

## Full length article

## End-to-end point cloud-based segmentation of building members for automating dimensional quality control



Kaveh Mirzaei<sup>a</sup>, Mehrdad Arashpour<sup>a,\*</sup>, Ehsan Asadi<sup>b</sup>, Hossein Masoumi<sup>a</sup>, Amir Mahdiyar<sup>c</sup>, Vicente Gonzalez<sup>d</sup>

<sup>a</sup> Department of Civil Engineering, Monash University, Melbourne, Australia

<sup>b</sup> School of Engineering, RMIT University, Melbourne, Australia

<sup>c</sup> School of Housing, Building, and Planning, Universiti Sains, Malaysia

<sup>d</sup> Department of Civil and Environmental Engineering, University of Alberta, Canada

## ARTICLE INFO

## Keywords:

Point cloud  
Deep learning  
Dimensional quality inspections  
Geometric locations  
Laser scanner sensors  
Machine learning  
Object detection  
Semantic segmentation

## ABSTRACT

A frequent and accurate quality inspection procedure to assess the quality requirements during the life cycle of buildings is crucial. Among different quality measures, the dimensional quality that involves spatial features of buildings is of significant importance. However, the traditional manual inspection of dimensional quality in buildings is unreliable and tedious. Thus, this study presents an end-to-end method for quality inspection of building structural members using point cloud datasets. The proposed method, first, detects and labels structural members within the point cloud based on a set of domain-specific geometric and semantic definitions. Then, each structural member's section width, height, and length are obtained with the proposed bounding box method. Experiments on three real-world buildings' point clouds with various geometric features and noise levels, occlusion, and outliers were also conducted, illustrating the performance efficiency and accuracy of the proposed model for dimensional quality inspection of building structural members.

## 1. Introduction

Satisfying quality requirements during the project life cycle is one of the most critical indicators of a successful construction and infrastructure project [1]. Poor quality is reflected as defects and failures in constructed facilities, leading to increased direct and indirect costs if not dealt with promptly. A recent report on the quality of buildings in Australia's multi-residential sector showed that 97% of buildings in New South Wales, 74% in Victoria, and 71% in Queensland had at least one type of defect [2]. Also, more than 13.6% of bridges were found to be functionally defective in a recent infrastructure report by the American Society of Civil Engineering (ASCE) [3]. These construction projects may be considered safe; however, they need immediate attention to prevent hazardous situations in the future [4–6]. Therefore, a frequent and accurate quality control procedure to assess the as-built conditions of construction projects, especially buildings, is needed. To achieve this goal, a Quality Inspection and Management (QIM) plan is required in buildings during different life cycle phases. A QIM plan oversees various quality criteria, in which "dimensional quality" that involves spatial

features of buildings' elements (i.e., position, dimensions, shape) is of significant importance due to the high probability of system failures that is associated with the lack of proper assessment of such features [7].

Two primary aspects of QIM plans are information acquisition and information communication [8]. Currently, dimensional information for a QIM plan is primarily acquired using visual inspection and manual measurements of targeted building elements, such as the position and dimensions of structural members in a building. While simple, manual acquiring of dimensional information is costly, tedious, and unreliable [9]. For obtaining information from buildings, non-contact sensing technologies, such as 2D photogrammetry and laser scanners, have shown promising performance in the literature [10,11]. 2D cameras have provided a relatively inexpensive solution with high precision for acquiring information from buildings [12,13]; however, they are susceptible to buildings' environmental conditions, such as different varieties of noises and illuminations. Most importantly, they cannot provide accurate depth information, which is crucial for dimensional quality inspection purposes. In such instances, laser scanners have proven to be a good alternative and supplement [14]. Laser scanners capture the

\* Corresponding author at: Monash University, Department of Civil Engineering, Vic 3800, Australia.

E-mail addresses: [kaveh.mirzaei@monash.edu](mailto:kaveh.mirzaei@monash.edu) (K. Mirzaei), [mehrdad.arashpour@monash.edu](mailto:mehrdad.arashpour@monash.edu) (M. Arashpour), [ehsan.asadi@rmit.edu.au](mailto:ehsan.asadi@rmit.edu.au) (E. Asadi), [Hossein.masoumi@monash.edu](mailto:Hossein.masoumi@monash.edu) (H. Masoumi), [amirmahdiyar@usm.my](mailto:amirmahdiyar@usm.my) (A. Mahdiyar), [vagonzal@ualberta.ca](mailto:vagonzal@ualberta.ca) (V. Gonzalez).

external surfaces or an overall 3D geometry of an object in a point cloud format. A point cloud is a collection of points in a 3D coordinate system with geometric or geodetic coordinates [15]. While being a useful asset for acquiring information for QIM plans, communication of raw point clouds with different stakeholders of buildings for quality inspection purposes is challenging due to the lack of semantic information in point clouds [16]. Thus, the theoretical benefits of point clouds for developing QIM plans of buildings [17–19] can only be achieved after the processing of point clouds. This calls for methods capable of processing point clouds accurately and efficiently. In practice, however, time, precision, and computing resources required for manual point cloud processing prevent construction projects from realizing such benefits [20–22]. Thus, various point cloud processing methods have been developed for inspecting different aspects of dimensional quality in buildings, such as dimensions of building façades [23], dimensions of building interior environment [24,25], and deflection/deformation of structures [26,27]. However, the methods proposed by the previous research are unsuitable for an end-to-end dimensional quality inspection of structural members in different building types and life cycle stages [7,28].

Thus, this study aims to overcome the limitations of the traditional manual methods and fill in the research gap by proposing an end-to-end point cloud processing method to use point cloud data of buildings, from which the dimensional quality of building structural members (beams, columns, and bracings) is automatically inspected. The proposed method takes the raw point cloud of building structural members as inputs and outputs each structural member's cross-section width and height along with member length for dimensional quality inspection. The contributions of this paper are threefold: (1) a comprehensive end-to-end method that utilizes both geometric and shape definitions of structural members for more robust object detection in point clouds with various levels of obstacles, outliers, missing information, and noises, (2) a method for automatically generating and adopting synthetic training dataset for semantic segmentation using PointNet is presented, and (3) the approach is supported by contextual knowledge of civil and structural engineering to be generalizable to other buildings and construction projects in different life cycle stages with various geometric features. The extracted information can also be directly transformed into the digital twin of construction models. The rest of this paper is organized as follows. Section 2 introduces the research background on point cloud segmentation methods and discusses different aspects of dimensional quality inspection with point clouds. The details of the proposed end-to-end method are described in Section 3. Section 4 introduces the validation experiments performed to assess the proposed method's performance. The discussion on the results obtained from applying the proposed method is presented in Section 5, and Section 6 concludes this article.

## 2. Background

Over the past decade, much work has been conducted to inspect the quality of construction projects with non-contact sensing devices. Noteworthy examples are radio-frequency identification (RFID) [8], 2D cameras [29–31], 3D cameras [32], and laser scanners [33–35]. Considering the capabilities and accuracy of 3D scanned data [36], this work focuses on inspecting the dimensional quality of structural members with 3D point cloud data obtained from laser scanners. To achieve this target, first, structural members should be detected and labeled within the point cloud based on a set of domain-specific geometric and semantic definitions. Then, the dimensions will be obtained based on the specifications of each structural member. An overview and a report on previous literature on the abovementioned steps are presented in this section.

### 2.1. Geometric segmentation

To date, numerous point cloud processing methods have been developed to utilize the unique spatial and geometric definitions and relationships between structural members. Using geometric definitions and relationships, it would be possible to detect possible structural members in point clouds of buildings [37]. This process is usually initialized by a segmentation algorithm to cluster similar points together. In the next step, clusters of similar points would be labeled by adopting contextual hard-coded knowledge. For instance, considering columns, the potential points belonging to columns must have a normal vector perpendicular to the Z-axis. The most common point cloud segmentation algorithms used in the previous literature are shape-fitting and clustering [38].

Shape-fitting algorithms have been developed to group points into clusters that fit geometric models such as planes [39]. One of the most used shape-fitting algorithms is the RANdom SAmple Consensus (RANSAC). The RANSAC algorithm is an optimization method that generates candidate solutions for a predefined mathematical model [40]. This method is iterative, which begins with the recognition of random sampling of a minimal number of points to estimate the parameters of shapes, such as planes or cylinders [41]. The next step consists of labeling the points so that a set of points at a certain distance from the model is appointed inliers while the rest are outliers. After these iterative rankings and labeling, the best result or consensus will be chosen [42]. Given its computationally expensive nature, applications of the RANSAC method are limited to fitting simple shapes such as planes and spheres. Therefore, RANSAC is mostly suitable for applications such as façade modeling [43] and building roof segmentation [44], in which point clouds consist of simple shapes and a low level of noise.

Another group of algorithms for the geometric segmentation of point clouds are clustering methods. Clustering methods are based on identifying patterns within different point cloud features to automatically generate subgroups of similar types, also known as clusters [45]. Some of the most common point cloud features used for clustering are spatial position and Euclidian distance between points [46], the normal vector of points [47], and point density of point clouds [48]. Also, based on the application in hand, handcrafted features can be utilized for point cloud clustering. Next, a heuristic approach combining domain-specific rules is required to break down the point cloud into regions of interest and assign a label to each cluster [49,50]. Clustering methods were utilized for building indoor reconstruction by segmenting structured point clouds using the distance between room faces as a feature [51]. Also, a multiscale feature detection method accounting for surface roughness and curvature was developed for the segmentation of building point cloud models to detect architectural/structural features and Mechanical, Electrical, and Plumbing (MEP) systems [52]. While applicable for small and noiseless datasets, the performance of clustering techniques with multiscale features is susceptible to the presence of noise and outliers in the point cloud. Also, discontinuities in the point cloud, such as sharp features, heavily impact the performance of multiscale feature clustering techniques [53]. As a result, relying solely on geometric definitions and contextual relationships does not result in robust information for detecting structural members.

### 2.2. Semantic segmentation

In addition to geometric specifications, structural members, such as columns, are also defined by section shapes. Therefore, assigning a section shape label to point clusters, also known as semantic segmentation, along with considering their contextual relationships would lead to a robust model for detecting structural members in real-world point clouds. To reach such a purpose, machine learning methods, such as Convolutional Neural Networks (CNN), have performed well in a variety of applications [54–56], such as autonomous vehicles [57] and text classification [58]. However, point clouds are irregular, unstructured,

and unordered data that prevent the convolution operations of CNN methods from being established [59]. Moreover, a high rate of noise and occlusion in real-world point clouds affects the performance of CNNs for semantic segmentation of point clouds. Thus, many methods have been developed to apply 3D deep learning to point clouds for semantic segmentation purposes, which are divided into two main groups: 1. Converting point clouds into a structured grid [60,61], and 2. Applying deep learning to unstructured point clouds [62].

