of components in the vertical direction, we then define a small number of elevation bands. For each band, we specify a subset of component types that are admissible at that height, and implement this prior as constraints during model training and testing. Table 1 summarises the band definition. The vertical range is partitioned into four bands using thresholds (Z_{lo}, Z_{mid} and Z_{hi}), corresponding to "bottom", "lower-middle", "upper-middle", and "top" regions. Each band is matched to the component types that typically exist at that height range, consistent with the normal vertical arrangement of the bridge substructure and superstructure. Specifically, the bottom band allows only piers, indicating that the primary substructure occupies the lowest elevations. The lower-middle band allows for abutment, pier, and pier cap, representing the transition from the foundation/substructure to the pier caps. The upper-middle band is a broader superstructure zone that permits abutment, pier cap, girder, diaphragm, deck, and parapet. The top band is restricted to the deck and parapet, which typically occupy the highest elevations.

Table 1. Definition of vertical band for elevation-based prior constraints.

Bands	Definition	Permitted components
Bottom	$Z^{i}_{\mathrm{ele}} < Z_{\mathrm{lo}}$	Pier
Lower-middle	$Z_{\mathrm{lo}} \le Z^{i}_{\mathrm{ele}} < Z_{\mathrm{mid}}$	Pier, pier cap, abutment

Upper-middle	$Z_{\mathrm{mid}} \leq Z^{i}_{\mathrm{ele}} < Z_{\mathrm{hi}}$	Abutment, pier cap, girder, deck, diaphragm, parapet
Top	$Z_{\mathrm{hi}} \leq Z^{i}_{\mathrm{ele}}$	Deck, parapet

In this study, the three thresholds Z_{lo}, Z_{mid} and Z_{hi} are set to 0.3, 0.7, and 0.9, respectively, based on a statistical analysis of the normalised vertical positions of components in our bridge dataset. These values should be regarded as empirical reference thresholds rather than universal constants. They can be updated based on statistical analyses of more typical RC girder bridges in the real world. More importantly, when defining the four elevation bands, we deliberately chose relatively broad and soft intervals to accommodate the complexity and variability of real bridge design, rather than introducing many narrow bands that would attempt to separate every component type, which could harm the model's generalisability.

4 Experiment and results

4.1 Data preparation

The dataset used in this study comprises 34 bridge point clouds, covering slab-beam bridges, single or multi-box girder bridges, and T-girder bridges. Among them, 24 bridge point clouds are generated from 3D BIM models, while ten bridge point clouds are obtained from real-world laser scanning data provided by Lu et al. [21]. As the real scanned data primarily consist of slab-beam bridges, additional synthetic models are incorporated to enhance structural diversity within the dataset, including 8 T-girder bridges, 8 multi-box girder bridges, and 8 slab/single-box girder bridges. For synthetic point clouds from 3D BIM models, the Heidelberg LiDAR Operations Simulator (Helios++) [22] is adopted to generate more realistic bridge point clouds. All point clouds are first aligned along the longitudinal axis via principal component analysis (PCA) and subsequently levelled using the deck as a reference plane. After point cloud generation, each bridge point cloud is manually annotated into different component point clouds for model training and prediction.

4.2 Implementation details

All experiments were conducted in PyTorch on Ubuntu 20.04, utilising an Intel Xeon Gold 6242R CPU and a single NVIDIA Quadro RTX 5000 (16 GB). We evaluated 34 bridges with a 7-fold cross-validation. Different RC girder bridges were randomly assigned to folds, enabling each fold to contain all four RC bridge types and real-world bridge data. Each model was trained for 100 epochs with bridge-wise batching. The input point number was set to 4096, and seven bridge component types were considered in the classification results. Bridge-level dynamic class weight was utilised to mitigate class imbalance. Data augmentation, following standard point cloud practice, was applied, including random point dropout, isotropic scaling, and translation.

