scientific reports

Check for updates

OPEN

Automatic generation of structural geometric digital twins from point clouds

Kaveh Mirzaei[1], Mehrdad Arashpour[1], Ehsan Asadi[2], Hossein Masoumi[1] & Heng Li[3]✉

A geometric digital twin (gDT) model capable of leveraging acquired 3D geometric data plays a vital role in digitizing the process of structural health monitoring. This study presents a framework for generating and updating digital twins of existing buildings by inferring semantic information from as-is point clouds (gDT's data) acquired regularly from laser scanners (gDT's connection). The information is stored in updatable Building Information Models (BIMs) as gDT's virtual model, and dimensional outputs are extracted for structural health monitoring (gDT's service) of different structural members and shapes (gDT's physical part). First, geometric information, including position and section shape, is obtained from the acquired point cloud using domain-specific contextual knowledge and supervised classification. Then, structural members' function and section family type is inferred from geometric information. Finally, a BIM is automatically generated or updated as the virtual model of an existing facility and incorporated within the gDT for structural health monitoring. Experiments on real-world construction data are performed to illustrate the efficiency and precision of the proposed model for creating as-is gDT of building structural members.

The structural health of building members affects the lives and safety of the public[1]. Monitoring structural health consists of on-site observations, condition evaluation, data management, decision-making, planning, and executing the required repairs[2]. Conventional monitoring methods rely on visual inspection and manual measurements of structural members, which are tedious and error-prone[3]. Also, the effectiveness of procedures depends on the skill and self-discipline of the responsible personnel[4]. Considering a large number of aging buildings and infrastructure projects[5], an automated framework for proactive and accurate structural assessment and health monitoring is of the utmost importance in the Architecture, Engineering, and Construction (AEC) industry[6].

Digital Twins (DTs) have gained a plethora of attention due to improvements in data acquiring technologies, data processing and simulation capabilities, and accessibility of computing infrastructure[7,8]. DT[9–11] is a comprehensive tool for collecting and simulating information with a feedback loop to ensure the coordination of the physical spaces and the digital model of cyberspace during the entire life cycle of a project for reasoning and decision-making[8]. The DT concept has been extensively developed in the manufacturing industry in different life cycle stages such as design stage[8] and configuration stage[12,13] and also for a variety of applications including but not limited to parallel controlling of smart workshop[14], designing of board-type furniture production line[15], and designing of automated flow-shop manufacturing system[16]. In the construction industry, DT has substantially contributed to improvements in applications such as construction quality monitoring and management[17], defect detection in construction projects[18], and construction asset monitoring[19,20]. However, the lack of efficient and accurate algorithms, software, and clearly defined modeling procedures for building structural members hinders DT's development for structural health monitoring purposes[21].

A DT consists of five main parts: (1) physical part; (2) virtual model; (3) connection, which is the device or technique used for obtaining data for integrating virtual and physical spaces[14]; (4) data that is obtained from the physical part; and (5) service, which is the target application of DT[22]. The foremost step and cornerstone of creating an efficient and accurate DT for structural health monitoring purposes is to generate a virtual geometric representation of the as-is asset[23] in a parametric and updatable platform[21], also known as a geometric Digital Twin (gDT)[23,24]. Building Information Model (BIM) has been proven to allow incorporating semantically rich information for a gDT modeling approach throughout the life cycle of a building to help various applications, including building structural health monitoring and maintenance[25]. To adopt BIM as the gDT's virtual model for structural health monitoring purposes, it should be accurate and contain updated information on building

[1]Department of Civil Engineering, Monash University, Melbourne, Australia. [2]School of Engineering, RMIT University, Melbourne, Australia. [3]Department of Building and Real Estate, Hong Kong Polytechnic University, Hung Hom, Hong Kong. ✉email: heng.li@polyu.edu.hk

nature portfolio

1

structural members. However, most buildings' existing information on structural members is out-of-date[26] and primarily stored as 2D drawings in hard copy and/or electronic Computer-Aided Design (CAD) formats[23]. The current approach for creating as-is BIM to be used as gDT's virtual model is to manually obtain information from buildings[21,27] and create/update BIM in modeling and simulation software such as Autodesk Revit, which is tedious, time-consuming, and lacks accuracy due to difficulties associated with accessing structural members in the majority of buildings[28]. Thus, there is a need to develop accurate and efficient methods capable of creating as-is BIM as gDT's virtual model for building structural members.

Recently, the rapid development of surveying and non-contact sensing technologies has improved the accuracy and efficiency of generating gDTs' virtual models of the existing facilities in the format of BIM[21]. Noteworthy examples of such devices (i.e., gDT connection devices) for acquiring accurate data are radio-frequency identification (RFID)[29], 2D camera[30–32], 3D camera[33], and laser scanner[34–36]. Laser scanners have shown a high level of accuracy in capturing data of the physical part in point clouds format (i.e., gDT data part)[37]. Point clouds store 3D geometric information consisting of geometric or geodetic coordinates[38,39]. However, laser scanners cannot obtain semantic information (e.g., object class) required for generating BIMs as gDTs' virtual models and data processing approaches are needed to obtain and infer semantic information.

Xue et al.[40] demonstrated the use of 2D images for developing semantically rich as-built BIMs, in which they categorized semantic information into two main categories of (1) geometric information, such as member position, section shape, and dimensional tolerance, and (2) non-geometric information, such as member function (i.e., beam, column, etc.) and type (i.e., section family type). Geometric information can be obtained directly from point clouds for generating BIMs as gDT's virtual models, non-geometric information should be interpreted and assessed from geometric information.

The first type of geometric information required for creating as-is BIM of structural members as gDT's virtual model is position information consisting of (1) geometric definitions of structural members, such as columns being vertical members, and (2) spatial relationships between structural members, such as bracings positioned under beams[41]. Segmentation algorithms have been used to congregate points with similar geometric features of structural members[41]. Clustering algorithms[42] were widely used to detect similar patterns within a point cloud, also known as clusters based on different features[43], comprising spatial position[44], points normal vector[45], and density of points within point clouds[46]. While research efforts have focused on detecting structural members using position information[41], relying solely on position information is error-prone[47], sensitive to noise in point cloud datasets, and only labels point clusters as potential structural members.

Other research utilized machine learning methods, such as Convolutional Neural Networks (CNNs), to obtain section shape information from point clouds as the second type of geometric information required for creating an as-is virtual model of structural members for gDT creation. Machine learning methods have been deployed in various applications[48,49], such as construction waste detection[50]. However, due to the irregularity, unorderness, and unstructured nature of point cloud data, convolutional operations of CNN methods are incapable of being established[51]. Other works have deployed deep learning methods. The first group of these methods converts the point cloud into a structured grid[52], which is memory-intensive and leads to losing much information[51]. The second group of methods, like PointNet[53], directly applied deep learning to point clouds. PointNet acts as a local feature learner to generate a global point signature by aggregating individual point features. PointNet achieved state-of-the-art performance on a variety of benchmark datasets. Recent works have directly detected structural members[54] and Mechanical, Electrical, and Plumbing (MEP) systems[55] using machine learning-based methods. These methods required large numbers of pre-annotated real-world training datasets similar to their investigated case studies, which opposes a challenge for applying such techniques to other construction projects. Further research utilized slicing methods and image processing approaches to overcome such challenges for detecting section shapes within point cloud datasets[28,56]. The obtained geometric information was used to infer two ending points of the component centerline and the corresponding section shape information for automatically modeling MEP components in a BIM[28]. These research efforts achieved a high detection accuracy due to using matured image processing techniques and methods. However, the performance of such approaches is heavily impacted by the existence of large numbers of non-related objects in point clouds. Also, projecting 2D images to 3D space impacts the overall performance of such approaches.

Non-geometric information, inferred from point clouds, is also vital for creating the gDT's virtual model of structural members. Recent works have inferred two ending points of the component centerline and the corresponding cross-section information for automatically modeling MEP components in a BIM[28]. While applicable for MEP components, the viability of this method for inferring non-geometric information from point clouds and modeling structural members in a BIM remains unproven.

Considering the body of research studied above[35,41,42,56–63], challenges in the automated creation of gDT's virtual model from as-is point clouds for building structural health monitoring can be summarized in four categories: (1) manual processing steps for removing noise and occlusions from point cloud data and manually classifying and detecting structural members, which is not efficient and practical for creating a gDT's virtual model; (2) heavy reliance of previous research on accurate local features in point clouds may not be reliable for real-world point cloud datasets with high levels of noise and occlusions; (3) limited applications of previous research to a specific type of building structure member or a structural section shape, which cannot be generalized to other building types and structural member section shapes; and (4) the need for adopting various technologies that leads to complex data processing seps[23]. Due to the importance of efficiency, accuracy, and generalizability for the automated creation of gDT's virtual model for realizing the target services of gDT, there is a strong need for methods capable of automatically obtaining semantic information from as-is point clouds and transforming such information into BIMs of building structural members[40,64].