The main semantic segmentation methods that convert point clouds into a structured grid are multiview-based methods and voxel-based methods [63]. Voxel-based methods transform the point clouds into predefined fixed-size binary voxels [61]. Then, a deep learning network is applied to the point cloud for assigning class labels to each point in the dataset [64]. Voxel-based methods were adopted for point cloud semantic segmentation in applications such as indoor scene interpretation [65,66] and building detection from an urban scene [67,68]. However, voxel-based methods have a high memory requirement, limiting the number of voxelization cubes. Thus, voxel-based methods lack the desirable performance for semantic segmentation of building structural members.

On the other hand, multiview-based approaches transform point clouds into a set of 2D images to generate structured datasets [69]. The main advantages of multiview-based approaches over voxel-based methods are; 1. Avoiding quantization of voxelization, and 2. Adopting well-researched and optimized 2D CNN methods. However, transforming point clouds into structured grids, in general, is memory-intensive and leads to the loss of much information [59], which makes adopting point clouds for detecting structural members of buildings a challenging task.

To overcome issues related to converting point clouds into structured grids, many techniques have been developed, starting with PointNet [70], to apply deep learning to an unstructured point cloud. PointNet is a local feature learner that encodes spatial features of each point in a point cloud dataset to generate a global point signature by aggregating individual point features. While achieving state-of-the-art performance on various benchmark point cloud datasets, PointNet's performance declines in real-world datasets due to missing the local dependency among points [71]. To compensate for this issue, many deep learning approaches were developed to not only deal with the inherent unstructuredness of point clouds but also hierarchically capture the local structure of point clouds. The aforementioned approaches are categorized into three main classes of non-exploring local correlation approaches, such as PointNet++ [62], approaches that explore local correlation, such as PointCNN [72], and graph-based approaches, such as Dynamic Graph CNN (DGCNN) [73]. These methods have addressed the challenges of real-world datasets for the semantic segmentation of point clouds [74]. A comparison among deep learning-based methods for semantic segmentation of point clouds of building interiors showed that PointNet had outperformed other local feature learners in predicting long objects such as columns [75]. Thus, considering local dependency among points being addressed by geometric segmentation of point cloud, PointNet is a suitable local feature learner for semantic segmentation of point clouds to identify the section shape of structural members.

Recent studies have detected structural members [76] and MEP systems [74] using machine learning-based methods. While achieving high detection accuracy, these research efforts required large numbers of pre-annotated real-world training datasets similar to their investigated case studies, which opposes a challenge for using such methods for other construction projects. To overcome such challenges, further research efforts utilized slicing methods and image processing approaches for detecting section shapes within point cloud datasets [77,78]. 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.

### 2.3. Dimensional quality inspection

Dimensional quality inspection is the process of collecting spatial information such as size, position, and shape from targeted objects and comparing it with as-design models [79]. The current practice for dimensional quality inspection of building structural members is primarily based on manual visual inspection and measurements with simple tools such as tape [7,80]. While simple, the manual inspection of structural framing members is error-prone, tedious, and potentially hazardous [79]. As a result, laser scanners have recently gained a plethora of attention to be adopted for quality inspection of structural members by generating project as-built models in the form of point clouds [81]. Point clouds obtained from laser scanners were used in different aspects of dimensional quality inspection, such as automatically detecting corners and edges of precast concrete members in an off-site manufacturing factory [34,82]. Also, the positioning of rebars in reinforced precast concrete members with point clouds was investigated [83]. Moreover, point clouds were utilized to detect pipe spool members' position and orientation to identify local discrepancies and deviations from as-built models [84]. Additionally, a dimensional compliance check approach was developed by detecting 3D CAD objects from point clouds [85]. In this research, two dimensional quality inspection aspects, i.e., section dimensions and length of building structural members, are investigated.

Considering the body of research studied above, the current scope of study into dimensional quality inspection with point clouds is mainly limited to specific small-sized objects, such as precast concrete elements in a controlled environment, and only utilizes either geometric or semantic segmentation methods for detecting structural members. Also, the proposed methods are restricted to buildings with specific geometric and shape features. As a result, a general end-to-end dimensional quality method for structural members that inputs raw point clouds and automatically outputs the dimensions of structural members is sparse in the literature [7]. Hence, this study aims to develop an end-to-end dimensional quality inspection approach for building structural members in real-world situations that: (1) detects structural building members (columns, beams, and bracings) by applying a set of domain-specific geometric and semantic definitions, and (2) obtains the dimensions of structural framing members for a compliance check with ground truth dimensions.

## 3. Methods

The end-to-end dimensional quality inspection methodology developed in this study aims to automatically extract structural members of buildings at different life cycle stages using point clouds and quantify their dimensional quality. The main components of a building's structural system are beams and columns designed for taking vertical loads and bracings for resisting lateral loads. While each building, as a construction project, is unique in terms of architectural and structural designs, building structural members have common attributes and characteristics. Such common attributes form the contextual knowledge basis for detecting building structural members as below.

**Attribute 1:** beams are often distributed horizontally in rectangular grids that carry loads perpendicular to their longitudinal direction and transfer them to columns.

**Attribute 2:** columns are vertical members designed to take loads from beams and transfer them to the building foundation. Thus, columns are often distributed beneath beams. Also, the width of columns is usually designed to be equal to or more significant than the width of beams connected for adhering to the "strong column-weak beam" concept.

- Attribute 3:** bracings are diagonal members distributed beneath beams with an inclined angle between 30 and 60 degrees.  
**Attribute 4:** the cross-section shape of building structural members are often channel shapes, universal beams, rectangular shapes, or circular shapes.

The common attributes of building structural members are divided into geometric information consisting of geometric definitions and spatial relationships, semantic information consisting of cross-section shape information, and dimension acquisition consisting of structural design contextual knowledge for each member. Therefore, the end-to-end dimensional quality inspection of building structural members using point clouds is summarized into four major steps: 1. Preprocessing, 2. Geometric segmentation, 3. Semantic segmentation, and 4. Obtaining of structural members dimensions. The proposed method workflow is visually illustrated in Fig. 1.

### 3.1. Point cloud preprocessing

The first preprocessing step is floors, ceiling slabs, and wall filtration from point clouds. A substantial number of acquired points by a scanner belong to the floor, ceiling slab, and wall, with no beneficial information for this study. Those redundant points cause a computational burden

during further point cloud processing steps. An improved RANSAC algorithm based on Normal Distribution Transformation (NDT) cells [86] is implemented to detect planes within the point cloud. This method represents the point cloud with a set of NDT cells. The geometric features of NDT cells are then utilized to classify the NDT cells into planar and non-planar cells. Next, planes will be classified into horizontal and non-horizontal planes that can be utilized for removing walls and floors. Also, another method proposed in the literature for detecting floor, ceiling, and wall points consists of three main steps of decomposing point clouds into 2D cells in the xy planes using a quadtree representation, local surface extraction of floor and ceiling slab, and cell-based region growing segmentation (CRG) for filtering out final points of floor, ceiling slab, and walls [28].

The next preprocessing step is aligning building structural members with the point cloud coordinate system. The columns are correctly adjusted to the Z-axis during the scanning procedure with Terrestrial Laser Scanners (TLS) and require no further adjustments. However, the beam system in point clouds has an arbitrary orientation in geo-space and should be rotated to align with the X and Y axes. This transformation is achieved by multiplying the input point cloud ( $P = \{p_i = (x_i, y_i, z_i)\} \in R^3\}$ ) with a rotation matrix, as shown in Eq.1, about the Z-axis,

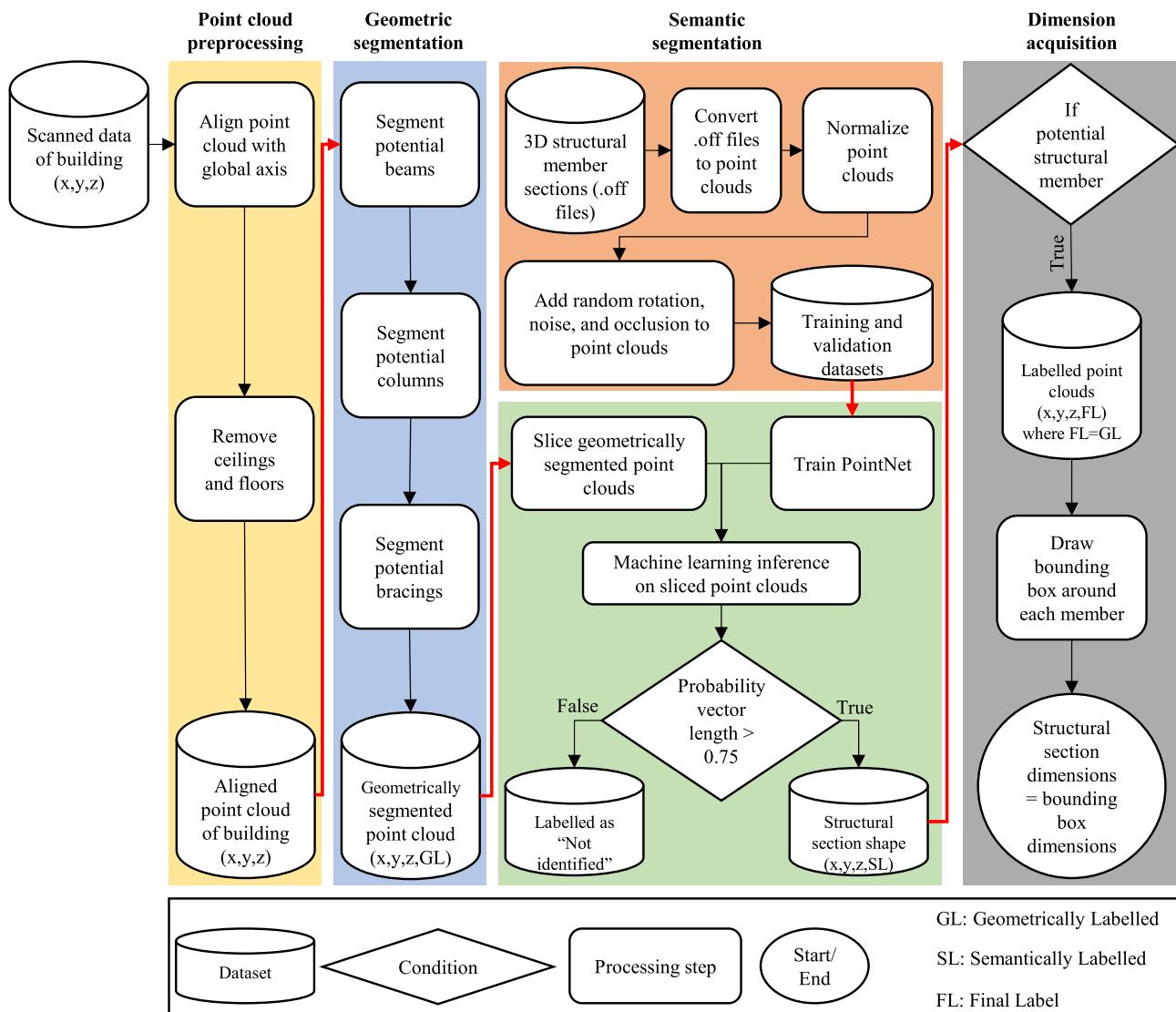


Fig. 1. The workflow for an end-to-end dimensional quality inspection of structural members with point clouds.

$$\mathbf{P}_{rotated,i} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \quad (1)$$

where  $\theta$  is the rotation angle. Since the segmentation proposed in this study is generally robust against slightly skewed elements, this angle is calculated automatically by adopting Principal Component Analysis (PCA) method. Geometrically speaking, PCA projects the data in the direction that explain a maximal amount of variance. This principal component direction, which has an angle of  $\theta$  to the global axis, is aligned with the largest dimension of buildings in xy plane. Thus, using Eq. 1, the building will be aligned with the global x and y axes. The preprocessing steps are shown in Fig. 2, in which xyz is the global coordinate system, and x' y' z' is the local coordinate system along building structural members.