4.3 Evaluation metrics

Performance was evaluated using overall accuracy (OA) and mean class accuracy (mAcc), as defined by Equations (6) and (7).

$$\mathrm{OA} = \frac{\sum_{c=1}^{C} TP_c}{N} \tag{6}$$

$$\mathrm{mAcc} = \frac{1}{C}\sum_{c=1}^{C}\frac{TP_c}{N_c} \tag{7}$$

where C is the number of classes and N is the total number of labelled instances. TP_c is the component that is correctly labelled with class c.

To assess both performance and training stability, we evaluate OA and mAcc over epochs using two criteria: $\mathrm{BestAcc}_{\mathrm{last30}}$ (the highest test accuracy in the last 30 epochs) and $\mathrm{WorstAcc}_{\mathrm{last30}}$ (the lowest test accuracy in the last 30 epochs). Focusing on the final 30 epochs ensures the model has sufficiently trained and provides a reliable estimate of the learned representation for classification.

4.4 Results and evaluation

Figure 3 shows the average training accuracy of PointNet with or without prior knowledge for seven-fold cross-validation, including training accuracy (left), test OA (middle), and test mAcc (right). Across all views, models augmented with prior knowledge consistently achieve higher accuracy and reach high accuracy earlier than the baseline models. Importantly, the baselines exhibit significant fluctuations, even in later epochs, indicating unstable convergence. This behaviour is largely due to the presence of many repeated components in bridge point clouds. Once a specific component type is misclassified, the error tends to propagate across numerous similar instances, leading to significant fluctuations in overall performance. After introducing prior knowledge, the constraints effectively improve the classification results, and the training curves become steadier, with consistent high accuracy.

Furthermore, Table 2 and Table 3 present the quantitative evaluation results under seven-fold cross-validation. For the $\mathrm{BestAcc}_{\mathrm{last30}}$ (peak capability), across the seven folds, introducing prior knowledge yields clear gains in peak performance. Specifically, our model reaches 100% OA in four of the seven folds. The average $\mathrm{BestAcc}_{\mathrm{last30}}$ increases from 96.0% OA and 94.0% mAcc

(Baseline) to 99.3% OA and 98.6% mAcc (Ours).

For $\mathrm{WorstAcc}_{\mathrm{last30}}$ (stability and convergence), the stability benefits are more noticeable. The average $\mathrm{WorstAcc}_{\mathrm{last30}}$ rises from 85.7% OA and 78.0% mAcc (Baseline) to 97.4% OA and 94.5% mAcc (Ours). The improvements are especially large on the difficult fold (e.g., Fold 1, the OA increased from 74.1% to 96.4%).

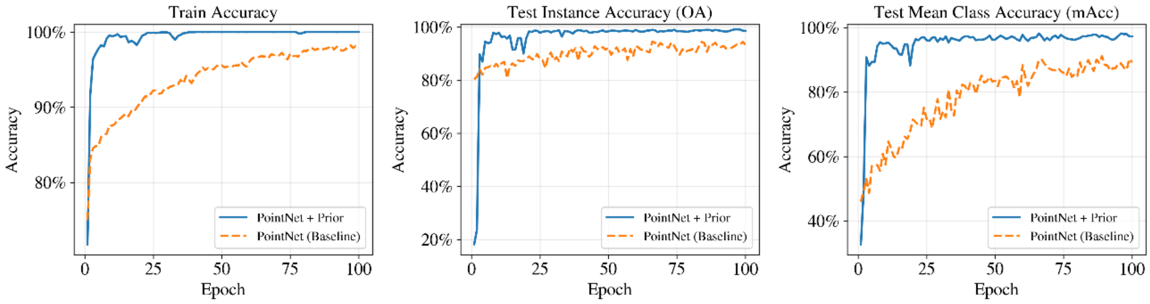

Figure 3. Comparison of PointNet (top) with or without prior knowledge, including training accuracy (left), test OA (middle), and test mAcc (right).

Table 2. $\mathrm{BestAcc}_{\mathrm{last30}}$ under different folds.

Fold		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Avg.
Baseline	OA	90.6	98.9	96.9	96.5	100.0	90.0	99.4	96.0
	mAcc	80.4	97.0	91.1	97.3	100.0	93.0	99.5	94.0
Our	OA	97.8	99.6	97.9	100.0	100.0	100.0	100.0	99.3
	mAcc	96.1	98.6	95.6	100.0	100.0	100.0	100.0	98.6

Table 3. $\mathrm{WorstAcc}_{\mathrm{last30}}$ under different folds.

