Heuristic optimization for digital twin modeling of existing bridges from point cloud data by parametric prototype models

M. Saeed Mafipour[1*], Simon Vilgertshofer[1], and André Borrmann[1]

[1]TUM School of Engineering and Design, Technical University of Munich, Germany.
(Corresponding Author)* ORCID: https://orcid.org/0000-0002-2076-8653,
Email: m.saeed.mafipour@tum.de

ABSTRACT

Digital twins (DTs) can support the operation and maintenance process of bridges by providing a digital model representing the actual asset in reality. The underlying semantic-geometric model of bridges can be created from point cloud data (PCD), obtained by laser scanning or photogrammetry. The bridge PCD, however, needs to be processed and abstracted to a parametric model to handle geometric updates. Today, this process is conducted manually which in turn increases the geometric modeling costs. This paper aims to automate semantic segmentation and parametric modeling as essential steps in the geometric modeling of bridges. The point cloud of bridges is semantically segmented first through a deep-learning model. The value of parameters is then extracted by a heuristic optimization algorithm. Finally, the model of the entire bridge is created. The results of the paper show that the geometric modeling process of bridges can be automated to a large extent through computational methods.

INTRODUCTION

Following the ASCE report card (ASCE, 2021), the number of existing bridges requiring substantial attention is increasing as the rate of deterioration is exceeding the rate of rehabilitation and repair. Despite the feasibility of the conventional approaches for the operation and maintenance of bridges, these methods are loosely supported by digital methods, thus leading to more manual effort and higher costs. Bridge information modeling (BrIM) is a technique to generate the digital representation of bridges based on the physical and functional characteristics of the structure. BrIM can be used in as-designed, as-built, and as-is phases of the structure for accelerated bridge construction, virtual design, structural analysis, and health monitoring (Vilgertshofer et al., 2022). Most recently, Digital Twins (DTs) have been also proposed for bridges to represent the digital counterpart of the asset (Lu et al., 2020, Mafipour et al., 2021). The DT of a bridge is defined a purpose-driven manner based on a set of criteria and requirements. Contrary to BrIM, a DT necessarily requires a link/connection to the existing asset to handle bidirectional updates. This model can even visualize the occurring deteriorations on the body of the structure. These features enable a DT to act as a flexible model to facilitate the operation and maintenance process of bridges (Lu et al., 2020).

Terrestrial laser scanning and photogrammetry are comparatively low-effort techniques to capture existing bridges. Both methods generally result in Point Cloud Data (PCD) representing the partial geometric-semantic information of the bridge. PCD can be considered a resource to create the geometric model of existing bridges. However, it needs to be processed and abstracted further to provide the model fulfilling the requirements of a BrIM or DT. Semantic segmentation and parametric modeling are two primary steps in the geometric modeling process of bridges.

Today, these steps are conducted in a manual manner which is not only labor-intensive but also error-prone. To alleviate the costs associated with the geometric modeling of bridges, this paper aims to propose an approach to automating these steps.

The point cloud of bridges is semantically segmented by training and testing a deep learning model for photogrammetric bridge point clouds. A base model is created and the spatial features of points in different local neighborhoods are described. These features are then processed further and used for classifying points. To create the model of the bridge, parametric prototype models (PPMs) are proposed to extract the dimensions of the bridge elements from the PCD. PPMs are instantiated with random values in ranges inspired by bridge engineering knowledge, and a metaheuristic algorithm is used to fit each PPM into the PCD. The extracted values are then used to create the 3D model of the entire bridge. The paper describes all steps in detail and presents a case study comprising six concrete bridges in Bavaria, Germany.

BACKGROUND AND RELATED WORKS

The proposed approaches for the geometric digital twinning of bridges can be categorized as bottom-up, top-down, and deep learning-based ones. Here, a general overview of these approaches is presented:

Bottom-up: The bottom-up methods start from low-level features such as the x, y, z coordinates of points to cover or generate a system at higher levels. The low-level features can be directly used or converted to high-level features such as normals. The bottom algorithms generally initialize a set of seed points and employ low-level or high-level features to expand the local neighborhood. The bottom-up algorithms have been principally or partially used in the geometric modeling of bridges. Truong-Hong and Lindenbergh (2022) proposed a bottom-up algorithm based on voxel-based region growing (VRG) to detect planer surfaces and elements. Qin et al. (2021) computed and vertical and horizontal density of points to detect the point clusters of bridge elements. Lee et al. (2020) detected and measured the distance between planar surfaces of bridge decks to model specific types of decks.