Therefore, this study aims to develop an automated, accurate, and generalizable framework for creating gDT of building' structural members including beams, columns, and bracing members (gDT's physical part)

Figure 1. Image of the case study construction projects (**a**) multi-level carpark, and (**b**) Woodside building.

by obtaining and inferring semantic information from point clouds (gDT's data) acquired from laser scanners (gDT's connection) and storing it in updatable BIMs (gDT's virtual model) for structural health monitoring purposes (gDT's service). To reach this aim, we use geometric definitions and spatial relationships between structural members backed up by inputs from standards and regulations for automated detection of potential structural members and filtering out noises and non-structural members. Then, PointNet trained by synthetically generated models is implemented for detecting the section shape of structural members, followed by a general approach for inferring non-geometric information from point clouds using contextual information of building structural members. Finally, an Application Programming Interface (API) in Revit is developed to automatically model the detected structural members in a BIM. Specifically, we aim to answer the following research questions: (Q1) How is geometric information required for generating gDT obtained automatically from raw point cloud data? (Q2) What non-geometric semantic information is inferable from point clouds? (Q3) How can semantic information obtained from point clouds be translated into BIMs as gDT's virtual models? (Q4) How accurate and efficient can the proposed automated point cloud processing method generate the as-is gDTs' virtual model of structural members?

To answer Q1, we adopted a clustering method based on contextual hard-coded knowledge of structural members along with a shape detection method backboned by the PointNet trained by synthetic structural shapes to obtain position and shape information. To answer Q2, we defined a set of conditional rules for detecting each structural member's function and section family type using its geometric information. We answered Q3 by creating a database of semantic information obtained from raw point clouds and connecting that to the database of Revit for creating BIMs. To answer Q4, we evaluated the performance of the proposed method by comparing the obtained dimensions from BIM to the ground truth dimensions and the time required for point cloud processing to the manual method of creating BIMs for two real-world case study construction projects (a) N1 Monash multi-level carpark (hereinafter referred to as multi-level carpark), and (b) Woodside Building for Technology and Design (hereinafter referred to as Woodside building), as shown in Fig. 1.

Results

The results section is divided into obtaining geometric information from point clouds, inferring non-geometric information, and creating BIM steps. The assumptions regarding processing algorithms inputs are limited to the multi-level carpark case study; the Woodside building case study produces similar results.

Obtaining geometric information from point clouds. The first geometric information required for detecting structural members within a point cloud dataset is position information. For that, we adopted contextual hard-coded knowledge and geometric definition of beams, columns, and bracing members. In structural design concepts, beams are usually geometrically defined as horizontal members leading to the congestion of points associated with beams in horizontal slices within point clouds. Therefore, to detect points with the geometric definition of beams (i.e., potential beam points), we sliced the point clouds of case study buildings with horizontal planes. While a higher number of slicing planes results in more accurate results, it would be a computationally heavy process due to the high number of points that should be processed. Considering the proposed section shapes in AS/NZS 3679.1:2016[65–67], the lowest difference between section heights is 20 mm. Thus, the number of horizontal slices should be identified to yield a slice thickness of less than 10 mm (considering both the top and bottom of beams) to obtain the correct section dimensions. Considering the ceiling height of roughly 4 m, we sliced the multi-level carpark case study point cloud with 400 horizontal planes. The average number of points within potential beam slices was 100,000 points and for slices with non-beam objects was 5000 points. Therefore, we extracted potential beam points using the statistical distribution of point numbers within slices. The outcome of this step is a point cloud of building beams. Thus, the MeanShift clustering method was utilized to cluster potential beam points based on the coordinates of each point to identify separate instances of
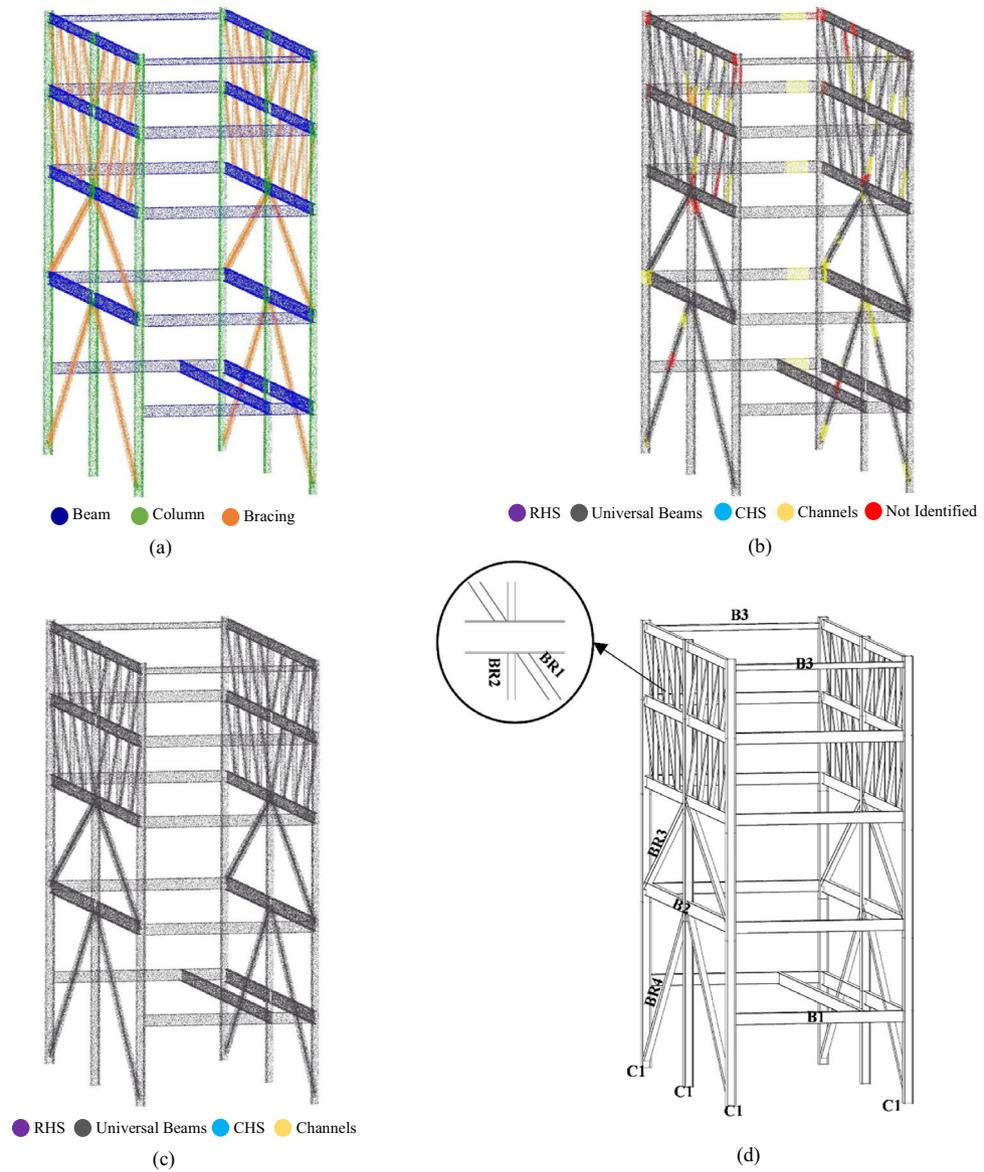
Figure 2. Applying the proposed method to the multi-level carpark case study: (**a**) Obtaining position information for detecting potential structural members. (**b**) Detection of section shapes. (**c**) Homogenization of structural shapes along the length of each member. (**d**) Automatically generated BIM model in Revit.

beams. As for the point cloud investigated in this study, we used the MeanShift clustering function proposed in Scikit-learn with a quantile value of 0.4 and a sample number of 550[68]. The inputs of the MeanShift clustering algorithm were chosen based on the perpendicular distance of neighboring beam point clusters, which ranges from 3 to 6 m in typical buildings. Next, potential column points were geometrically segmented by estimating the point normal in the point cloud with a search radius of 0.7, meaning that points in the 0.7 m radius of each point were used for plane fitting and normal calculation[69]. Potential column points were detected by filtering out the points with an absolute normal vector value in the Z direction (nZ) lower than 0.02, as recommended in AS/NZS 5131:2016[70] for the permissible inclination of columns. Finally, we identified potential bracing members by considering them as members beneath the bounding box of potential beam instances. Figures 2a and 3a illustrate the result of obtaining positional information for detecting potential structural members of multi-level carpark and Woodside building, respectively. Also, Table 1 includes the inputs and thresholds used for finding position information in both case study buildings.