Fold		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Avg.
Baseline	OA	74.1	96.4	91.4	83.9	78.4	80.0	96.0	85.7
	mAcc	67.5	80.0	75.9	82.4	81.9	77.3	81.1	78.0
Our	OA	96.4	98.2	95.4	100.0	98.1	95.7	98.3	97.4
	mAcc	93.3	92.9	88.7	100.0	95.7	94.6	96.0	94.5

Table 4 further reports per-class accuracies, and Figure 4 display the corresponding confusion matrices for the best and worst epochs over the last 30 epochs.

It can be observed that the PointNet baseline exhibits noticeable fluctuations for deck (from 91.2% $\mathrm{BestAcc}_{\mathrm{last30}}$ to $\mathrm{WorstAcc}_{\mathrm{last30}}$ 55.9%), girder (from 85.3% $\mathrm{BestAcc}_{\mathrm{last30}}$ to $\mathrm{WorstAcc}_{\mathrm{last30}}$ 64.7%), and abutment (from 84.6% $\mathrm{BestAcc}_{\mathrm{last30}}$ to $\mathrm{WorstAcc}_{\mathrm{last30}}$ 70.8%), indicating unstable convergence during model training.

The confusion matrix in Figure 4 further reflects this instability. For $\mathrm{BestAcc}_{\mathrm{last30}}$, the dominant errors in the PointNet baseline are: abutments are incorrectly labelled as pier caps and diaphragms; piers are incorrectly labelled as abutments and diaphragms; and diaphragms are incorrectly labelled as abutments and piers. For $\mathrm{WorstAcc}_{\mathrm{last30}}$, a similar error pattern is observed, but the misclassification becomes substantially more severe. For example, a markedly larger number of diaphragm samples are wrongly predicted as piers and abutments.

After incorporating prior knowledge into model training, the model achieves 100% $\mathrm{BestAcc}_{\mathrm{last30}}$ for pier, girder, diaphragm, deck, and parapet, and also improves the classification accuracy for abutment (from 84.6% to 95.4%). In the $\mathrm{WorstAcc}_{\mathrm{last30}}$, the model keeps the diaphragm at 100%. However, similar misclassification problems can still be found for abutments and pier caps.

Table 4. Class-wise classification result.

Class		Pier	Pier cap	Abutment	Girder	Diaphragm	Deck	Parapet
$\mathrm{BestAcc}_{\mathrm{last30}}$	baseline	96.7	93.5	84.6	85.3	99.1	91.2	97.1
	Our	100.0	93.5	95.4	100.0	100.0	100.0	100.0
$\mathrm{WorstAcc}_{\mathrm{last30}}$	baseline	92.9	95.2	70.8	64.7	88.4	55.9	91.2
	Our	99.5	91.9	75.4	97.1	100.0	97.1	98.5