Top-down: The top-down approach starts with the entire system and decomposes it to subordinate elements. Most of the existing algorithms in this category are heuristic. These algorithms generally leverage the existing geometric and semantic information for classifying points. This approach has been also used for the geometric digital twinning of bridges. Lu et al. (2019) expressed the existing geometric relationships between the points, and segmented the point cloud of RC bridges based on a set of criteria. Girardet and Boton (2021) proposed an approach using a visual programming tool to foster the modeling process of bridges. Pan et al. (2019) constructed a graph based on the spatial relationships between the points and classified points using a rule-based algorithm. Yan and Hajjar (2021) employed the existing connection rules to segment the steel-concrete composite bridges based on a top-down approach.

Deep learning-based: Deep learning (DL) models can receive a PCD containing low-level or high-level features and extract further features for automated semantic segmentation of point clouds. Most of the proposed DL models are supervised and require a dataset for training. However, the trained models can provide more flexibility, especially facing point clouds that might not meet presumptions. DL models have been also used for the geometric modeling of bridges. Hu et al. (2021) extracted features from photogrammetric data (images) through a multi-view convolutional neural network (CNN) and connected it to a multilayer perceptron (MLP) to segment the point cloud of bridges. Lee et al. (2021) collected the features from the local neighborhoods of points through a k-d tree and KNN search algorithm and improved the performance of PointNet (Qi et al., 2017) and dynamic graph CNN (DGCNN) (Wang et al., 2019). Xia et al. (2022) described a local reference frame for points and described the local neighborhoods through a set of features for the centroid point. They further used the extracted features and classified points.

SEMANTIC SEGMENTATION

Semantic segmentation of a bridge point cloud is the process of associating points with predefined labels representing the bridge elements. Bridges mostly consist of vertical and horizontal elements such as the bridge deck and abutments. To increase awareness of the network about intersecting regions between these elements, the bridge point clouds can be encoded more efficiently. In this section, the encoder of RandLA-Net (Hu et al., 2020) is modified and a modified version of this model is presented.

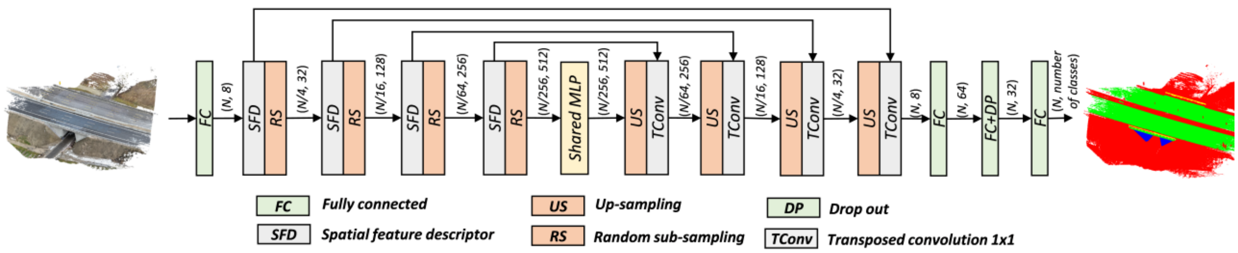
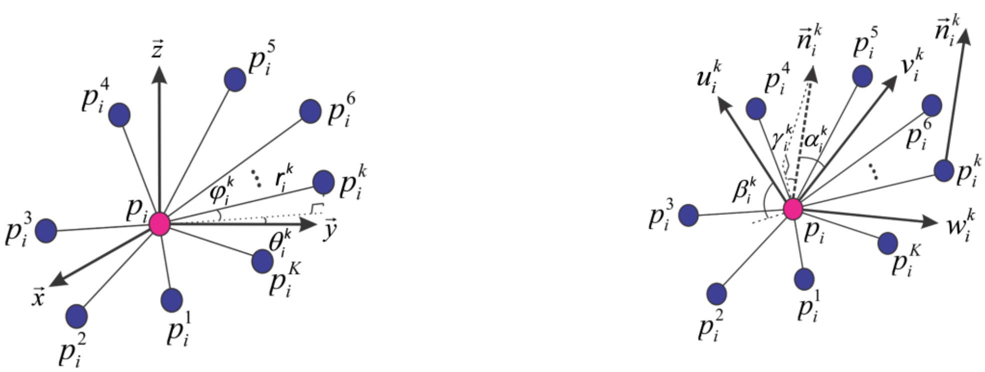


Figure 1. The architecture of Modified RandLA-Net (retrieved from (Hu et al., 2020))

Modified RandLA-Net. RandLA-Net (Hu et al., 2020) benefits from a U-shaped autoencoder as shown in Figure 1. The encoder of this model consists of a spatial encoder followed by random sampling. The spatial encoder describes the relationship between points and the random sub-sampling layer reduces the processing load. To improve the performance of the network, a spatial feature descriptor (SFD) module is used which is similar to the feature aggregation module in RandLA-Net, however, encodes the points through two blocks named Local Spherical Representation (LSpR) and Local Surface Representation (LSuR).