### 3.2. Geometric segmentation

The first step for detecting building structural members in a point cloud is to identify points possessing common geometric attributes of structural members. To achieve this, the contextual knowledge of common geometric attributes, i.e., geometric definitions and spatial relationships, of structural members is deployed for each type of building structural members in the following sections.

#### 3.2.1. Beams

In buildings, beams are often horizontal members that carry vertical loads perpendicular to their longitudinal direction. Therefore, beams are geometrically defined as vertical members distributed in a slice parallel to the xy plane. To utilize the geometric definition of beams, first, the point cloud ( $P = \{p_i = (x_i, y_i, z_i)\} \in R^3$ ) of a building is divided into horizontal slices. To find the number of horizontal slices  $N_{hor\_slice}$ , the maximum Z-axis coordinate  $Z_{max}$  and the minimum Z-axis coordinate  $Z_{min}$  is retrieved from the point cloud. Then  $N_{hor\_slice}$  is determined based on Eq. (2).

$$N_{hor\_slice} = \frac{Z_{max} - Z_{min}}{t_{hor\_slice}} \quad (2)$$

where  $t_{hor\_slice}$  is the thickness of horizontal slices. In the next step, each point  $p_i$  in the point cloud is assigned to a horizontal slice  $S_{hor\_slice,j}$  if  $z_i \in p_i(x_b, y_b, z_b)$  satisfies the condition in Eq. (3).

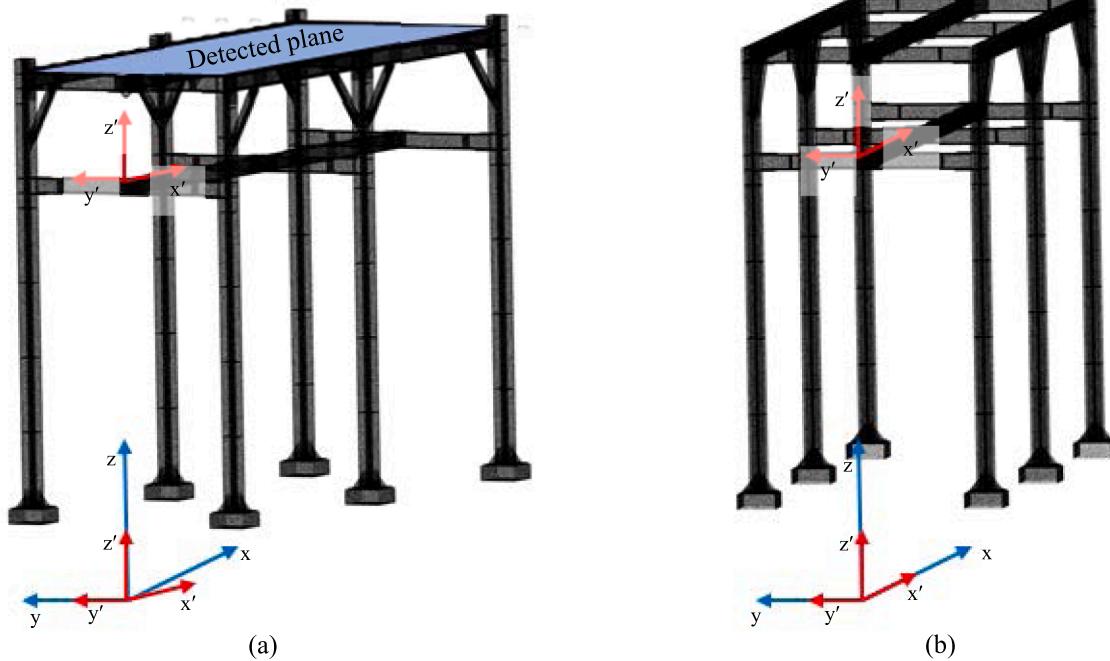
$$p_i \in S_{hor\_slice,j} \text{ if } \{Z_{min} + (t_{hor\_slice} * (j - 1)) \leq z_i \leq Z_{min} + (t_{hor\_slice} * j)\} \quad (3)$$

where j goes from 1 to  $N_{hor\_slice}$ . After distributing points in horizontal slices, it is hypothesized that the number of points in slices containing potential beam points  $p_{pot\_beam,i} (x_b, y_b, z_b, GLbeam)$  is higher than in other slices. To implement this hypothesis, the number of points in each horizontal slice  $N_{p\_hor\_slice}$  is retrieved and stored in a list as  $L_{hor\_slice} = [N_{p\_hor\_slice,1}, N_{p\_hor\_slice,2}, \dots, N_{p\_hor\_slice,N_{hor\_slice}}]$ . Then, the standard deviation  $\sigma_{hor\_slice}$  and mean values  $\mu_{L_{hor\_slice}}$  of the members in  $L_{hor\_slice}$  are calculated. Then, points in a slice are labeled as potential beam points  $p_{pot\_beam,i} (x_b, y_b, z_b, GLbeam)$  if  $N_{p\_hor\_slice}$  of that horizontal slice satisfies the condition in Eq. (4).

$$p_i \in S_{hor\_slice,j} \rightarrow p_{pot\_beam,i} \text{ if } \left\{ \frac{N_{p\_hor\_slice,i} - \mu_{L_{hor\_slice}}}{\sigma_{L_{hor\_slice}}} > threshold \right\} \quad (4)$$

where j goes from 1 to  $N_{hor\_slice}$ , and the threshold value is set based on the statistical distribution of data and level of outliers in the point cloud dataset. Next, each horizontal slice containing potential beam points is extracted from the building point cloud. Based on the thickness of horizontal slices, there might be more than one horizontal slice containing potential beam points of each building floor. Therefore, to merge the potential beam point clouds of each floor, a Density-Based Spatial Clustering of Applications with Noise (DBSCAN)[87] is adopted to cluster potential beam points based on  $z_i \in p_{pot\_beam,i} (x_b, y_b, z_b, GLbeam)$ . DBSCAN operates on the assumption that clusters are dense regions of data separated by regions of lower density. DBSCAN is suitable for clustering potential beam points based on  $z_i \in p_{pot\_beam,i} (x_b, y_b, z_b, GLbeam)$  as it is robust to outliers and does not require the number of clusters to be specified by the user. Finally, the clustered potential beam points of each floor  $j$  are saved as a new point cloud ( $P_{pot\_beam\_floor,j} = \{p_{pot\_beam,i} = (x_b, y_b, z_b, GLbeam)\} \in R^4$ ).

In the structural design of buildings, designers often place beams in a rectangular grid along the x and y axes due to the practical and structural problems associated with diagonal (zigzag) beams. This spatial



**Fig. 2.** Point cloud preprocessing steps: (a) raw point cloud (b) after preprocessing steps.

relationship between different beam members in a building is adopted to further cluster potential beam points of each floor ( $P_{pot\_beam\_floor,j} = \{p_{pot\_beam,i} = (x_b, y_b, z_b, GL_{beam})\} \in R^4$ ) obtained from the previous step into individual instances of potential beam points ( $P_{pot\_beam} = \{p_{pot\_beam,i} = (x_b, y_b, z_b, GL_{beam})\} \in R^4$ ). To implement this spatial relationship, the beams in the x direction must be separated from those in the y direction. Thus, a vertical slicing method is adopted to separate beams in the x direction from the y direction in potential beam points of each floor. This process is only needed to be performed in either the x or y direction. The method proposed here considers the vertical slicing for beams in the x direction. The number of vertical slices  $N_{ver\_slice}$  is determined based on Eq. (5).

$$N_{ver\_slice} = \frac{Y_{max} - Y_{min}}{w_{ver\_slice}} \quad (5)$$

where  $Y_{max}$  is the maximum Y-axis coordinate,  $Y_{min}$  is the minimum Y-axis coordinate in each potential beam point cloud of each floor, and  $w_{ver\_slice}$  is the width of vertical slices. In the next step, each potential beam point  $p_{pot\_beam,i}$  is assigned to a vertical slice  $S_{ver\_slice,j}$  if  $y_i \in P_{pot\_beam,i}$  ( $x_b, y_b, z_b$ ) satisfies the condition in Eq. (6).

$$P_{pot\_beam,i} \in S_{ver\_slice,j} \text{ if } : \{Y_{min} + (w_{ver\_slice} * (j - 1)) \leq y_i \leq Y_{min} + (w_{ver\_slice} * j)\} \quad (6)$$

where j goes from 1 to  $N_{ver\_slice}$ . After distributing potential beam points of each floor in vertical slices, it is hypothesized that the number of points in vertical slices containing potential beam points in the x direction is higher than in other slices. This hypothesis is applied by extracting the number of points  $N_{p\_ver\_slice}$  in each vertical slice  $S_{ver\_slice,j}$  and generating a list as  $L_{ver\_slice} = [N_{p\_ver\_slice,1}, N_{p\_ver\_slice,2}, \dots, N_{p\_ver\_slice,N_{ver\_slice}}]$ . Then, the standard deviation  $\sigma_{L_{ver\_slice}}$  and mean values  $\mu_{L_{ver\_slice}}$  of the members in  $L_{ver\_slice}$  are calculated. Potential beam points in a vertical slice are considered to be in the x direction  $p_{pot\_beam\_x,i}$  if  $N_{p\_ver\_slice}$  of that vertical slice satisfies the condition in Eq. (7).

$$P_{pot\_beam,i} \in S_{ver\_slice,j} \rightarrow p_{pot\_beam\_x,i} \text{ if } : \left\{ \frac{N_{p\_ver\_slice,j} - \mu_{L_{ver\_slice}}}{\sigma_{L_{ver\_slice}}} > threshold \right. \quad (7)$$

where j goes from 1 to  $N_{ver\_slice}$ , and the *threshold* value is set based on the statistical distribution of data and contextual knowledge of building structural members, such as the average distance between neighboring beams. Next, potential beam members in the x direction are removed from potential beam points of each floor and stored as new point clouds, and the remaining points are labeled as potential beam members in the y direction. At this stage, potential beam members of each floor are divided into potential beam members in the x direction  $p_{pot\_beam\_x,i}$  ( $x_b, y_b, z_b, GL_{beam}$ ) and potential beam members in the y direction  $p_{pot\_beam\_y,i}$  ( $x_b, y_b, z_b, GL_{beam}$ ). Finally, a DBSCAN clustering is implemented to cluster separate instances of potential beam members in both x and y directions and save each of them as a new point cloud ( $P_{pot\_beam} = \{p_{pot\_beam,i} = (x_b, y_b, z_b, GL_{beam})\} \in R^4$ ).