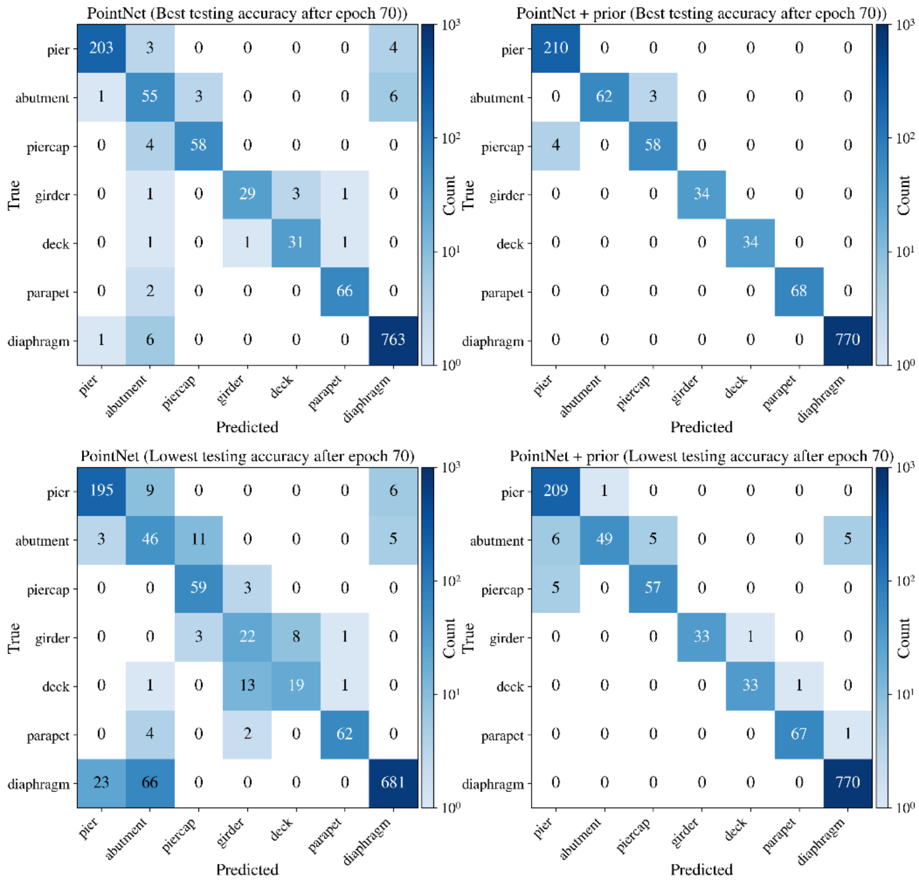

Figure 4. Confusion matrix of classification results of PointNet, including the best testing accuracy and worst testing accuracy in the last 30 epochs.

5 Conclusion

To enhance the point cloud application in bridge scenes, this study proposed a prior knowledge-enhanced point cloud classification model. In our method, we designed a bridge-wise batching training strategy to encode prior knowledge, and then proposed elevation-based priors for model training and prediction. The proposed method was validated using components from 34 RC girder bridge point clouds, covering four typical girder types. The results revealed that introducing prior knowledge yields clear gains in peak performance, as indicated by $BestAcc_{last30}$, and stability for convergence, as indicated by $WorstAcc_{last30}$. The average $BestAcc_{last30}$ improves from 96.0% OA and 94.0% mAcc (PointNet baseline) to 99.3% OA and 98.6% mAcc (Ours).

However, the proposed method is not intended to be a universal solution. Although the bridge-wise batching strategy is potentially applicable to different bridge types, its thresholds still need to be adjusted based on statistical analyses of component distributions for specific bridge categories. Meanwhile, further investigation is still required to validate the applicability of the proposed method to diverse real-world bridge scenarios. In future work, we will focus on: (i) validating the framework on RC girder bridges with more diverse component inventories, while further developing more effective structural priors and refining hyperparameters to improve robustness; and (ii) extending the methodology to other bridge types (e.g., arch and cable-stayed bridges) and infrastructures by designing corresponding structural priors.

References

[1] H. Yue, Q. Wang, H. Zhao, N. Zeng, Y. Tan. Deep learning applications for point clouds in the construction industry. *Automation in Construction*, 168:105769, 2024.

[2] F. Bosché, M. Ahmed, Y. Turkan, C.T. Haas, R. Haas. The value of integrating Scan-to-BIM and Scan-vs-BIM techniques for construction monitoring using laser scanning and BIM: The case of cylindrical MEP components. *Automation in Construction*, 49:201–213, 2015.

[3] Q. Wang, M.K. Kim. Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018. *Advanced Engineering Informatics*, 39:306–319, 2019.

[4] T. Yang, Y. Zou, Enrique del Rey Castillo. ENHANCING SCAN-TO-BIM IN BRIDGE ENGINEERING THROUGH POINT, in: Sixth Int. Conf. Civ. Build. Eng. Informatics, 2025: pp. 85–93.

[5] T. Yang, Y. Zou, R. Castillo, L. Hou, J. Zhong. Enhancing Scan-to-BIM for reinforced concrete bridges using point cloud completion techniques. *Automation in Construction*, 181:106606, 2026.