The shape information of object instances found in the previous step is the second geometric information required for detecting structural members within a point cloud dataset. For that, we created a synthetic training dataset consisting of Circular Hollow Sections (CHSs), Universal Beams (UBs), Rectangular Hollow Section (RHSs), and Channels (Cs) for the training of the PointNet network used for shape detection. A total of 6000 synthetic 3D models evenly distributed between different categories were picked for training, and 2000 synthetic 3D models were used for validation. We then converted 3D models into point clouds by evenly distributing a predefined number of points on their surfaces. The predefined number of points for converting synthetic 3D models to point clouds is related to the density of points in the case study point cloud. This number should create a similar point density in the training dataset to the point density in the case study. Therefore, synthetic 3D models were automatically converted to point clouds with 15,000 points to create a point density in the training dataset samples similar to the point density generated by the laser scanner. Also, we applied a random rotation, noise, and occlusion to the training dataset to increase its similarity with the case study point cloud.

We trained the PointNet network with a dropout rate of 0.7, a learning rate of 0.001, and other hyperparameters, as suggested in its original paper[53]. The network training took 200 epochs using a batch size of 32 for converging the loss function, which took 20–21 h. Next, we sliced object instances found in the previous step to make them similar to the training dataset and removed the length of structural members as a feature to increase the performance of the shape detection network. The outcome of the shape detection network is a probability vector. Thus, we utilized the Probability Vector Length (PVL) parameter to evaluate the confidence of the shape detection network using Eq. (1). A high value of PVL demonstrates a high variation between the parameters of the probability vector, meaning that the shape detection had a higher level of confidence for detecting section

4

Figure 3. Applying the proposed method to the Woodside building case study: (**a**) Obtaining position information for detecting potential structural members. (**b**) Detection of section shapes. (**c**) Homogenization of structural shapes along the length of each member. (**d**) Automatically generated BIM model in Revit.

Algorithm input	Structural member	Multi-level carpark	Woodside building
No. of horizontal planes for slicing	Beams	400	400
MeanShift clustering quantile value	Beams	0.4	0.5
MeanShift clustering sample number	Beams	550	550
Normal vector search radius	Columns	0.7	0.8

Table 1. Inputs and thresholds used for finding position information from point clouds.

shapes. The maximum value for the PVL is always equal to one, and the lowest value ($\sigma = 0$) is related to the population of the probability vector. Since four categories of section shapes for classification were defined, the minimum value for PVL, i.e., lowest network confidence for prediction, was 0.5. We set the threshold for the lowest acceptable PVL to be 0.75, and the members with PVL lower than this threshold were labeled as "Not

	Section shape	Section name	Structure member function	x0	y0	z0	x1	y1	z1
1	IPE	310UB40	Beam	−68.57	−9.16	279.16	15.18	−9.16	279.1
2	CHS	CHS273	Bracing	−67.94	−7.06	261.78	−67.94	8.29	268.1
3	IPE	310UB40	Column	−67.90	−9.13	261.81	−67.90	−9.13	278.4
4	IPB	100UC15	Column	−49.55	−2.61	270.91	−49.55	−2.61	278.4
5	IPB	310UC98	Column	−67.80	37.56	260.67	−67.80	37.5	268.7
6	IPE	310UB40	Column	−13.37	−9.15	261.74	−13.37	−9.15	278.4

Table 2. Information obtained from structural members point clouds.

identified". Figures 2b and 3b depict the results obtained from the shape detection network for multi-level carpark and Woodside building point clouds, respectively.

$$PVL = \sqrt{n\sigma^2 + \frac{1}{n}}. \tag{1}$$

Inferring non-geometric information. Structural member function information is the first non-geometric information required to create building structural members' virtual model. We manually labeled the "Not identified" point clouds using CloudCompare software. Then, considering the rarity of having multiple section shapes along structural members, we homogenized the section label along each instance by choosing the section label with the maximum number of points as the section label for the whole object instance, as shown in Figs. 2c and 3c for multi-level carpark and Woodside building point clouds, respectively. Next, we defined structural members as object instances that were geometrically labeled as structural members (beam, column, or bracing) with a structural section shape assigned to them and inferred their function by their geometric label.

The second non-geometric information required for creating a virtual model of structural members is section family type. To infer such information, we created a database consisting of structural member function (i.e., beam, column, bracing), section shape (i.e., UB, CHS, RHS, C), section width and height, member length, and section center points for the start and end of the structural member. Table 2 includes the obtained information from random point cloud examples within the dataset. This information was obtained by investigating the bounding box around each structural member. Moreover, UB cross-sections were divided into IPE and IPB members based on the proportion between section height and width. Also, section names were defined similarly to the default families in Revit to create a connection between the obtained information and BIM model generation software. Finally, x0, y0, and z0 are the coordinate of the center of the bounding box section at the starting location of the structural member, and x1, y1, and z1 are the coordinate of the center of the bounding box section at the end of the structural member.

Creating BIM. For creating a BIM of structural members from point cloud datasets to be used as gDT's virtual model, we developed an API that connects the information stored in Table 2 to the database of Revit. The developed API reads each row of Table 2 for obtaining the required information such as section name, member length, member position, and member function for automatically creating object instances within the Revit, as depicted in Figs. 2d and 3d for multi-level carpark and Woodside building point clouds, respectively. The obtained BIM model is the as-is gDT virtual model of the case study buildings.

Discussion

Two main approaches were proposed for obtaining position and section shape information from point clouds. We utilized structural members' geometric definitions and spatial relationships to get position information. The advantage of the proposed slicing method for geometric segmentation of potential beam points is that it is irrespective of the noise level and non-structural points in the point cloud due to the immersive difference between the number of points in potential beam slices and non-beam slices. This advantage will increase the generality and repeatability of the proposed framework by eradicating the need for manual noise removal and point cloud clearing steps. Also, the proposed method demonstrated a satisfactory performance for detecting potential column points. One of the challenges of the proposed method is finding the optimized parameters for the MeanShift clustering method in cases where the distance between side-by-side beams is not constant, such as the multi-level carpark case study. Overall, using contextual hard-coded knowledge and geometric definition of beams, columns, and bracing members not only segments out potential points belonging to structural members but also improves the performance of other steps by reducing the computational burden for the following steps by decreasing the number of points and also, removing noises and non-structural members without geometric definitions of structural members.

We utilized PointNet and trained it with a synthetically generated dataset and a slicing method for detecting section shape information. The trained network demonstrated satisfactory performance on the classification of the validation dataset with an average accuracy of 96% over different categories, as depicted in Fig. 4. The proposed method for identifying structural members represented a satisfactory performance of 94% accuracy if excluding "Not identified" sections from the results and 89% accuracy if considering "Not-identified" sections as incorrect predictions. The method mostly faced difficulty predicting cross-section shapes that were not fully
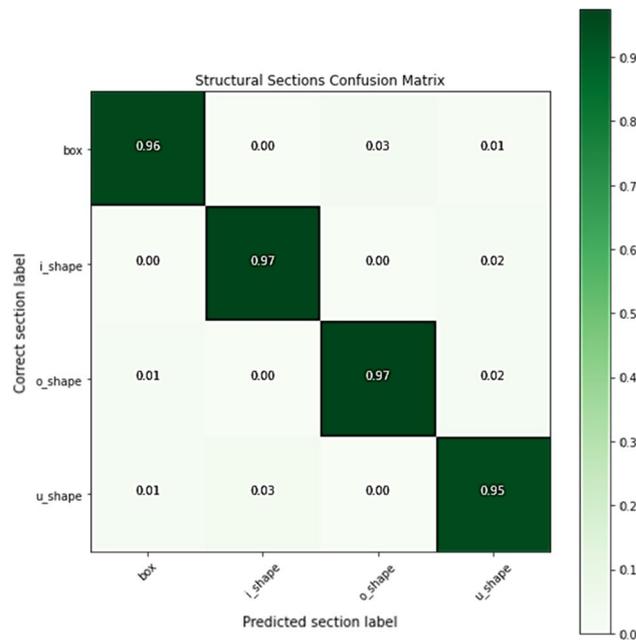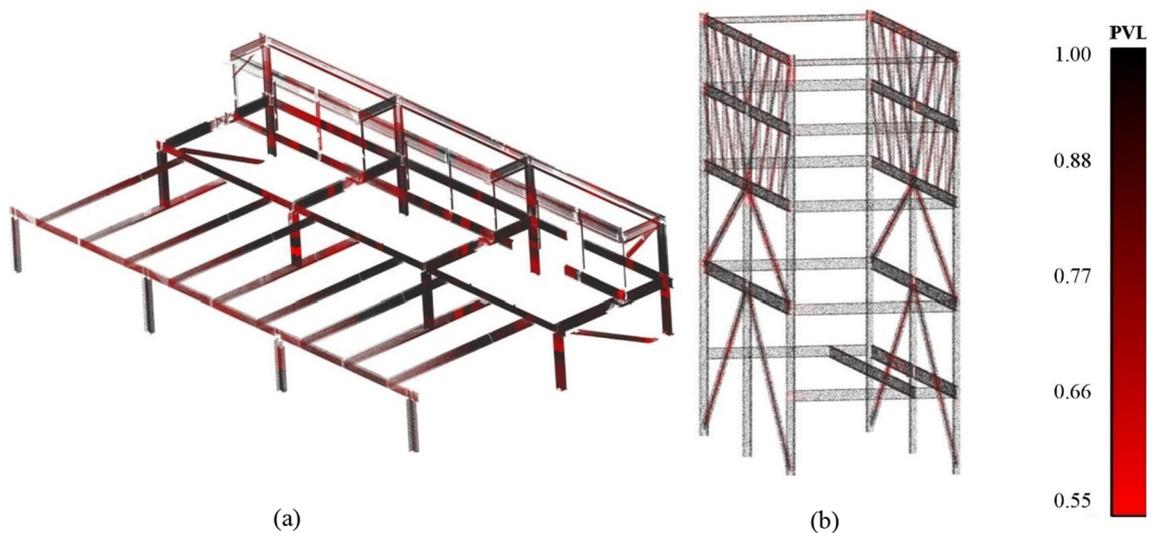
Figure 4. PointNet performance in the validation dataset.