### 3.2.2. Columns

In buildings, columns are often geometrically defined as vertical members. This geometric definition is utilized by calculating the normal vector  $\vec{n}_i = (n_x, n_y, n_z)$  of each point in the point cloud ( $P = \{p_i = (x_i, y_i, z_i)\} \in R^3$ ). To achieve this purpose, a least square plane is fitted to each point in the point cloud, and its k nearest neighbors and the normal vector perpendicular to the plane are obtained and assigned to the seed point. Then, the obtained normal vector for each point is concatenated with its coordinate vector as  $p_{normal,i} = (x_b, y_b, z_b, n_x, n_y, n_z) \in R^6$ . To find the verticality of each point in the point cloud, the angle  $\theta$  of the normal vector ( $\vec{n}_i$ ) with Z-axis ( $\vec{v}$ ) is calculated. Then, each point  $p_i$  is considered vertical if the condition in Eq. (8) is satisfied.

$$p_i \rightarrow p_{vertical,i} \text{ if } : \left\{ \theta = 90^\circ - \cos^{-1} \left( \frac{\vec{n}_i \cdot \vec{v}}{|\vec{n}_i| \cdot |\vec{v}|} \right) \right\} \left\langle threshold \right. \quad (8)$$

where the *threshold* value is set based on the contextual knowledge of building structural members.

In the building structural engineering concept, columns are designed to take loads from beams and transfer them to the building foundation. Thus, columns are often distributed beneath beams. To utilize this spatial relationship, each vertical point  $p_{vertical,i} = (x_b, y_b, z_b, n_x, n_y, n_z) \in R^6$  is labeled as a potential column point if it exists beneath a potential beam member as shown in Eq. (9).

$$p_{vertical,i} \rightarrow p_{pot\_column,i} \text{ if } : \left\{ \begin{array}{l} x_{Min,j} < x_i < x_{Max,j} \\ y_{Min,j} < y_i < y_{Max,j} \\ z_i < z_{Min,j} \end{array} \right. \quad (9)$$

where  $x_{Min,j}$  and  $x_{Max,j}$  are the minimum and maximum X-axis coordinate of  $j^{th}$  potential beam point,  $y_{Min,j}$  and  $y_{Max,j}$  are the minimum and maximum Y-axis coordinate of  $j^{th}$  potential beam point,  $z_{Min,j}$  is the minimum Z-axis coordinate of  $j^{th}$  potential beam point, and j goes from 1 to the number of potential beam members identified in section 3.2.1. Finally, a DBSCAN clustering is implemented to cluster separate instances of potential column members and save each of them as a new point cloud ( $P_{pot\_column} = \{p_{pot\_column,i} = (x_b, y_b, z_b, GL_{column})\} \in R^4$ ) and remove them from the building point cloud.

### 3.2.3. Bracings

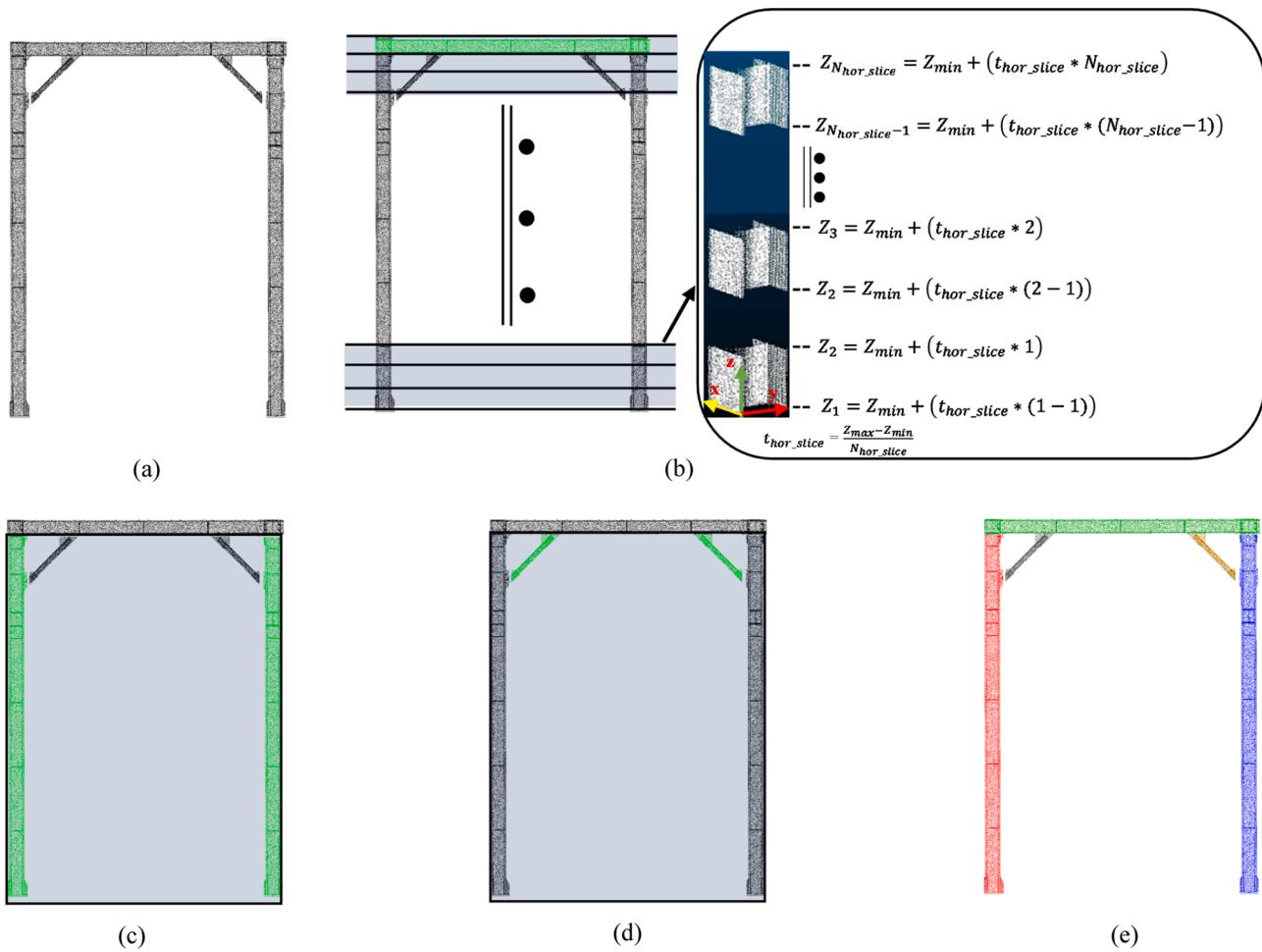
Bracings are often geometrically defined in buildings as diagonal members with an inclined angle of between 30 and 60 degrees from xy plane. To implement this geometric definition, the normal vector  $\vec{n}_i = (n_x, n_y, n_z)$  of each point in the point cloud ( $P = \{p_i = (x_i, y_i, z_i)\} \in R^3$ ) is calculated using the method described in section 3.2.2. Then, the angle  $\theta$  of each point in the point cloud is obtained by calculating the angle  $\theta$  of the normal vector ( $\vec{n}_i$ ) with Z-axis ( $\vec{v}$ ). Next, the condition stated in Eq. (10) is checked for each point  $p_i$  to obtain the points satisfying the geometric definition of bracings.

$$p_i \rightarrow p_{geometric\_bracing,i} \text{ if } : \{30^\circ < \theta = \cos^{-1} \left( \frac{\vec{n}_i \cdot \vec{v}}{|\vec{n}_i| \cdot |\vec{v}|} \right) < 60^\circ\} \quad (10)$$

In the building structural engineering concept, bracings are distributed beneath beams to provide lateral load resistance for buildings. To utilize this spatial relationship, each  $p_{geometric\_bracing,i} = (x_b, y_b, z_b) \in R^3$  is labeled as a potential bracing point if it exists beneath a potential beam member, as shown in Eq. (11).

$$p_{geometric\_bracing,i} \rightarrow p_{pot\_bracing,i} \text{ if } : \left\{ \begin{array}{l} x_{Min,j} < x_i < x_{Max,j} \\ y_{Min,j} < y_i < y_{Max,j} \\ z_i < z_{Min,j} \end{array} \right. \quad (11)$$

where  $x_{Min,j}$  and  $x_{Max,j}$  are the minimum and maximum X-axis coordinate of  $j^{th}$  potential beam point,  $y_{Min,j}$  and  $y_{Max,j}$  are the minimum and maximum Y-axis coordinate of  $j^{th}$  potential beam point,  $z_{Min,j}$  is the minimum Z-axis coordinate of  $j^{th}$  potential beam point, and j goes from 1 to the number of potential beam members identified in Section 3.2.1. Finally, a DBSCAN clustering is implemented to cluster separate instances of potential bracing members and save each of them as a new point cloud ( $P_{pot\_bracing} = \{p_{pot\_bracing,i} = (x_b, y_b, z_b, GL_{bracing})\} \in R^4$ ) and remove them from the building point cloud. The geometric segmentation process for each type of building structural member is shown in Fig. 3.



**Fig. 3.** Geometric segmentation of possible structural members: (a) original point cloud, (b) slicing for identifying potential beam points, (c) detecting potential column points, (d) segmenting potential bracing points, and (e) geometrically segmented point cloud.

### 3.3. Semantic segmentation

The geometric segmentation performed in the previous section identifies and labels the points based on the geometric definitions of structural members. Due to buildings' dynamic and complicated environment, many non-structural members with geometric attributes similar to structural members can be mistakenly detected as structural members. A common attribute regarding building structural members' cross-section shape is utilized to avoid this mistake. The cross-section shape of building structural members are often channel shapes, universal beams, rectangular shapes, or circular shapes. Therefore, a semantic segmentation step is developed to classify the section shape of objects found in the geometric segmentation step.

Semantic segmentation is the semantic classification of point clouds into clusters of points with similar cross-section shape label properties. For this purpose, a local feature learner, PointNet, for the classification step of the semantic segmentation network is utilized. The network applies a symmetric function to point clouds to make them irrespective of the input order of points to deploy deep learning algorithms without converting point clouds into a structured grid. Also, multilayer perceptrons are used to transform the feature dimensions of the points in the point cloud to a higher dimensional space. Moreover, the max-pooling function is deployed as a symmetric function to generate a global high-dimensional feature vector from the dataset used for part segmentation and classification purposes. PointNet, as a deep learning method, must be trained with a training dataset before being applied for the section-shape classification of geometrically labeled structural

members detected in Section 3.2. The following sections discuss the proposed method for generating a training dataset from synthetic data and applying PointNet for section-shape classification.