(a) **(b)**

Figure 2. Representation of a local neighborhood: (a) local spherical representation (LSpR); (b) local surface representation (LSuR).

Local Spherical Representation (LSuR). RandLA-Net encodes ten features including the coordinate of the centroid point and its neighboring points, the relative position of the neighboring points to the centroid point, and their distance. Among these features, the coordinate of points is highly vulnerable to possible transformations such as rotation and translation. In addition, relative positions might become very small due to normalization. To address these issues, the relative position of points can be expressed in the spherical coordinate system. Contrary to the Cartesian coordinate system, this system defines data points with two angles limited in the range of $[0, 2\pi]$ and distance/radius from the origin. Figure 2(a) shows the local neighborhood of the query point p_i and its K neighbors $\{p_i^1, p_i^2, \dots, p_i^K\}$ obtained by the K-nearest neighbors (KNN) algorithm. The relative position of p_i to its neighbors can be described as below:

$$r_i^k = \sqrt{x_i^{k^2} + y_i^{k^2} + z_i^{k^2}}, \phi_i^k = arctan(\frac{y_i^k}{x_i^k}), \theta_i^k = arctan(\frac{z_i^k}{\sqrt{x_i^{k^2} + y_i^{k^2} + z_i^{k^2}}}) \quad (1)$$

Local Surface Representation (LSuR). The previous module cannot solely provide adequate information about the underlying surface the points are representing. To enhance the awareness of the network, the Darboux frame of the points can be defined and the curvature, normal curvature, and relative torsion of the surface are encoded implicitly. As shown in Figure 2(b), a Darboux frame with the axes $(u_i^k, \vec{v}_i^k, \vec{w}_i^k)$ can be defined for each pair of (p_i, p_i^k) and the dependencies between each pair (n_i, n_i^k) is described by three angles α_i^k, β_i^k , and γ_i^k as follows (Rusu et al., 2008):

$$\vec{n}_i^k = \vec{n}_i, \qquad \vec{v}_i^k = u_i^k \times \frac{p_i^k - p_i}{\left\|p_i^k - p_i\right\|_2}, \quad \vec{w}_i^k = u_i^k \times \vec{v}_i^k, \quad (2)$$

$$\vec{\alpha}_i^k = \vec{v}_i^k.\vec{n}_i^k, \quad \vec{\beta}_i^k = u_i^k.\frac{p_i^k - p_i}{\left\|p_i^k - p_i\right\|}, \quad \vec{\gamma}_i^k = arctan(\frac{\vec{w}_i^k.\vec{n}_i^k}{u_i^k.\vec{n}_i^k}), \quad (3)$$

LSuR and LSpR modules result in six features, five out of the six expressed by angles. The modified RandLA-Net employs these six features only to encode the local neighborhoods.

Training and validation. The model is trained on the photogrammetric point cloud of six bridges, collected in Bavaria, Germany. To reduce the processing load, each point cloud is sub-sampled by uniform grid sampling with a grid size of 5 cm. The bridge samples consist of a bridge deck, railings, abutments, and background. Thus, these four classes are considered for semantic segmentation. The model is trained on a single GPU (RTX 3080) with 16 GB RAM and its performance is tested on an unseen sample. Due to imbalanced classes, the corresponding weights of the classes are calculated and multiplied by the value of loss resulting from each class. All the point clouds are translated to the origin and normalized in the range of zero to one. The model has four layers with 1/4 sampling in each layer as shown in Figure 1. The number of 16 neighbors is considered for the KNN algorithm and a batch size of two for training. The number of points in each batch is also limited to 60,000 points and a learning rate of 0.001 is used in training.