Figure 5. Probability vector length of the classified (**a**) multi-level carpark, and (**b**) Woodside building.

captured during the scanning process. For example, the TLS device had a sharp angle for capturing structural members of the first floor in the multi-level carpark point cloud. Therefore, the point cloud quality was low in those areas, resulting in low prediction performance. Another challenge for the prediction model was detecting the bracing-beam connection members as they were not included in the training dataset, and the classification network correctly labeled them as "Not identified". Also, the probability vector length of the correct prediction is mainly near the maximum value of one, demonstrating the model's high confidence in predicting the correct section shapes, as shown in Fig. 5a,b for multi-level carpark and Woodside building point clouds, respectively.

Based on the results obtained from the section shape information step, the completeness of the point cloud significantly impacts the classification network's performance. A shortcoming noted in the proposed method is that it relies on the similarity of training data and the case study point clouds in terms of noise level and point density. The impact of this phenomenon can be seen in the difference between the performance of detection of structural shapes in the Woodside building point cloud, which has a lower level of noise compared to the multi-level carpark case study as depicted in Fig. 5. Thus, it is suggested that future research focus on developing methods for generating training datasets with a dynamic level of noise and point density. In summary, the proposed method for obtaining geometric information from point clouds demonstrated satisfactory performance in getting position and section shape information, answering this paper's first research question (Q1).

For inferring non-geometric information from point clouds, we applied a set of conditional rules based on the contextual knowledge definitions of building structural members. The function information was found based on the geometric information of the points. We fitted a bounding box around each member and utilized its dimensions for the type information. This method had a satisfactory performance in members without attached non-structural objects, such as MEP systems. The attached non-structural objects increased the bounding box's size, resulting in incorrect size and type predictions in members. Thus, it is suggested that future research focuses on filtering out attached non-structural objects using RGB values before fitting the bounding box. Moreover, the homogenization step had a satisfactory performance for turning the obtained geometric information into construction-related information that can be used for various applications. The homogenization step considers the same weight for all section shape predictions. However, Fig. 5 shows that network confidence in predicting section shape differs in different point cloud slices. Therefore, future research must focus on developing a homogenization method that applies confidence weights to the section slices. We found that structural member function and section family type information can be inferred from point clouds, answering this paper's second research question (Q2).

We used the Revit database to generate the BIM of structural members required for gDT models. The proposed method creates instances of objects in the Revit environment. One of the challenges observed is that object instances automatically created by the proposed API clash in the connection areas and require manual clash detection and removal procedures. Thus, one of the promising research areas for future endeavors is incorporating clash detection in the proposed API in the connection area of structural members. Also, connection types can be identified from the point cloud dataset to increase the level of details of the created BIM. While this paper focuses on generating a gDT of existing buildings through capturing the as-is condition of facilities in point clouds and processing that information offline to create a digital replica, a promising research area for future research is to create an online connection between the physical asset and the digital model with a feedback loop leading to the creation of DTs from gDTs. The online data received from the physical model can maximize the efficiency of DTs by providing a joint optimization decision-making system for the facilities[71]. Also, cyber-physical systems have demonstrated an excellent promise for integrating the virtual and physical worlds[14]. In summary, we found that our proposed method is not only capable of connecting geometric and non-geometric information to Revit's dataset using the developed API for creating a BIM for various object types but also creates a platform for cyber-physical models to be implemented, integrating physical and virtual worlds, answering the third research question (Q3).

Lastly, we evaluated the performance of the proposed method for not only creating gDT's virtual model from point clouds but also the service of health monitoring via comparing the final results obtained with the ground truth information, as shown in Table 3. Also, to evaluate the accuracy of the proposed method for creating BIM of building structural members to be used as gDT virtual models for structural health monitoring purposes, the structural members of the case study building were manually measured five times, and the average value was reported. The responsible measuring personnel reported an average tolerance of ± 25 mm, which is slightly higher than the values reported in previous literature[72,73] due to difficulties in accessing building structure members for manual measurements.

The proposed method demonstrated an error tolerance of ± 24.06 mm from ground truth in multi-level car-park and an error tolerance of ± 24.75 mm from ground truth in the Woodside building for obtaining the length of structural members from the automatically created BIM. Also, the model's accuracy for detecting the length of structural members was calculated using Eq. (2).

$$Accuracy\,(\%) = 100 \times \left(1 - \frac{|Obtained\,dimension - Ground\,truth\,dimension|}{Ground\,truth\,dimension}\right). \tag{2}$$

Finding the correct length of bracing members opposed a challenge for the proposed method as they show higher error tolerance compared to manual measurements, primarily due to the lack of data within the point cloud dataset. Thus, it is recommended that future research focuses on optimizing the scanning locations to fully capture the as-is state of construction and civil infrastructure projects. Also, the proposed method could correctly detect section family types and shapes with an average accuracy of 81.63% in multi-level parking and 86.31% in Woodside building for creating gDT's virtual model. Also, the total processing time of the proposed method was 26 min on average for each case study building, which is generally shorter than traditional manual measurements. Overall, the proposed method promptly demonstrated a lower average error tolerance compared to manual measurement methods, showing its accuracy and efficiency to be utilized as a gDT creation method for structural health monitoring purposes, answering this paper's fourth research question (Q4).

Conclusion

An automatic method for creating BIMs as gDTs' virtual models of building structural members (beams, columns, and bracings) was introduced in this work. The proposed method directly processed unorganized point clouds defined by the coordinates (x, y, z) as data input for gDT, obtained from terrestrial laser scanners, as connection input for DT. First, the method obtained geometric information such as position and section shape from the point cloud dataset using contextual hard-coded knowledge and geometric definition of structural members. The main geometric definitions used are: (1) points corresponding to beams are congested in horizontal slices within the point cloud, (2) points corresponding to columns have a horizontal normal vector, and (3) points associated with bracing members are located beneath beam points.

Then, PointNet, a deep neural network trained by a synthetic dataset, is used to detect the section shape of structural members. Next, structural sections were homogenized for each structural member, assuming that having multiple section shapes along a structural member in a regular building is rare. After obtaining the geometric

Case study building	Member mark	Length (mm)				Section shape detection accuracy		
		Ground truth	Obtained	Variation	Accuracy (%)	No. of correct section predictions	Total no. of structural members in the group	Accuracy (%)
Multi-level carpark	B1	8300	8267	33	99.60	9	9	100
	B2	6000	6012	12	99.80	3	4	75
	B3	25,760	25,750	10	99.96	1	1	100
	B4	25,760	25,769	9	99.97	1	1	100
	B5	25,760	25,780	20	99.92	1	1	100
	B6	25,760	25,733	27	99.90	1	1	100
	B7	2500	2512	12	99.52	3	4	75
	B8	25,760	25,736	24	99.91	1	1	100
	B9	25,760	25,779	19	99.93	1	1	100
	C1	2550	2523	27	98.94	3	4	75
	C2	2550	2536	14	99.45	4	4	100
	C3	5250	5263	13	99.75	3	4	75
	C4	2550	2522	28	98.90	7	10	70
	BR1	6200	6152	48	99.23	1	2	50
	BR2	3500	3435	65	98.14	1	2	50
		Average length accuracy: 99.52				Average section type detection accuracy: 81.63		
Woodside building	B1	9120	9102	18	99.80	7	8	87.5
	B2	23,320	23,305	15	99.94	7	9	77.77
	B3	9120	9142	22	99.76	2	4	50
	C1	24,450	24,440	10	99.96	6	6	100
	BR1	5000	4976	24	99.52	28	32	87.5
	BR2	3450	3429	21	99.39	21	24	87.5
	BR3	12,800	12,751	49	99.62	4	4	100
	BR4	10,200	10,239	39	99.62	3	4	75
		Average length accuracy: 99.7				Average section type detection accuracy: 86.31		

Table 3. Comparison between the obtained dimensions and ground truth data for study building.

information, contextual knowledge and conditional statements were used to infer non-geometric information. Non-geometric information for creating a BIM consists of structural members' function and family type information. Lastly, geometric and non-geometric information was connected to the database of Revit by developing an API for creating a BIM as gDT's virtual model.