#### 3.3.1. Training dataset generation

One of the challenges of applying semantic segmentation networks to the point clouds of buildings for detecting structural members is the lack of publicly available labeled training datasets. Also, generating a training dataset from real-world structural member point clouds increases the possibility of having inadequate, nonrepresentative, and poor-quality training datasets. Therefore, this research developed a framework to adopt synthetic training datasets. This framework consists of six steps as below:

Step 1: obtaining 3D mesh models of structural sections (.off file formats) from the available libraries online (e.g., Sketchup Warehouse).

Step 2: converting the obtained .off 3D mesh models to point clouds (.pts). Meshes are represented by vertices, points in 3D space, and triangular faces created by three vertex indices. Then, a predefined number of points are sampled on triangular faces. To achieve a uniform point sampling, the number of points sampled on each triangular face is proportionate to its area.

Step 3: normalizing the coordinates of obtained point clouds.

Step 4: adding random rotation based on three different axes so the network can detect sections with various orientations. For each point  $p_i$  ( $x_i, y_i, z_i$ ) the rotation matrices are calculated as shown in Eq. (12).

$$\begin{aligned} \begin{bmatrix} x_z \\ y_z \\ z_z \end{bmatrix} &= \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \rightarrow \begin{bmatrix} x_x \\ y_x \\ z_x \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \cdot \begin{bmatrix} x_z \\ y_z \\ z_z \end{bmatrix} \rightarrow \begin{bmatrix} x_f \\ y_f \\ z_f \end{bmatrix} \\ &= \begin{bmatrix} \cos(\gamma) & 0 & \sin(\gamma) \\ 0 & 1 & 0 \\ -\sin(\gamma) & 0 & \cos(\gamma) \end{bmatrix} \cdot \begin{bmatrix} x_f \\ y_f \\ z_f \end{bmatrix} \end{aligned} \quad (12)$$

where  $(x_z, y_z, z_z)$  is the coordinates of point  $p_i (x_i, y_i, z_i)$  after rotation around Z-axis using a transformation matrix with a random angle of  $\theta$ ,  $(x_x, y_x, z_x)$  is the coordinates of point  $p_z (x_z, y_z, z_z)$  after rotation around X-axis using a transformation matrix with a random angle of  $\phi$ , and  $(x_f, y_f, z_f)$  is the coordinates of point  $p_x (x_x, y_x, z_x)$  after rotation around Y-axis using a transformation matrix with a random angle of  $\gamma$ , which is reported as the final coordinate of each point.

Step 5: adding random noise to point clouds to provide a similar noise situation to real-world datasets using a Normal (Gaussian) distribution.

Step 6: adding occlusions to random point clouds in random directions to familiarize the network with possible occlusions in building point clouds. The occlusion in point clouds is defined as missing a portion of points in a specific location. To utilize this definition, points in each point cloud are sorted based on a random coordinate direction. Then, a random number of points are removed from the point cloud.

**Fig. 4** illustrates the framework developed in this research for generating a synthetic training dataset.

### 3.3.2. PointNet implementation

After training the network with synthetic datasets, it can be applied to the potential structural members detected in the geometric segmentation step. The classification network is trained to classify the cross-section shape perpendicular to the length of potential structural members. However, building structural members are often described as lengthy objects, in which the length of members is more significant than the dimensions of the cross-section shape. Therefore, potential building structural members must be sliced along their length to minimize the

impact of member length in classifying the cross-section shape, as shown in **Fig. 5a**. While it is possible to manually design synthetic training samples with longer structural members, the time and manual labor required for this task contradict the aims of this paper as a fully automated method. Therefore, a slicing process breaks down the potential structural members point clouds ( $P_{pot\_beam}$  OR  $P_{pot\_column}$  OR  $P_{pot\_bracing} = \{p_i = (x_i, y_i, z_i, GL)\} \in R^4$ ) along their longitudinal direction into smaller point clouds ( $P_{pot\_sliced,j} = \{p_i = (x_i, y_i, z_i, GL)\} \in R^4$ ) so that the section shapes become the members' predominant feature. The slicing method initializes by picking the closest point to the corner of the point cloud model. Then, a K-Nearest Neighbor algorithm is adopted to group the k nearest points to the seed point based on Euclidian distance. Next, a bounding box is created around the selected points. Finally, any point in the potential structural members' point clouds will be extracted and saved as a new point cloud ( $P_{pot\_sliced,j} = \{p_i = (x_i, y_i, z_i)\} \in R^3$ ) if the conditions of Eq. (13) are met.

$$p_i \in (P_{pot\_beam}) \text{ OR } (P_{pot\_column}) \text{ OR } (P_{pot\_bracing}) \rightarrow p_i \in P_{sliced,j} \text{ if}$$

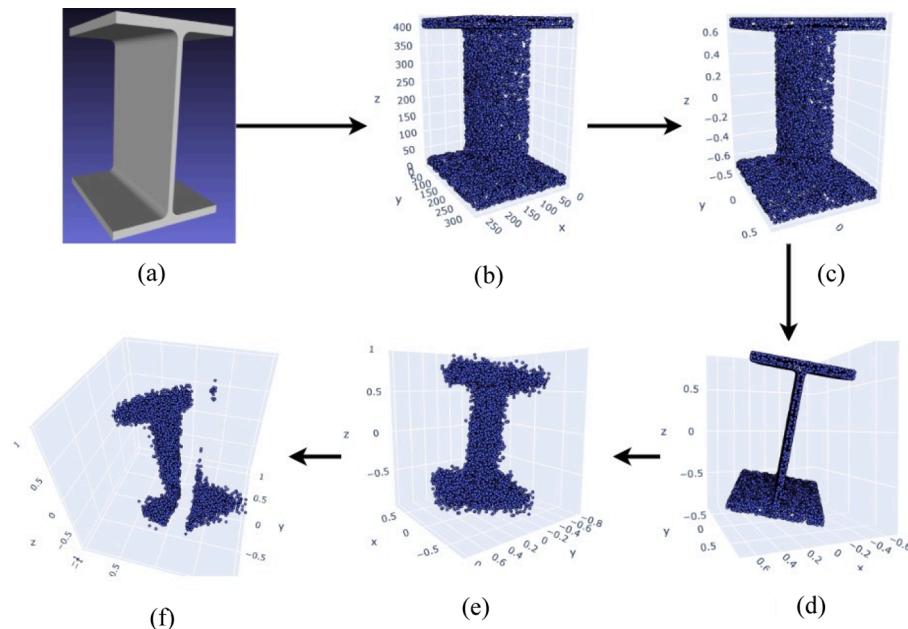
$$\begin{cases} x_{Min,j} < x_i < x_{Max,j} \\ y_{Min,j} < y_i < y_{Max,j} \\ z_{Min,j} < z_i < z_{Max,j} \end{cases} \quad (13)$$

where  $x_{Min,j}$  and  $x_{Max,j}$  are the minimum and maximum X-axis coordinate of  $j^{th}$  bounding box,  $y_{Min,j}$  and  $y_{Max,j}$  are the minimum and maximum Y-axis coordinate of  $j^{th}$  bounding box,  $z_{Min,j}$  and  $z_{Max,j}$  are the minimum and maximum Z-axis coordinate of  $j^{th}$  bounding box, and  $j$  goes from 1 to the number of bounding boxes created in this step.

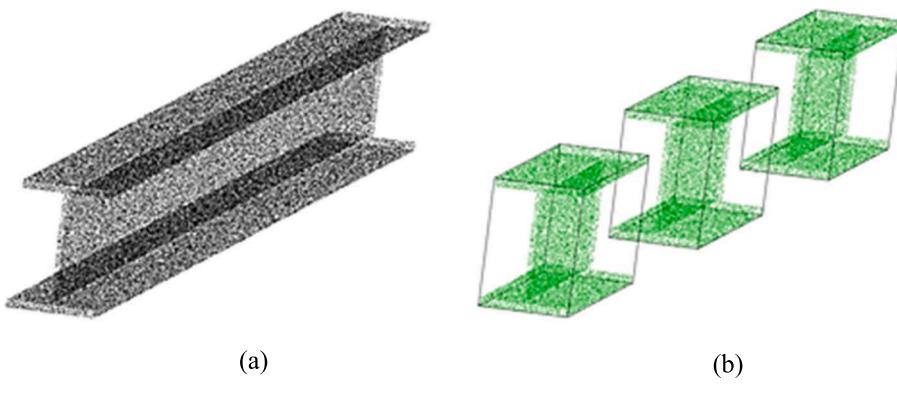
Then, the network will be applied to the extracted point clouds for section shape inference, as shown in **Fig. 5b**. The output of the classification procedure on each point cloud slice is a probability distribution vector containing the probability values for different classification class categories. The confidence score of the classification network is obtained by calculating the Probability Vector Length (PVL) of each classified point cloud slice using Eq. (14).

$$PVL = \sqrt{n(\sigma_{PVL})^2 + \frac{1}{n}} \quad (14)$$

where  $n$  is the size of the probability vector population and  $\sigma_{PVL}$  is the



**Fig. 4.** Framework for generating synthetic training dataset: (a) 3D model of structural section shapes, (b) structural section shapes point cloud converted from 3D models, (c) normalized structural section shapes point clouds, (d) random rotation of the structural section shapes, (e) structural section shapes with random noise, and (d) structural section shapes with random occlusion.



**Fig. 5.** (a) Geometrically segmented point cloud without semantic section shape labels, and (b) sliced possible structural members with semantic section shape labels.

standard deviation of the probability vector population. The maximum value of PVL always equals one, meaning that the probability of one of the population members is one, and the rest is zero. In this case, the classification network has the highest prediction confidence. The minimum value of PVL equals to  $\frac{1}{\sqrt{n}}$  meaning that all probability population members have the same probability, which is the lowest classification network confidence. Therefore, the threshold for the acceptable confidence of the classification network is defined using Eq. (15).

$$\text{Max}_{PVL} = 1, \text{Min}_{PVL} = \frac{1}{\sqrt{n}}, \text{Confidencethreshold} = \frac{\text{Max}_{PVL} + \text{Min}_{PVL}}{2} \quad (15)$$

$$P_{\text{structural\_member}_j} = \{p_i = (x_i, y_i, z_i, FL)\} \in R^4 : \begin{cases} \text{if } FL = \text{beam} : \{\text{Length} = \text{Max}(\dim_{1,j}, \dim_{2,j}), \text{Width} = \text{Min}(\dim_{1,j}, \dim_{2,j}), \text{Height} = (\dim_{3,j})\} \\ \text{if } FL = \text{column} : \{\text{Length} = (\dim_{3,j}), \text{Width} = \text{Min}(\dim_{1,j}, \dim_{2,j}), \text{Height} = \text{Max}(\dim_{1,j}, \dim_{2,j})\} \\ \text{if } FL = \text{bracing} : \{\text{Length} = \text{Max}(\dim_{1,j}, \dim_{2,j}), \text{Width} = \text{Min}(\dim_{1,j}, \dim_{2,j}), \text{Height} = (\dim_{3,j})\} \end{cases} \quad (17)$$

Point cloud slices with higher classification confidence than the confidence threshold will be semantically labeled. However, point cloud slices with lower classification confidence than the confidence threshold will be labeled as “not identified” to be further manually investigated by the user to decide whether these points are part of structural members or they are a part of non-structural members possessing geometric attributes of structural members.