[6] L. Truong-Hong, R. Lindenbergh. Automatically extracting surfaces of reinforced concrete bridges from terrestrial laser scanning point clouds. *Automation in Construction*, 135:2022.

[7] T. Yang, Y. Zou, X. Yang, E. del Rey Castillo. Domain knowledge-enhanced region growing framework for semantic segmentation of bridge point clouds. *Automation in Construction*, 165:105572, 2024.

[8] N. Cui, H. Chen, X. Guo, Y. Zeng, Z. Hua, G. Xiong, R. Yue, J. Liu. Self-prompting semantic segmentation of bridge point cloud data using a large computer vision model. *Automation in Construction*, 167:105729, 2024.

[9] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, M. Bennamoun. Deep Learning for 3D Point Clouds: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:4338–4364, 2021.

[10] W. Zhou, K. Liu, W. Jin, Q. Wang, Y. She, Y. Yu, C. Ma. Advancements in deep learning for point cloud classification and segmentation: A comprehensive review. *Computers and Graphics*, 130:104238, 2025.

[11] C.R. Qi, H. Su, K. Mo, L.J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation, in: 2017 IEEE Conf. Comput. Vis. Pattern Recognit., IEEE,2017: pp. 77–85.

[12] H. Zhao, L. Jiang, J. Jia, P. Torr, V. Koltun. Point Transformer, in: Proc. IEEE Int. Conf. Comput. Vis., 2021: pp. 16239–16248.

[13] V. Kasireddy, B. Akinci. Assessing the impact of 3D point neighborhood size selection on unsupervised spall classification with 3D bridge point clouds. *Advanced Engineering Informatics*, 52:101624, 2022.

[14] C. Wen, X. Sun, J. Li, C. Wang, Y. Guo, A. Habib. A deep learning framework for road marking extraction, classification and completion from mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 147:178–192, 2019.

[15] L. Yuan, J. Guo, Q. Wang. Automatic classification of common building materials from 3D terrestrial laser scan data. *Automation in Construction*, 110:103017, 2020.

[16] S.A. Faroughi, N.M. Pawar, C. Fernandes, M. Raissi, S. Das, N.K. Kalantari, S.K. Mahjour. Physics-Guided, Physics- Informed, and Physics-Encoded Neural Networks and Operators in Scientific Computing: Fluid and Solid Mechanics. *Journal of Computing and Information Science in Engineering*, 24:1–31, 2024.

[17] A. Momeni, B. Rahmani, B. Scellier, L.G. Wright, P.L. McMahon, C.C. Wanjura, Y. Li, A. Skalli, N.G. Berloff, T. Onodera, I. Oguz, F. Morichetti, P. del Hougne, M. Le Gallo, A. Sebastian, A. Mirhoseini, C. Zhang, D. Marković, D. Brunner, C. Moser, S. Gigan, F. Marquardt, A. Ozcan, J. Grollier, A.J. Liu, D. Psaltis, A. Alù, R. Fleury. Training of physical neural networks. *Nature*, 645:53–61, 2025.

[18] H. Zheng, V. Moosavi, M. Akbarzadeh. Machine learning assisted evaluations in structural design and construction. *Automation in Construction*, 119:103346, 2020.

[19] A.I.F. Al-Adly, P. Kripakaran. Developing physics-informed neural networks for virtual sensing in beams with moving loads. *Engineering Structures*, 338:120535, 2025.

[20] Z. Lai, C. Mylonas, S. Nagarajaiah, E. Chatzi. Structural identification with physics-informed neural ordinary differential equations. *Journal of Sound and Vibration*, 508:116196, 2021.

[21] R. Lu, I. Brilakis. Digital twinning of existing reinforced concrete bridges from labelled point clusters. *Automation in Construction*, 105:102837, 2019.

[22] A.M. Esmoris, M. Yermo, H. Weiser, L. Winiwarter, B. Hofle, F.F. Rivera. Virtual LiDAR simulation as a high performance computing challenge: Towards HPC HELIOS++. *IEEE Access*, 10:1–1, 2022.