The results indicated an error tolerance of ± 24.06 mm from ground truth in multi-level carpark and error tolerance of ± 24.75 mm from ground truth in Woodside building for obtaining the length of structural members from the automatically created gDT virtual model. Also, the proposed method could correctly detect section family types and shapes with an average accuracy of 81.63% in multi-level parking and 86.31% in Woodside building for creating the gDTs' virtual models. Overall, it is concluded that the proposed automatic method for gDT generation using raw point cloud data stands out in terms of accuracy and efficiency compared to both traditional manual methods and previously proposed approaches.

Future work includes developing methods for reducing the time required for obtaining and registering as-is point clouds of construction and infrastructure projects. Also, more categories of construction-related objects, such as structural members' connections, will be added to the method to enhance the proposed approach's generality. Finally, a promising research area for future research is to create an online connection between the physical asset and the digital replica with a feedback loop to integrate virtual and physical spaces using the platform developed in this study.

Materials and methods

The BIM approach for the gDT generation of structural members in construction and civil infrastructure projects from point clouds is summarized into four significant steps: 1. Point cloud acquisition and preprocessing 2. Geometric information obtainment 3. Non-geometric information inference, and 4. BIM generation for creating/ updating the gDT model. The proposed method workflow is visually illustrated in Fig. 6.

Point cloud acquisition and preprocessing. Point cloud data were obtained by a terrestrial laser scanner (FARO® Focus M70) from the interior and exterior environment of (a) multi-level carpark, and (b) Woodside building to validate the performance of the proposed gDT generation method, as shown in Fig. 1. A total of 11 scans from the interior of each case study building with ¼ resolution and 4× quality settings were obtained with the average point number of 3×10^6 points in each scan. The same scan parameters were used for both case study buildings to demonstrate the impact of different noise and point densities on the overall performance of the proposed method. Then, a point cloud registration was performed to merge the obtained scans from the case
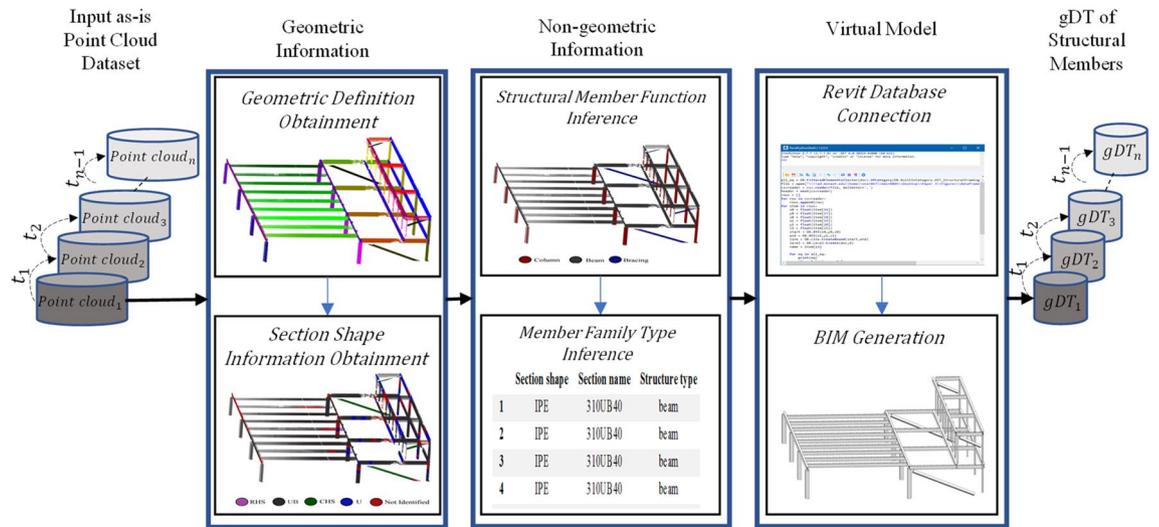
Figure 6. The workflow for the geometric digital twin-driven dimensional quality inspection of building structural members.

Metadata	Multi-level carpark	Woodside building
No. of scans	11	11
No. of points in the point cloud	3,342,352	2,281,412
Scanner model	FARO® Focus M70	FARO® Focus M70
Scanner range	0.6–70 m	0.6–70 m
Ranging errors at 10 m and 25 m	± 1 mm	± 1 mm
Registration error	± 3 mm	± 4 mm

Table 4. Metadata of the case study buildings point cloud datasets.

study building into a complete 3D point cloud using FARO® SCENE Software. The registration took 10–12 h using a desktop computer (Intel i7-9700 CPU @ 3.00 GHz, 32 GB RAM, and 500 GB SSD), resulting in a complete 3D point cloud for each building. Finally, a mixed part of the point cloud from buildings was segmented manually to evaluate the proposed method's performance and reduce computational time. Specifications of the laser surveys for obtaining point clouds of case study buildings are given in Table 4.

The role of a gDT is to be a dynamic digital representation of an asset during different life cycle phases. Thus, a gDT must be updated frequently due to unavoidable changes in the as-is condition of an asset over time (i.e., gDT_1, gDT_2, gDT_3,…, gDT_n). The frequency (i.e., t_1, t_2, t_3,…, t_{n-1}) of capturing as-is data (i.e., point cloud$_1$, point cloud$_2$, point cloud$_3$,…, point cloud$_n$) and updating a gDT model is related to the service component of gDT. For structural health monitoring of buildings, an optimized frequency of capturing as-is data leads to in-time detection and rectifying defects with the right level of resources and inspection costs. Since structural health monitoring of buildings is a soft real-time task, in which the deadlines can be allowed for delays as long as the tasks are timely executed, a risk-based approach should be used to find the optimal frequency of capturing as-is data to update gDT. While the minimum frequency of capturing as-is data for structural health monitoring purposes is stated in standards and regulations for different construction projects[74], an increase in the level of building importance, defect risk, and hazard consequences can further escalate the frequency of capturing as-is data. Therefore, in consultation with designers, the statutory building inspector can set the frequency of capturing as-is data for a soft real-time structural health monitoring of buildings using the framework proposed in Fig. 6.

The preprocessing steps are primarily designed to (1) reduce the computational burden of point cloud processing steps by removing floor and ceiling points within the dataset and (2) align structural members with the point cloud coordinate system. For the first step, Ceiling and floor planes are filtered out by an improved RANSAC algorithm operating on Normal Distribution Transformation (NDT) cells[75] through classifying point clouds into planar and non-planar cells. For aligning the structural members with the point cloud coordinate system, points within the point cloud should be multiplied by a transformation matrix as a vector (x, y, z), as shown in Eq. (3).

$$R_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin\theta & 0 \\ \sin\theta & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where θ is the rotation angle about the Z-axis since the Z-axis is correctly adjusted during the scanning and registration processes. While angle θ can be determined manually using visualization software such as Cloud
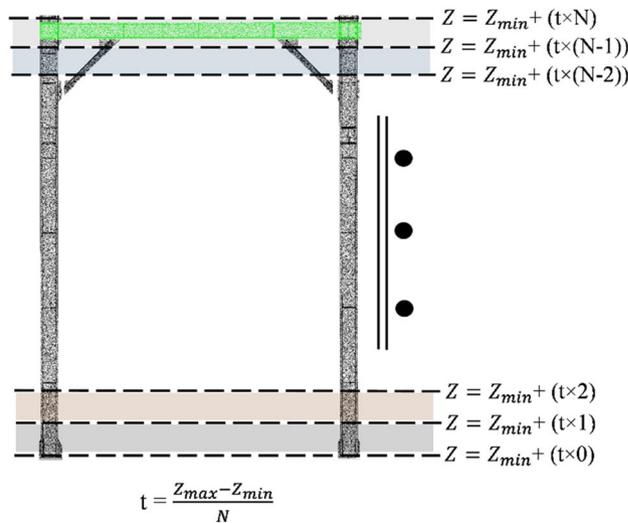
Figure 7. Horizontal slicing of a frame.

Compare, Principal Component Analysis (PCA) method is used for automated and exact identification of angle θ[76].