Semantic segmentation results. To evaluate the performance of the Modified RandLA-Net, it is compared with RandLA-Net (Hu et al., 2020) in terms of accuracy (Acc) and intersection over union (IoU). These statistical metrics are obtained from the confusion matrix of points based on the total number of true positive, false negative, and false positive predictions. Their mean value, i.e. mean Acc (mAcc) and mean IoU (mIoU), also show the overall performance of the model. Table 1 shows the results of training RandLA-Net and its modified version on bridges 2-5 and testing on the unseen sample bridge 1. As can be seen, both models have gained almost the same performance in terms of mAcc while the value of mIoU after applying the modifications in the encoder of RandLA-Net is 3.54% higher. This conveys that the new encoder can improve the prediction results. Figure 3 also shows a visual comparison of the models in which the modified version has been more successful in classifying points.

Table 1. Comparing the prediction results of the models

Model	Metric	Class				Mean (mAcc/mIoU)
		Abutment	Deck	Railing	Background	
RnadLA-Net	Acc	99.81	98.79	98.51	98.13	98.81
	IoU	94.03	96.51	77.69	97.78	91.50
Modified RandLA-Net	Acc	99.92	99.30	97.18	98.79	98.80
	IoU	94.94	97.96	89.24	98.04	95.05

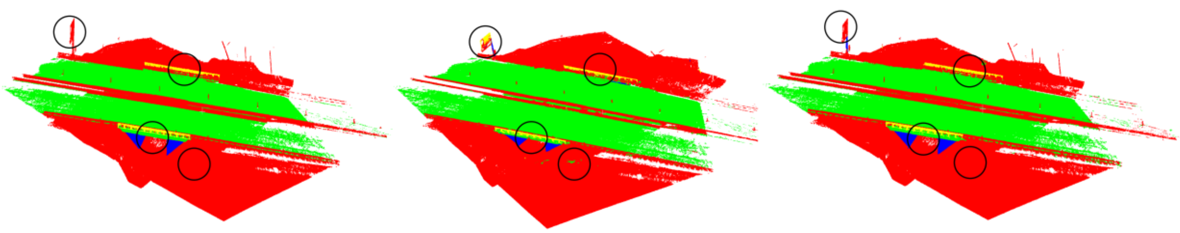


Figure 3. Comparing the prediction results: ground truth (left); RandLA-Net (middle); Modified RandLA-Net (right)

PARAMETRIC MODELING

Parametric modeling allows to create geometrically and topologically consistent and coherent Digital Twins of bridges consisting of several components. It is also necessary to handle the geometric alterations occurring throughout the life-cycle of a bridge. Through parametric modeling, the geometric model becomes capable of changing shape and mirroring the geometric conditions of the real asset. In this section, parametric prototype models (PPMs) are introduced as means for representing engineering knowledge on the shape of individual bridge components in a parametric manner.

Parametric Prototype Model (PPM). A parametric model comprises a set of parameters through which it can be altered. It also comprises a set of constraints that control and preserve the shape of the object. Similarly, a parametric prototype model (PPM) is defined as a dummy model comprising human-definable parameters and constraints with the ability to update its shape. A PPM has a particular class type and objects generated from this class all preserve the type and only differ in attributes such as the value of parameters. A PPM can be defined in 2D or 3D depending on its particular use case. For instance, Figure 4 shows the 2D PPM of a typical bridge deck described by a set of parameters. As can be seen, any change in the value of parameters leads to a new instance of the bridge deck. Considering a point cloud associated with this bridge deck, a list of candidates can be proposed for the value of dimensions the point cloud is representing. To determine the value of parameters through a PPM, each candidate needs to be quantified. To this end, a fitness function is defined and connected to an optimization algorithm, as described in the next section.