### 3.4. Dimension acquisition

To obtain the dimensions of each detected structural member point cloud ( $P_{\text{structural\_member},j} = \{p_i = (x_i, y_i, z_i, FL)\} \in R^4$ ), first, the point cloud is rotated using the PCA algorithm and Eq. (1) to align with the global coordinate system xyz. Then, the main dimensions of each detected structural member point cloud are obtained by circumscribing a bounding box around it. The dimensions of the circumscribed bounding box for each detected structural member point cloud are calculated using Eq. (16).

$$\text{Boundingbox}_j : \begin{cases} \dim_{1,j} = x_{\text{Max},j} - x_{\text{Min},j} \\ \dim_{2,j} = y_{\text{Max},j} - y_{\text{Min},j} \\ \dim_{3,j} = z_{\text{Max},j} - z_{\text{Min},j} \end{cases} \quad (16)$$

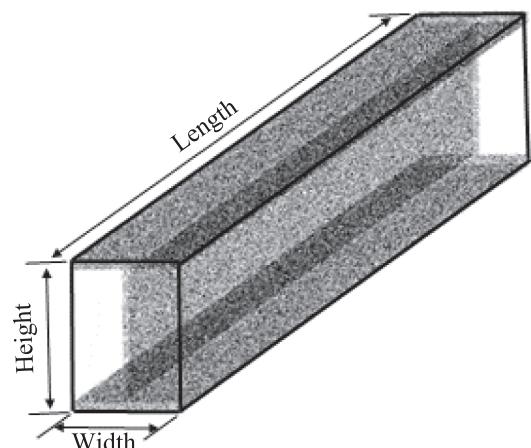
where  $x_{\text{Min},j}$  and  $x_{\text{Max},j}$  are the minimum and maximum X-axis coordinate of  $j^{\text{th}}$  detected structural member point cloud,  $y_{\text{Min},j}$  and  $y_{\text{Max},j}$  are the minimum and maximum Y-axis coordinate of  $j^{\text{th}}$  detected structural member point cloud,  $z_{\text{Min},j}$  and  $z_{\text{Max},j}$  are the minimum and maximum Z-

axis coordinate of  $j^{\text{th}}$  detected structural member point cloud, and  $j$  goes from 1 to the number of detected structural member point clouds.

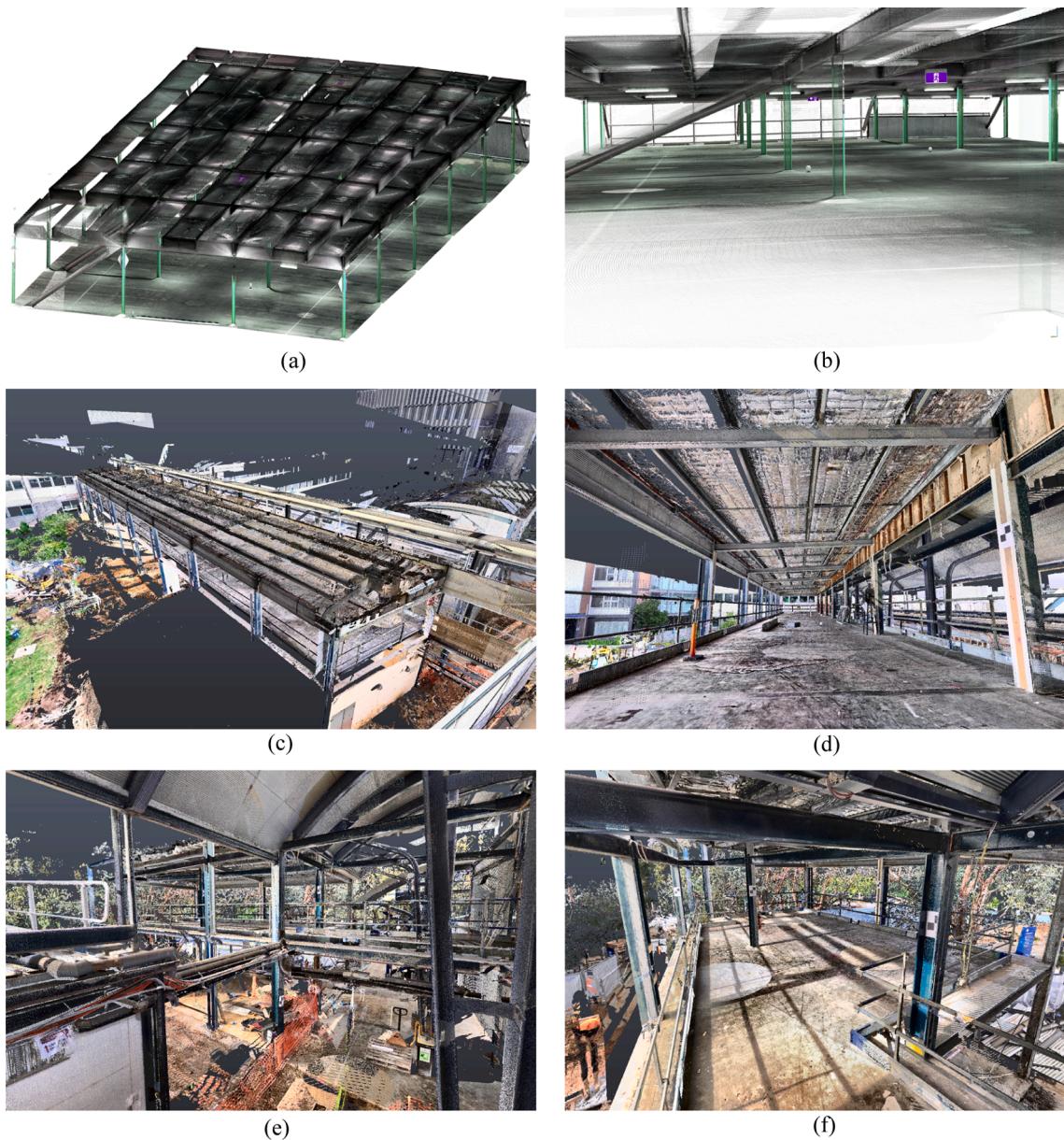
The dimensions are labeled based on the obtained structural label and contextual knowledge of designing structural members in buildings. In the building structural design concept, beams are often intended to have a height more significant than the width to provide better bending moment resistance. Also, columns under axial loading are usually designed with square cross-section dimensions to provide better structural support and practical buildability. Thus, the structural members’ length, height, and width are obtained using Eq. (17); an example is depicted in Fig. 6.

## 4. Experiments and results

The experiments and results section includes data collection, method implementation, and performance evaluation. The data collection stage



**Fig. 6.** Example of a bounding box for beam.

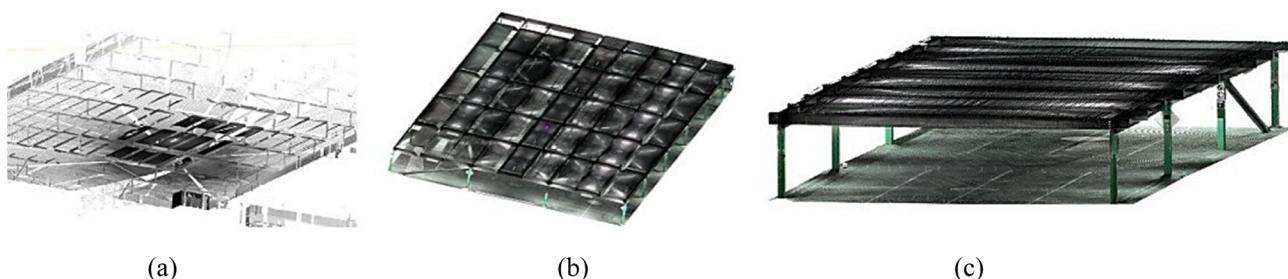


**Fig. 7.** (a,b) Monash University N1 carpark building point cloud, (c,d) Monash Smart Manufacturing Hub laboratory building point cloud, (e,f) point cloud of the educational building in Monash University.

presents the scanning procedure of three real-world building point clouds. Then, the implementation details of the method on the real-world point cloud datasets are discussed in the method implementation stage. Finally, the performance evaluation stage discusses the precision metrics of the proposed method.

#### 4.1. Data collection

To validate the performance of the proposed end-to-end structural members dimensional quality inspection, validation experiments were performed on three real-world building point clouds with varying levels



**Fig. 8.** Registration of the carpark building (a) one scan, (b) full model, and (c) subsampled point cloud.

of noises, outliers, and geometric features in different life cycle stages of buildings, as shown in Fig. 7.

For the first real-world building point cloud, multiple scans were taken by a terrestrial laser scanner (FARO® Focus M70) from the interior of the Monash University N1 carpark building. The scanner was positioned in the middle of each four columns for the maximum coverage of structural members from each side. After obtaining 18 scans of the building with  $\frac{1}{4}$  resolution and 4X quality settings, an automatic registration process was performed by using FARO SCENE Software (Autodesk Recap and Leica Cyclone software can also be used) to stitch the scans together, as shown in Fig. 8. Based on the settings mentioned above, the registration step for the whole building took 8–10 h using a desktop computer (Intel i7-9700 CPU @ 3.00 GHz, 32 GB RAM, and 500 GB), resulting in a point cloud with  $3.6 \times 10^8$  points (hereinafter referred to as the carpark building). The carpark case study building represents the applicability of the proposed method for regular building framing systems in the operation phase of the building life cycle. Other than structural members, the carpark point cloud includes non-structural objects such as traffic signs, electrical equipment, etc., a high level of occlusion due to scanning blind spots for some structural members, and noises caused by moving objects in the building.

Using the same scanner resolution and quality setting, the second real-world building point cloud was obtained from the renovation site of a laboratory facility in Monash University (Monash Smart Manufacturing Hub) consisting of  $3.1 \times 10^8$  points (hereinafter referred to as the laboratory building). The laboratory case study building represents the applicability of the proposed method in the refurbishment phase of the building life cycle. This case study includes structural members, construction material scattered in the point cloud, plumbing objects such as pipes, and electrical systems. The laboratory building was chosen to demonstrate the capability of the proposed method in detecting and measuring structural members in an unconventional structural system, including sloped and non-prismatic beams.