Geometric information obtainment. Generating structural members' BIM as gDT's virtual model requires geometric and non-geometric information from point clouds. The first group of geometric information is position information. In the position information step, points possessing geometric definitions and relationships of building structural members are segmented. Building beams are defined as horizontal members parallel to support horizontal structures like floors. Thus, it can be geometrically inferred that beam points are congested in horizontal slices within the point cloud dataset. Therefore, a slicing method is applied to create horizontal point cloud slices, as shown in Fig. 7. An optimal value of the horizontal slices can be found by using contextual information of standard structural section dimensions (e.g., AS/NZS 3679.1:2016[65]) as shown in Eq. (4) as

$$\frac{Z_{\max} - Z_{min}}{N} \leq \frac{d_s}{2}, \tag{4}$$

where Z_{max} and Z_{min} are the maximum and minimum Z-axis coordinates, respectively, d_s is the minimum difference between the height of standard structural sections as stated in standards, and N is the number of horizontal slices. After finding the N, the thickness of horizontal slices is calculated using Eq. (5) as

$$t = \frac{Z_{\max} - Z_{min}}{N}, \tag{5}$$

where t is the thickness of the horizontal slices. Next, points are distributed between slices i and $i + 1$ using their Z-axis coordinates as Eq. (6) as

$$Z_{min} + (t \times i) \leq z \leq Z_{min} + (t \times (i + 1)), \tag{6}$$

where $0 \leq i \leq N$. After distributing the points between horizontal slices, the number of points within each horizontal slice is statistically compared to find the slices with a relatively higher density of points. The outcome of this step is beam points associated with each floor of the building. Since point cloud segmentation aims to detect different instances of structural members, the MeanShift clustering algorithm is used to segment the beam points of each floor based on the X and Y coordinates into single instances of beam members. Next, columns will be geometrically detected within the point cloud dataset. Columns are geometrically defined as vertical structural members for transferring compressive force in the building. Points belonging to vertical members within a point cloud dataset can be detected by calculating the normal vector of that point as an array of $[n_x, n_y, n_z]$, in which if the absolute value of n_z is close to zero, that point can be considered a potential column point. The outcome of this step is a group of points that have the geometric definition of a column; therefore, the MeanShift algorithm is used to cluster out different instances of columns based on the X and Y coordinates of points. Lastly, bracings are geometrically defined as diagonal members existing beneath beams in buildings. Thus, points associated with bracings in a point cloud dataset can be geometrically detected as points with the X and Y coordinates within each beam's X and Y boundaries with a Z coordinate less than the minimum Z coordinate of beam points.

The next group of geometric information required for generating gDT of building structural models from a point cloud dataset is section shapes of the geometrically segmented members obtained from the previous step. A supervised classification method is proposed for detecting different section shapes of geometrically segmented

point clouds. For this purpose, PointNet, a deep neural network, is utilized for predicting the section shape of each geometrically segmented point cloud[53]. PointNet uses a symmetric function (i.e., max pooling) that is order-invariant to act as a local feature learner and aggregate local features to acquire global features from each point. Thus, PointNet architecture can deploy multilayer perceptrons without converting point clouds into structured grid datasets for classifying the shape of geometrically segmented members.

PointNet heavily relies on precise and abundant training data as a deep learning method. Thus, a synthetic training dataset of structural shapes is proposed to compensate for the lack of a publicly available dataset of structural section shapes. This dataset consists of point cloud sections automatically generated from 3D shapes. Also, random levels of noises, rotations, and occlusions are applied to the training dataset to increase its similarity to real-world point clouds. While PointNet trained by synthetic dataset is capable of being directly used on geometrically segmented members for classifying point cloud section shapes, the length of geometrically segmented point clouds opposes a problem for the classification network as the network considers the members' length as the main feature of the point cloud instead of its section shape. Thus, prior to applying PointNet for classifying section shapes, geometrically segmented members are sliced into smaller sections to increase the classification network's performance.

The output of the classification network on each point cloud slice is a probability distribution over the set of section categories. Therefore, the confidence of the classification network is inferred by calculating the PVL of each classified point cloud slice using Eq. (1). In Eq. (1), n is the size of the probability vector population, and σ is the standard deviation of the probability vector population. The value of PVL highlights the differences between the probability values in the classification network outcome's probability distribution. A higher value of PVL means a more significant disparity between probability values, showing a higher confidence level for the classification network. Finally, the section label will be added to each point cloud slice as an attribute required for generating a BIM model. Section shapes identified with low confidence are labeled as "Not identified", which the user should manually label.

Non-geometric information inference. Non-geometric information, such as member function and section family type, is necessary for generating the BIM as a gDT virtual model of structural members. However, only geometric information can be directly obtained from point clouds. To cover this gap, contextual knowledge of structural members is used for inferring non-geometric information from geometric information. Member function information is the first non-geometric information required to create a structural members' gDT virtual model. First, the section shape along each structural member should be homogenized since structural members were sliced in the semantic segmentation step, and each can have a different section label. To homogenize section shapes along a structural member, the section shape with the most significant number of points is assigned as the section shape for the whole structural member. A conditional rule is defined for inferring member function from geometric information as "if a point is geometrically labeled as a structural member AND geometrically labeled to be a part of structural section shape, then it can be inferred that the member function of that point is the same as its geometrical definition label". This conditional rule filters out non-structural objects possessing structural members' geometric definitions from the point cloud.

The next group of non-geometric information required for generating BIM as gDT virtual replica of structural members is section family type. This information is inferred by drawing a bounding box around each member to get section size (i.e., section width, height, length) and conditional rules to detect the subcategory of section shapes (i.e., IPE and IPB in the UB category). For labeling the section dimensions, the maximum and minimum values of X, Y, and Z coordinates are subtracted from each other. Their highest value corresponds to the elongation of the structural member and is labeled as member length. As for the other two values, the smaller one is labeled as section width, and the bigger one is labeled as section height. For members with the UB section label, if the height is 50% bigger than the width, that section is labeled as IPE; otherwise, it would be labeled as IPB. To find the start and end point of the section, the coordinate of the center of mass of the section is calculated, which is the same in both starting and endpoints. For the third coordinate, the maximum value of the elongation dimension is the endpoint, and the minimum value is the starting point. Next, the obtained information from each structural member is saved within a database.

BIM generation. Finally, Revit generates the BIM model as gDT's virtual model. The underlying architecture of Revit is a database used to store and share information. Thus, after obtaining the necessary information for generating a BIM model, the obtained database is connected to the Revit database to automate the generation of BIM models. To reach this purpose, first, a list of all of the family symbols loaded in the Revit is acquired by applying the "FilteredElementCollector" command within the Revit dataset. Then, for each structural member, if the obtained section name from the point cloud matches any section names in the Revit database, that section name in the Revit database and its corresponding family symbol would be chosen as the input for the "CreateNewFamilyInstance" command in Revit. Next, the member starting and endpoints saved in the database obtained from labeled point clouds are used for defining the line on which the structural member is drawn in Revit. Finally, the structural function (i.e., beam, column, bracing) read from the point cloud dataset is the final input for creating the new family instance inside Revit. After repeating the procedure for all structural member point clouds, the BIM required for gDT is generated. Table 5 includes the pseudocode for creating BIM from the geometric and non-geometric information obtained from case study buildings' point clouds.

Input The output of "Geometric information obtainment" and "Non-geometric information inference" steps stored in a dataset containing each member's boundary coordination, section shape, section family name, and structural function **Output** BIM in Revit environment as gDT's virtual model	
BIM generation process 1. Select the active document in Revit (doc) 2. Select all of StructuralFraming objects loaded in doc using FilteredElementCollector (all_eq) 3. Load the database file into API (file) 4. Read the database file (csvreader) 5. For row in csvreader do 6. Select the start coordination stored in the database (start) 7. Select the end coordination stored in the database (end) 8. Draw a line between the start and end points (line) 9. Select the floor number stored in the database (level) 10. Extract section family name from the database (name) 11. Extract structure member function from the database (function) 12. For equipment in all_eq do 13. If name==equipemnt's name do 14. Create NewFamilyInstance using line, name, level, function 15. End if 16. End for 17. End for	

Table 5. Pseudocode for automatically creating BIM using information obtained from point clouds.

Data availability

All data generated or analyzed during this study are included in this published article.