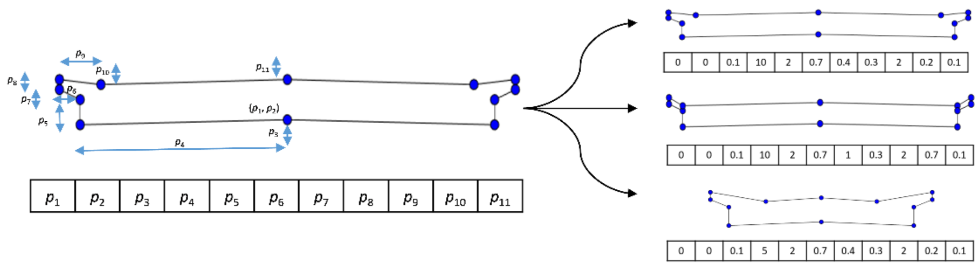


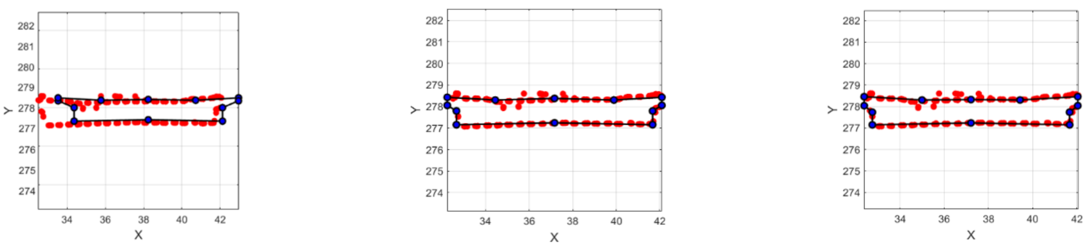
Figure 4. The PPM of a typical bridge deck and its instances.

PPM-to-cloud fitting. Contrary to the conventional model fitting methods, the PPMs pave the way to fitting into not only primitive shapes but also non-primitive shapes. The programming process of a PPM is started from an origin and extended to other vertices based on the value of parameters. Concurrently, constraints such as parallelism, connectivity, perpendicularity, and symmetry are implicitly applied to the prototype model. The mathematical model of the PPM cannot be simply expressed and derived by gradient-based algorithms. Therefore, metaheuristic algorithms can be employed to adjust PPMs and fit them into point clouds. To instantiate a PPM, random values can be generated in predefined ranges derived from bridge engineering knowledge. To fit a PPM, the shortest Euclidean distance of the edges to the point cloud is required to be minimized. Considering a set of points $S = \{s_i \mid i = 1, \ldots, n\}$, where $s_i \in \mathbb{R}^2$, and a 2D PPM described by a set of parameters $P = \{p_j \mid j = 1, \ldots, m\}$ with lower bound l_j and upper bound u_j, in which $p_r \in [l_j, u_j]$, the following objective/fitness function can be defined in the term of the root mean squared error (RMSE):

$$To\ min: F(p_1, p_2, \ldots, p_m) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2}, \qquad Subjected\ to: l_j \le p_j \le u_j \tag{4}$$

where e_j is the shortest distance of the points to the edges and vertices of the PPM. As can be seen, the parameter set P is not directly represented in the objective function of the problem. Therefore, derivatives of the function cannot be calculated simply by gradient-based algorithms. To handle this challenge, metaheuristic algorithms can be employed as they are derivative-free and do not require the closed-form formulation of the fitness function.

Parametric modeling results. The parameter values of a bridge point cloud can be extracted from the faces and cross-sections through the 2D PPMs. To this end, the points of the bridge deck, detected through semantic segmentation, are projected onto the *xy* plane. As the variance of points along the length of the bridge deck is higher, principal component analysis (PCA) can be applied to align the bridge point cloud with the *x*-axis. Using the abutment point clouds, the distance between the retaining walls can be considered as the bridge deck. These points are projected onto the *yz* plane and a 2D PPM resembling the deck is fitted into the points as shown in Figure 5. Similarly, PPMs can be used for extracting the value of parameters from the other segmented point clouds such as abutments. To create the model of the entire bridge, a 3D PPM of the bridge was created that is capable to represent a large variety of single-span RC bridges through parameter variation. As mentioned, PPMs require the parameter values to resemble the point clouds. To this end, the values determined from analyzing the faces and cross-sections through 2D PPMs are fed into this model. To assemble PPMs after model-fitting, a simple average function is applied to the parameters that are common in different elements. This list of values is finally imported into the 3D PPM. The result is a high-quality geometric-semantic digital twin of the captured bridge (Figure 6).



(a) **(b)** **(c)**

Figure 5. PPM-to-cloud fitting: (a) iteration 1; (b) iteration 10; (c) iteration 100