Finally, the third real-world building point cloud was obtained from the construction site of an educational facility at Monash University consisting of  $2.8 \times 10^8$  points (hereinafter referred to as the educational building). The educational building case study has a high level of noise, occlusion, and outliers, and the entire geometry of structural members in the boundary of the building was not captured during the scanning due to the restrictions associated with construction sites. The educational case study building represents the applicability of the proposed method in the construction phase of the building life cycle. The educational building was selected to demonstrate the capability of the proposed method in detecting and measuring structural members in point clouds with high levels of noise and missing data, along with an unconventional structural system consisting of diagonal (zigzag) beams. The details of each building are shown in Fig. 7 and Table 1.

#### 4.2. Method implementation

The proposed end-to-end dimensional quality inspection was run using a Tesla K80 GPU. For visualization of results, CloudCompare software has been used. For the remainder of this section, specific examples from the carpark building point cloud at each step of the developed methodology are included.

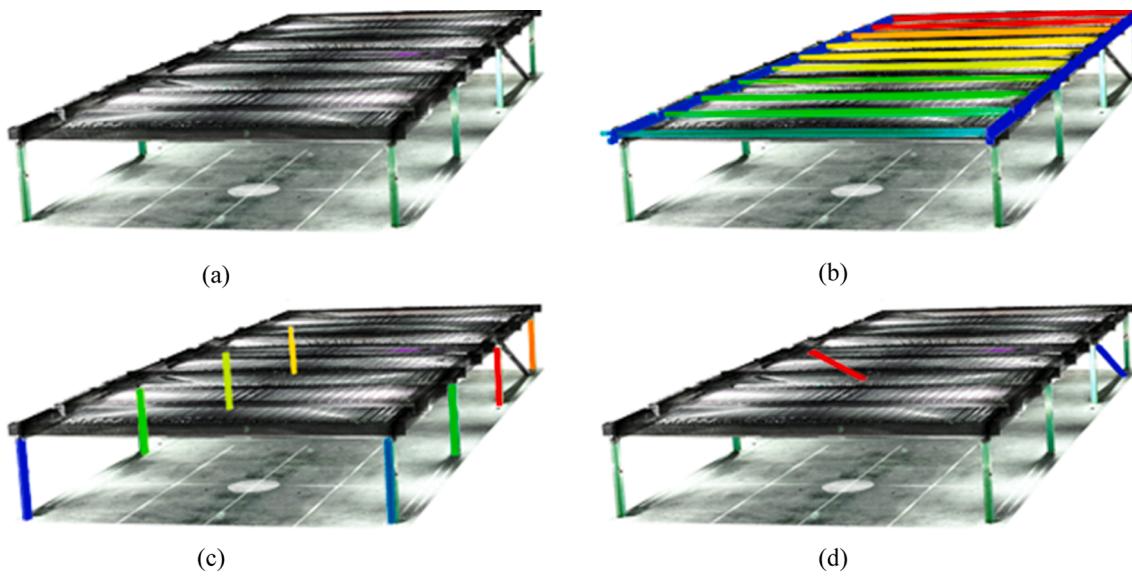
The point cloud was divided into  $N_{hor\_slice} = 50$  horizontal slices using a slice thickness  $t_{hor\_slice} = 50$  mm. Then, the list  $L_{hor\_slice}$  containing the number of points in each slice was obtained with  $\sigma_{L_{hor\_slice}} = 105126$  and

$\mu_{L_{hor\_slice}} = 47020$ . The *threshold* value for detecting potential beam points  $P_{pot\_beam}$  based on Eq. (4) was defined based on the standard normal distribution of 90% probability that a horizontal slice does not contain potential beam points considering the height of beams to be 10% of the total floor height. Thus,  $threshold = 1.645$  was set to detect potential beam points. In the next step, DBSCAN was utilized to merge the potential beam points of each floor. DBSCAN requires two main inputs of  $\epsilon_{hor}$  and  $N_{min\_points\_hor}$ . While contextual knowledge and point cloud specifications are essential in defining  $\epsilon_{hor}$  values and  $N_{min\_points\_hor}$ , this research implemented general rules discussed in [88] for DBSCAN inputs to avoid subjectiveness. Therefore,  $\epsilon_{hor} = 0.04$  and  $N_{min\_points\_hor} = 6$  were set for DBSCAN to merge the potential beam points of each floor in the carpark case study. In the next step, The point cloud of potential beam points was divided into  $N_{ver\_slice} = 486$  vertical slices using a vertical slice width  $w_{ver\_slice} = 50$  mm to separate beams in the x direction from beams in the y direction. For detecting potential column points, a search radius of 0.7 m around each point was used for point normal estimation with a tolerance of 5% for detecting vertical members. The number of considered adjacent points affects the performance of normal vector estimation, as per discussed in [89]. The *threshold* for labeling points as vertical was set 0.3 degrees, as recommended in AS/NZS 5131:2016[90] for the permissible inclination of columns. The  $\epsilon_{ver} = 0.04$  and  $N_{min\_points\_ver} = 6$  were set for DBSCAN to separate potential column instances in the carpark case study. Finally, potential bracing members were segmented based on the conditions of Eq. (10) and Eq. (11). The results of geometric segmentation on the carpark building point cloud are shown in Fig. 9. A summary of the input values for the proposed method is shown in Table 2.

For semantic segmentation steps, four steel section categories of channels, universal beams, rectangular hollow sections, and circular hollow sections with a total number of 5336 synthetic models for training and 1824 synthetic models for validation were used. The obtained 3D models of steel sections were automatically converted into point clouds with 10,000 points with a random level of noise and occlusion as specified in section 3.3.1. The network parameters were tuned, and the dropout rate of 0.7 on the last fully connected layer was found to be optimum. Adam optimizer with a learning rate of 0.001 was used for the network [70]. As such, the network was trained from scratch for 150 epochs using a batch size of 32 so that the loss function converges during the training process. The training time takes 15–18 h to converge on average based on the proposed configuration. The trained network can detect structural section shapes in all of the investigated case study buildings. Also, the trained network can be updated to detect structural shapes in any other point clouds by adding appropriate training datasets without needing to be trained from scratch. The confusion matrix corresponding to the results of the classification network's performance in the validation dataset is shown in Fig. 10. The classification network has an average accuracy of 94% across different section shapes. The accuracy of the classification network in circular cross-section shapes is lower than the average, and the classification network confuses them with rectangular cross-section shapes. After the manual observation of the testing and validation datasets, it was noted that after adding random noise to the point clouds, the cross-section shape of circular shapes changes into a rectangular shape, which causes confusion for the classification network. After training the classification network, possible structural members detected in the geometric segmentation step were sliced using 10,000 neighboring points to be similar to the training dataset point cloud point

**Table 1**  
Physical specifications and dimensions of case study buildings.

Building	Width (m)	Length (m)	Ceiling height (m)	No. of beams	No. of columns	No. of bracings	Sections
Carpark building	10.2	24.3	2.5	12	8	2	Universal beam – circular hollow section
Laboratory building	6.0	35.0	2.5	17	14	0	Universal beam
Educational building	12.3	23.6	2.5	7	12	0	Universal beam



**Fig. 9.** Geometric segmentation of carpark study building (a) raw point cloud, (b) extraction of potential beam points, (c) extraction of potential column points, and (d) extraction of potential bracing points.

**Table 2**

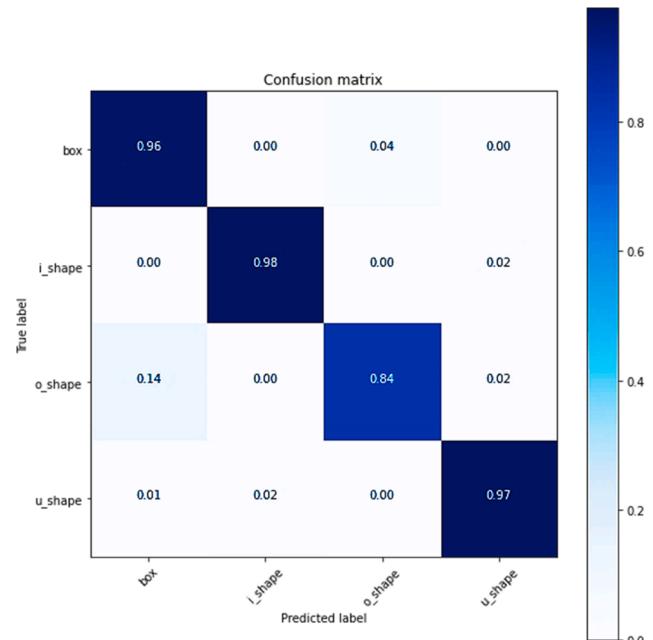
A summary of the input values for the proposed method.

Parameter	Input value	Reference
$t_{hor\_slice}$	50 mm	Contextual knowledge of structural design
The threshold value in Eq. (4)	1.645	Statistical analysis knowledge
DBSCAN( $\epsilon_{hor}$ )	0.04	[88]
DBSCAN( $N_{min\_points\_hor}$ )	6	[88]
$w_{ver\_slice}$	50 mm	Contextual knowledge of structural design
Normal estimation search radius	0.7 m	[89]
The threshold value in Eq. (8)	0.3 degrees	AS/NZS 5131:2016[90]
DBSCAN( $\epsilon_{ver}$ )	0.04	[88]
DBSCAN( $N_{min\_points\_ver}$ )	6	[88]

numbers. Next, the probability vector length for the prediction of each slice was calculated. Considering four different section shapes used in this study, the minimum value for probability vector length, i.e., lowest network confidence for prediction, would be 0.5. Therefore, the threshold of 0.75 for the lowest acceptable network confidence for prediction was set. Finally, the dimensions of detected structural members were obtained by applying a bounding box around each member. Fig. 11 depicts the results obtained after the implementation of preprocessing, geometric segmentation, and semantic segmentation steps on the carpark building. While the preprocessing step missed to filter out some parts of the floor and ceiling (Fig. 11b), the geometric segmentation step was capable of showing a high level of robustness for filtering non-structural and noise points from the point cloud (Fig. 11c). However, some non-structural objects possessing geometric definitions of structural members, highlighted with red color, remained in the point cloud after the geometric segmentation step, which were removed by the semantic segmentation step (Fig. 11d).