References

1. Ma, Z. *et al.* Construction quality management based on a collaborative system using BIM and indoor positioning. *Autom. Constr.* **92**, 35–45. https://doi.org/10.1016/j.autcon.2018.03.027 (2018).
2. Agdas, D., Rice Jennifer, A., Martinez Justin, R. & Lasa Ivan, R. Comparison of visual inspection and structural-health monitoring as bridge condition assessment methods. *J. Perform. Constr. Facil.* **30**, 04015049. https://doi.org/10.1061/(ASCE)CF.1943-5509.0000802 (2016).
3. Phares Brent, M., Washer Glenn, A., Rolander Dennis, D., Graybeal Benjamin, A. & Moore, M. Routine highway bridge inspection condition documentation accuracy and reliability. *J. Bridg. Eng.* **9**, 403–413. https://doi.org/10.1061/(ASCE)1084-0702(2004)9:4(403) (2004).
4. Arashpour, M., Ngo, T. & Li, H. Scene understanding in construction and buildings using image processing methods: A comprehensive review and a case study. *J. Build. Eng.* **33**, 101672. https://doi.org/10.1016/j.jobe.2020.101672 (2021).
5. Tavakolan, M., Mostafazadeh, F., Jalilzadeh Eirdmousa, S., Safari, A. & Mirzaei, K. A parallel computing simulation-based multi-objective optimization framework for economic analysis of building energy retrofit: A case study in Iran. *J. Build. Eng.* **45**, 103485. https://doi.org/10.1016/j.jobe.2021.103485 (2022).
6. Kim, M.-K., Wang, Q. & Li, H. Non-contact sensing based geometric quality assessment of buildings and civil structures: A review. *Autom. Constr.* **100**, 163–179. https://doi.org/10.1016/j.autcon.2019.01.002 (2019).
7. Lim, K. Y. H., Zheng, P. & Chen, C.-H. A state-of-the-art survey of digital twin: Techniques, engineering product lifecycle management and business innovation perspectives. *J. Intell. Manuf.* **31**, 1313–1337. https://doi.org/10.1007/s10845-019-01512-w (2020).
8. Leng, J. *et al.* Digital twins-based smart manufacturing system design in Industry 4.0: A review. *J. Manuf. Syst.* **60**, 119–137. https://doi.org/10.1016/j.jmsy.2021.05.011 (2021).
9. Kahlen, F.-J., Flumerfelt, S. & Alves, A. *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches* 85–113 (Springer, 2017).
10. Huang, H., Yang, L., Wang, Y., Xu, X. & Lu, Y. Digital twin-driven online anomaly detection for an automation system based on edge intelligence. *J. Manuf. Syst.* **59**, 138–150. https://doi.org/10.1016/j.jmsy.2021.02.010 (2021).
11. Wu, L., Leng, J. & Ju, B. Digital twins-based smart design and control of ultra-precision machining: A review. *Symmetry* **13**, 1717–1717. https://doi.org/10.3390/sym13091717 (2021).
12. Leng, J. *et al.* Digital twin-driven rapid reconfiguration of the automated manufacturing system via an open architecture model. *Robot. Comput.-Integr. Manuf.* **63**, 101895. https://doi.org/10.1016/j.rcim.2019.101895 (2020).
13. Leng, J. *et al.* Digital twins-based remote semi-physical commissioning of flow-type smart manufacturing systems. *J. Clean. Prod.* **306**, 127278. https://doi.org/10.1016/j.jclepro.2021.127278 (2021).
14. Leng, J. *et al.* Digital twin-driven manufacturing cyber-physical system for parallel controlling of smart workshop. *J. Ambient. Intell. Humaniz. Comput.* **10**, 1155–1166. https://doi.org/10.1007/s12652-018-0881-5 (2019).
15. Yan, D. *et al.* Digital twin-driven rapid customized design of board-type furniture production line. *J. Comput. Inf. Sci. Eng.* **21**, 4050617. https://doi.org/10.1115/1.4050617 (2021).
16. Liu, Q., Zhang, H., Leng, J. & Chen, X. Digital twin-driven rapid individualised designing of automated flow-shop manufacturing system. *Int. J. Prod. Res.* **57**, 3903–3919. https://doi.org/10.1080/00207543.2018.1471243 (2019).
17. Hamledari, H., Rezazadeh Azar, E. & McCabe, B. IFC-based development of as-built and as-is BIMs using construction and facility inspection data: Site-to-BIM data transfer automation. *J. Comput. Civ. Eng.* **32**, 04017075. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000727 (2018).
18. Gao, T., Ergan, S., Akinci, B. & Garrett, J. Evaluation of different features for matching point clouds to building information models. *J. Comput. Civ. Eng.* **30**, 04014107. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000425 (2016).
19. Howell, S., Rezgui, Y. & Beach, T. Integrating building and urban semantics to empower smart water solutions. *Autom. Constr.* **81**, 434–448. https://doi.org/10.1016/j.autcon.2017.02.004 (2017).

20. Arashpour, M., Kamat, V., Heidarpour, A., Hosseini, M. R. & Gill, P. Computer vision for anatomical analysis of equipment in civil infrastructure projects: Theorizing the development of regression-based deep neural networks. *Autom. Constr.* **137**, 104193. https://doi.org/10.1016/j.autcon.2022.104193 (2022).
21. Jiang, F., Ma, L., Broyd, T. & Chen, K. Digital twin and its implementations in the civil engineering sector. *Autom. Constr.* **130**, 103838. https://doi.org/10.1016/j.autcon.2021.103838 (2021).
22. Tao, F., Zhang, H., Liu, A. & Nee, A. Y. C. Digital twin in industry: State-of-the-art. *IEEE Trans. Ind. Inf.* **15**, 2405–2415. https://doi.org/10.1109/TII.2018.2873186 (2019).
23. Lu, Q., Chen, L., Li, S. & Pitt, M. Semi-automatic geometric digital twinning for existing buildings based on images and CAD drawings. *Autom. Constr.* **115**, 103183. https://doi.org/10.1016/j.autcon.2020.103183 (2020).
24. Lu, R. & Brilakis, I. Digital twinning of existing reinforced concrete bridges from labelled point clusters. *Autom. Constr.* **105**, 102837. https://doi.org/10.1016/j.autcon.2019.102837 (2019).
25. Panah, R. S. & Kioumarsi, M. Application of building information modelling (BIM) in the health monitoring and maintenance process: A systematic review. *Sensors* **21**, 030837. https://doi.org/10.3390/s21030837 (2021).
26. Volk, R., Stengel, J. & Schultmann, F. Building information modeling (BIM) for existing buildings—Literature review and future needs. *Autom. Constr.* **38**, 109–127. https://doi.org/10.1016/j.autcon.2013.10.023 (2014).
27. Mirzaei, K. *et al.* Environmental, social, and economic benefits of buildings energy retrofit projects: A case study in Iran's construction industry. In *Construction Research Congress 2020*, 693–701. https://doi.org/10.1061/9780784482858.075.
28. Wang, B., Yin, C., Luo, H., Cheng, J. C. P. & Wang, Q. Fully automated generation of parametric BIM for MEP scenes based on terrestrial laser scanning data. *Autom. Constr.* **125**, 103615. https://doi.org/10.1016/j.autcon.2021.103615 (2021).
29. Wang, L.-C. Enhancing construction quality inspection and management using RFID technology. *Autom. Constr.* **17**, 467–479. https://doi.org/10.1016/j.autcon.2007.08.005 (2008).
30. Kwon, O.-S., Park, C.-S. & Lim, C.-R. A defect management system for reinforced concrete work utilizing BIM, image-matching and augmented reality. *Autom. Constr.* **46**, 74–81. https://doi.org/10.1016/j.autcon.2014.05.005 (2014).
31. Kardovskyi, Y. & Moon, S. Artificial intelligence quality inspection of steel bars installation by integrating mask R-CNN and stereo vision. *Autom. Constr.* **130**, 103850. https://doi.org/10.1016/j.autcon.2021.103850 (2021).
32. Hua, L., Lu, Y., Deng, J., Shi, Z. & Shen, D. 3D reconstruction of concrete defects using optical laser triangulation and modified spacetime analysis. *Autom. Constr.* **142**, 104469. https://doi.org/10.1016/j.autcon.2022.104469 (2022).
33. Iglesias, C., Martínez, J. & Taboada, J. Automated vision system for quality inspection of slate slabs. *Comput. Ind.* **99**, 119–129. https://doi.org/10.1016/j.compind.2018.03.030 (2018).
34. Kim, M.-K., Wang, Q., Yoon, S. & Sohn, H. A mirror-aided laser scanning system for geometric quality inspection of side surfaces of precast concrete elements. *Measurement* **141**, 420–428. https://doi.org/10.1016/j.measurement.2019.04.060 (2019).
35. Kim, M.-K. *et al.* Automated dimensional quality assurance of full-scale precast concrete elements using laser scanning and BIM. *Autom. Constr.* **72**, 102–114. https://doi.org/10.1016/j.autcon.2016.08.035 (2016).
36. Anil, E. B., Tang, P., Akinci, B. & Huber, D. Deviation analysis method for the assessment of the quality of the as-is building information models generated from point cloud data. *Autom. Constr.* **35**, 507–516. https://doi.org/10.1016/j.autcon.2013.06.003 (2013).
37. Wang, Q., Tan, Y. & Mei, Z. Computational methods of acquisition and processing of 3D point cloud data for construction applications. *Arch. Comput. Methods Eng.* **27**, 479–499. https://doi.org/10.1007/s11831-019-09320-4 (2020).
38. Poullis, C. A framework for automatic modeling from point cloud data. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**, 2563–2575. https://doi.org/10.1109/TPAMI.2013.64 (2013).
39. Hamraz, H., Contreras, M. A. & Zhang, J. Forest understory trees can be segmented accurately within sufficiently dense airborne laser scanning point clouds. *Sci. Rep.* **7**, 6770. https://doi.org/10.1038/s41598-017-07200-0 (2017).
40. Xue, F., Lu, W. & Chen, K. Automatic generation of semantically rich as-built building information models using 2D images: A derivative-free optimization approach. *Comput.-Aided Civ. Infrastruct. Eng.* **33**, 926–942. https://doi.org/10.1111/mice.12378 (2018).
41. Maalek, R., Lichti, D. D. & Ruwanpura, J. Y. Automatic recognition of common structural elements from point clouds for automated progress monitoring and dimensional quality control in reinforced concrete construction. *Remote Sens.* **11**, 02. https://doi.org/10.3390/rs11091102 (2019).
42. Romero-Jarén, R. & Arranz, J. J. Automatic segmentation and classification of BIM elements from point clouds. *Autom. Constr.* **124**, 103576. https://doi.org/10.1016/j.autcon.2021.103576 (2021).
43. Miller, C., Nagy, Z. & Schlueter, A. A review of unsupervised statistical learning and visual analytics techniques applied to performance analysis of non-residential buildings. *Renew. Sustain. Energy Rev.* **81**, 1365–1377. https://doi.org/10.1016/j.rser.2017.05.124 (2018).
44. Aldoma, A. *et al.* Tutorial: Point cloud library: Three-dimensional object recognition and 6 DOF pose estimation. *IEEE Robot. Autom. Mag.* **19**, 80–91. https://doi.org/10.1109/MRA.2012.2206675 (2012).
45. Czerniawski, T., Sankaran, B., Nahangi, M., Haas, C. & Leite, F. 6D DBSCAN-based segmentation of building point clouds for planar object classification. *Autom. Constr.* **88**, 44–58. https://doi.org/10.1016/j.autcon.2017.12.029 (2018).
46. Aljumaily, H., Laefer Debra, F. & Cuadra, D. Urban point cloud mining based on density clustering and MapReduce. *J. Comput. Civ. Eng.* **31**, 04017021. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000674 (2017).
47. Pătrăucean, V. *et al.* State of research in automatic as-built modelling. *Adv. Eng. Inform.* **29**, 162–171. https://doi.org/10.1016/j.aei.2015.01.001 (2015).
48. Mirzaei, K. *et al.* 3D point cloud data processing with machine learning for construction and infrastructure applications: A comprehensive review. *Adv. Eng. Inform.* **51**, 101501. https://doi.org/10.1016/j.aei.2021.101501 (2022).
49. Arashpour, M. *et al.* Predicting individual learning performance using machine-learning hybridized with the teaching-learning-based optimization. *Comput. Appl. Eng. Educ.* https://doi.org/10.1002/cae.22572 (2022).
50. Lu, W., Chen, J. & Xue, F. Using computer vision to recognize composition of construction waste mixtures: A semantic segmentation approach. *Resour. Conserv. Recycl.* **178**, 106022. https://doi.org/10.1016/j.resconrec.2021.106022 (2022).
51. Bello, S. A., Yu, S., Wang, C., Adam, J. M. & Li, J. Review: Deep learning on 3D point clouds. *Remote Sens.* **12**, 1729. https://doi.org/10.3390/rs12111729 (2020).
52. Coudron, I., Puttemans, S., Goedemé, T. & Vandewalle, P. Semantic extraction of permanent structures for the reconstruction of building interiors from point clouds. *Sensors* **20**, 236916. https://doi.org/10.3390/s20236916 (2020).
53. Charles, R. Q., Su, H., Kaichun, M. & Guibas, L. J. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 77–85.
54. Perez-Perez, Y., Golparvar-Fard, M. & El-Rayes, K. Scan2BIM-NET: Deep learning method for segmentation of point clouds for scan-to-BIM. *J. Constr. Eng. Manag.* **147**, 04021107. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002132 (2021).
55. Agapaki, E. & Brilakis, I. CLOI-NET: Class segmentation of industrial facilities' point cloud datasets. *Adv. Eng. Inform.* **45**, 101121. https://doi.org/10.1016/j.aei.2020.101121 (2020).
56. Smith, A. & Sarlo, R. Automated extraction of structural beam lines and connections from point clouds of steel buildings. *Comput.-Aided Civ. Infrastruct. Eng.* **37**, 110–125. https://doi.org/10.1111/mice.12699 (2022).
57. Laefer, D. F. & Truong-Hong, L. Toward automatic generation of 3D steel structures for building information modelling. *Autom. Constr.* **74**, 66–77. https://doi.org/10.1016/j.autcon.2016.11.011 (2017).