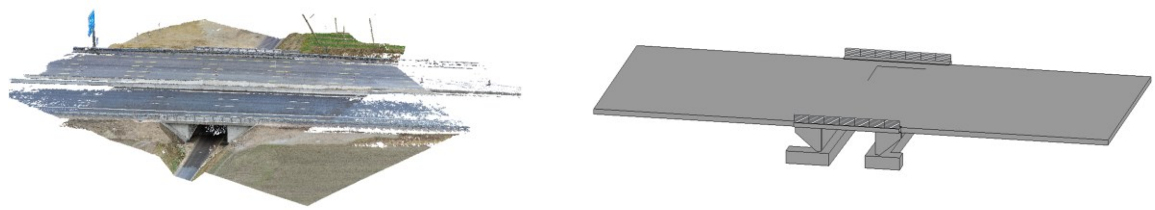


Figure 6. The geometric digital twin of the bridge

CONCLUSION

This paper presents an approach to creating the geometric model of the single-span RC bridges from their point clouds. The point clouds are semantically segmented and these parts are used for extracting the value of parameters. To this end, a deep learning model is employed that encodes the points based on six features mostly defined based on angles. The results of the paper show that these features can aid the network in classifying points. To extract the value of parameters from the segmented parts, Parametric Prototype Models (PPMs) have been proposed. These dummy models are capable of model-fitting into not only primitives but also the point cloud of more complicated elements that exist commonly in bridges. The results of the paper show that the proposed approach can provide a modeling solution that fulfills the industry's demand to a large extent. However, this approach still needs to be tested on more samples and more complicated bridges such as curved or multi-span bridges. Also, more classes can be considered for the semantic segmentation of the bridge point cloud to cover a more variety of bridge types.

ACKNOWLEDGMENT

We thank the German Federal Ministry for Digital and Transport (BMDV) for funding this research in the scope of the TwinGen project.

REFERENCES

ASCE (2021). ASCE's 2021 Infrastructure Report Card.

Girardet, A. and C. Boton (2021). "A parametric BIM approach to foster bridge project design and analysis." *Automation in Construction* **126**: 103679.

Hu, F., J. Zhao, et al. (2021). "Structure-aware 3D reconstruction for cable-stayed bridges: A learning-based method." *Computer-Aided Civil and Infrastructure Engineering* **36**(1): 89-108.

Hu, Q., B. Yang, et al. (2020). Randla-net: Efficient semantic segmentation of large-scale point clouds. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Lee, J. H., J. J. Park, et al. (2020). "Automatic bridge design parameter extraction for scan-to-BIM." *Applied Sciences* **10**(20): 7346.

Lee, J. S., J. Park, et al. (2021). "Semantic segmentation of bridge components based on hierarchical point cloud model." *Automation in Construction* **130**: 103847.

Lu, Q., L. Chen, et al. (2020). "Semi-automatic geometric digital twinning for existing buildings based on images and CAD drawings." *Automation in Construction* **115**: 103183.

Lu, R., I. Brilakis, et al. (2019). "Detection of structural components in point clouds of existing RC bridges." *Computer-Aided Civil and Infrastructure Engineering* **34**(3): 191-212.

Mafipour, M. S., S. Vilgertshofer, et al. (2021). Deriving Digital Twin Models of Existing Bridges from Point Cloud Data Using Parametric Models and Metaheuristic Algorithms. *Proc. of the EG-ICE Conference 2021*.

Pan, Y., Y. Dong, et al. (2019). "Three-dimensional reconstruction of structural surface model of heritage bridges using UAV-based photogrammetric point clouds." *Remote Sensing* **11**(10): 1204.

Qi, C. R., H. Su, et al. (2017). Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition* DOI: http://dx.doi.org/10.1109/CVPR.2017.16.

Qin, G., Y. Zhou, et al. (2021). "Automated Reconstruction of Parametric BIM for Bridge Based on Terrestrial Laser Scanning Data." *Advances in Civil Engineering* **2021** DOI: https://doi.org/10.1155/2021/8899323.

Rusu, R. B., N. Blodow, et al. (2008). Aligning point cloud views using persistent feature histograms. *2008 IEEE/RSJ international conference on intelligent robots and systems*, IEEE.

Truong-Hong, L. and R. Lindenbergh (2022). "Automatically extracting surfaces of reinforced concrete bridges from terrestrial laser scanning point clouds." *Automation in Construction* **135**: 104127.

Vilgertshofer, S., M. Mafipour, et al. (2022). TwinGen: Advanced technologies to automatically generate digital twins for operation and maintenance of existing bridges. *Proc. of European Conference on Product and Process Modeling 2022*.

Wang, Y., Y. Sun, et al. (2019). "Dynamic graph cnn for learning on point clouds." *Acm Transactions On Graphics (tog)* **38**(5): 1-12.

Xia, T., J. Yang, et al. (2022). "Automated semantic segmentation of bridge point cloud based on local descriptor and machine learning." *Automation in Construction* **133**: 103992.

Yan, Y. and J. F. Hajjar (2021). "Automated extraction of structural elements in steel girder bridges from laser point clouds." *Automation in Construction* **125**: 103582.