58. Zeibak-Shini, R., Sacks, R., Ma, L. & Filin, S. Towards generation of as-damaged BIM models using laser-scanning and as-built BIM: First estimate of as-damaged locations of reinforced concrete frame members in masonry infill structures. *Adv. Eng. Inform.* **30**, 312–326. https://doi.org/10.1016/j.aei.2016.04.001 (2016).
59. Truong-Hong, L. & Lindenbergh, R. Extracting structural components of concrete buildings from laser scanning point clouds from construction sites. *Adv. Eng. Inform.* **51**, 101490. https://doi.org/10.1016/j.aei.2021.101490 (2022).
60. Cabaleiro, M., Lindenbergh, R., Gard, W. F., Arias, P. & van de Kuilen, J. W. G. Algorithm for automatic detection and analysis of cracks in timber beams from LiDAR data. *Constr. Build. Mater.* **130**, 41–53. https://doi.org/10.1016/j.conbuildmat.2016.11.032 (2017).
61. Díaz-Vilariño, L., Conde, B., Lagüela, S. & Lorenzo, H. Automatic detection and segmentation of columns in as-built buildings from point clouds. *Remote Sens.* **7**, 15651–15667 (2015).
62. Perez-Perez, Y., Golparvar-Fard, M. & El-Rayes, K. Automatic 3D modeling of structural and mechanical components from point clouds. In *Construction Research Congress 2018*, 501–511. https://doi.org/10.1061/9780784481264.049.
63. Riveiro, B., Conde-Carnero, B., González-Jorge, H., Arias, P. & Caamaño, J. C. automatic creation of structural models from point cloud data: The case of masonry structures. *ISPRS Ann Photogramm. Remote Sens. Spatial Inf. Sci.* **II-3/W5**, 3–9. https://doi.org/10.5194/isprsannals-II-3-W5-3-2015 (2015).
64. Rausch, C., Lu, R., Talebi, S. & Haas, C. Deploying 3D scanning based geometric digital twins during fabrication and assembly in offsite manufacturing. *Int. J. Constr. Manag.* https://doi.org/10.1080/15623599.2021.1896942 (2021).
65. Standards Australia. *AS/NZS 3679.1:2016—Structural Steel, Part 1: Hot-Rolled Bars and Sections* (Standards Australia, 2016).
66. *Standards Australia/Standards New Zealand* (2016).
67. Standards Australia. *ASNZS 3679.1:2016—Structural Steel, Part 1: Hot-Rolled Bars and Sections* (Standards Australia/Standards New Zealand, 2016).
68. Pedregosa, F. *et al.* Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
69. Yu, Z., Wang, T., Guo, T., Li, H. & Dong, J. Robust point cloud normal estimation via neighborhood reconstruction. *Adv. Mech. Eng.* **11**, 1687814019836043. https://doi.org/10.1177/1687814019836043 (2019).
70. Standards Australia. *AS/NZS 5131:2016—Structural Steelwork—Fabrication and Erection* (Standards Australia/Standards New Zealand, 2016).
71. Leng, J. *et al.* Digital twin-driven joint optimisation of packing and storage assignment in large-scale automated high-rise warehouse product-service system. *Int. J. Comput. Integr. Manuf.* **34**, 783–800. https://doi.org/10.1080/0951192X.2019.1667032 (2021).
72. Kim, M.-K., Sohn, H. & Chang, C.-C. Automated dimensional quality assessment of precast concrete panels using terrestrial laser scanning. *Autom. Constr.* **45**, 163–177. https://doi.org/10.1016/j.autcon.2014.05.015 (2014).
73. Guo, J., Wang, Q. & Park, J.-H. Geometric quality inspection of prefabricated MEP modules with 3D laser scanning. *Autom. Constr.* **111**, 103053. https://doi.org/10.1016/j.autcon.2019.103053 (2020).
74. Dai, K., Smith, B. H., Chen, S.-E. & Sun, L. Comparative study of bridge management programmes and practices in the USA and China. *Struct. Infrastruct. Eng.* **10**, 577–588. https://doi.org/10.1080/15732479.2012.757332 (2014).
75. Li, L. *et al.* An improved RANSAC for 3D point cloud plane segmentation based on normal distribution transformation cells. *Remote Sens.* **9**, 433. https://doi.org/10.3390/rs9050433 (2017).
76. Smith, A. & Sarlo, R. Automated extraction of structural beam lines and connections from point clouds of steel buildings. *Comput. Aided Civ. Infrastruct. Eng.* **37**, 110–125. https://doi.org/10.1111/mice.12699 (2021).

Acknowledgements

Author contributions

K.M., M.A. and E.A. contributed to the design of the study. K.M., M.A., H.M. and H.L. collected the data. K.M. and M.A. analyzed and interpreted the data. K.M., M.A. and E.A. drafted the manuscript. E.A., H.M. and H.L. reviewed the manuscript. All authors read, commented, and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to H.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.