#### 4.3. Performance evaluation

The performance of the proposed end-to-end dimensional quality inspection of structural members was evaluated in two different aspects. First, the point-wise accuracy of the method in extracting structural members is evaluated. Next, a quantitative accuracy analysis was



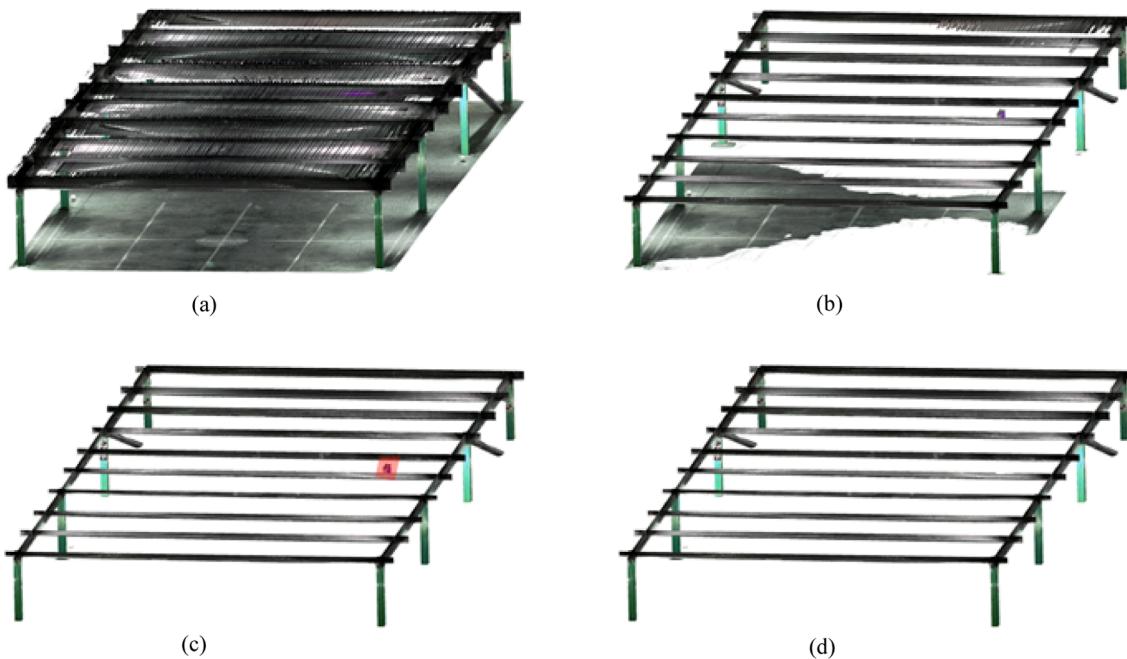
**Fig. 10.** Classification network performance in the validation dataset.

conducted to compare the structural members' dimensions obtained from the proposed method with the as-is dimensions acquired from manual measurements.

##### 4.3.1. Classification performance

The performance of the proposed method in classifying structural members is evaluated by quantifying the cross-section shape similarities and a point-wise Final Label (FL) comparison of extracted building structural members with respect to the ground truth [91].

To quantify the cross-section shape similarities, the obtained SL of each extracted structural member is compared with the ground truth cross-section shapes. A correct prediction is defined as points in which the confidence of the classification network is higher than the pre-defined PVL threshold of 0.75, and the predicted SL is the same as the ground truth cross-section shape label. Also, points with classified cross-



**Fig. 11.** (a) Raw input point cloud, (b) resulted point cloud from the preprocessing step, (c) resulted point cloud from the geometric segmentation step, and (d) resulted point cloud from the semantic segmentation step.

section shapes with PVL lower than 0.75 are labeled as “Not identified”. The “Not identified” label determines non-structural points with the geometrical definition of structural members labeled as a potential structural member in the geometric segmentation step without possessing a common cross-section shape of structural members. While “Not identified” label is a filtration mechanism for non-structural members, a high number of points with a “Not identified” label can also be a sign of classification network failure due to either having inadequate, nonrepresentative, and poor-quality training datasets or failure of the training process. Therefore, the point-wise accuracy of the semantic segmentation step for detecting the cross-section shapes of structural members is calculated by, first, considering points with “Not identified” label as correct prediction and then, points with “Not identified” label are counted as incorrect prediction, as shown in [Table 3](#). It can be seen that the proposed method had an average accuracy higher than 93.5%, with no single accuracy lower than 92.52% across all three real-world case buildings. However, the accuracy was reduced in real-world case buildings with higher noise levels and occlusions, such as the educational building. Also, considering the non-significant difference between considering points with the “Not identified” label as correct and incorrect prediction, it can be concluded that the semantic segmentation network trained with synthetically generated datasets performed well. Also, the results of the semantic segmentation step along with a comparison to ground-truth structural sections, are shown in [Fig. 12](#). Also, [Fig. 13](#) illustrates the PVL of the semantic segmentation network for case study buildings. It is observed that the semantic segmentation network

faces difficulties in classifying structural members at beam/column connection areas and the end of structural members. Another observation is that the level of correct classification of bracing members in the carpark building is lower than other structural members. Also, the performance of the proposed classification network dropped in the boundary structural members in the educational and laboratory buildings.

Moreover, the performance of the proposed method for obtaining the point-wise FL of building structural members with respect to the ground truth is evaluated in terms of Completeness, Correctness, and Quality of the results, as proposed in [\[91\]](#) and shown in Eq. [\(18\)](#).

$$\begin{aligned} \text{Completeness} &= \frac{|TP|}{|TP| + |FN|}, \quad \text{Correctness} = \frac{|TP|}{|TP| + |FP|}, \quad \text{Quality} \\ &= \frac{|TP|}{|TP| + |FN| + |FP|} \end{aligned} \quad (18)$$

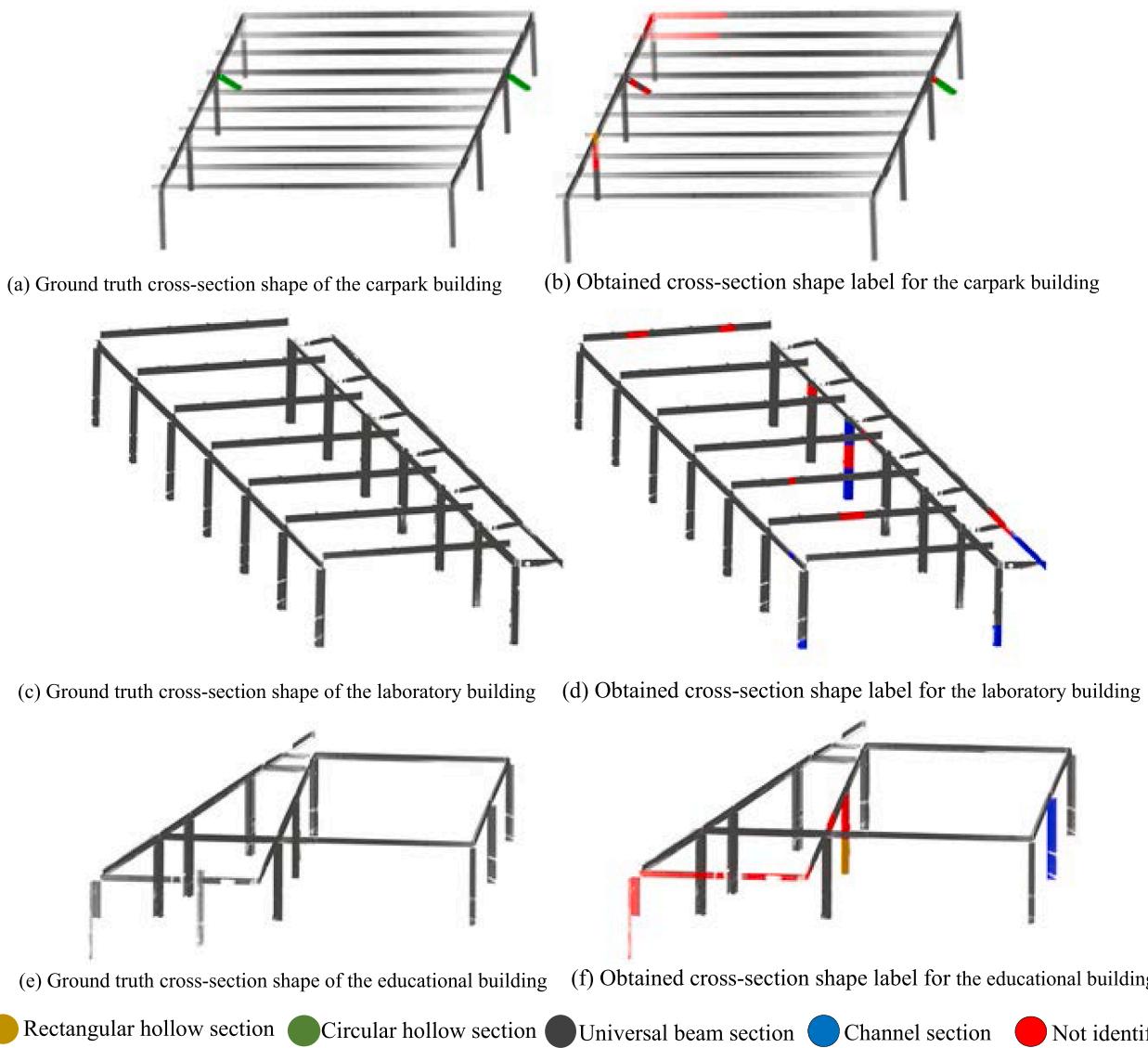
where the values of True Positive (TP), False Positive (FP), and False negative (FN) are obtained as proposed in [\[92\]](#). The obtained results from the point-wise FL evaluation are shown in [Table 4](#), in which the proposed method was able to extract beams with completeness, correctness, and quality no smaller than 95.5%, 95.9%, and 93.8% across case buildings. Also, columns were extracted with completeness, correctness, and quality no smaller than 90.1%, 97.8%, and 91.3% across case buildings. Finally, the proposed method could extract bracings with completeness, correctness, and quality no smaller than 85.4%, 95.2%, and 84.3% across case buildings. The proposed method demonstrated satisfactory performance for extracting structural members in real-world case buildings. Yet, the performance dropped in bracing members in the carpark building and columns in the educational building, which will be further investigated in the Discussion and Limitations section.

#### 4.3.2. Dimension acquisition performance

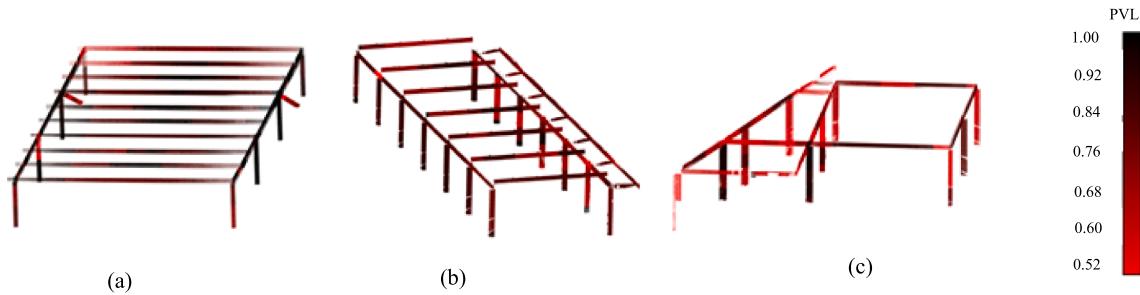
To identify the accuracy of the proposed method for obtaining structural members’ dimensions, the ground truth of the real-world buildings was manually measured three times, and the average value was reported. Based on the quality of measurement tools and the accuracy of the responsible measuring personnel, a tolerance of  $\pm 15$  mm was obtained from manual measurement of ground truth dimensions,

**Table 3**  
Point-wise accuracy of the classification algorithm in real-world case buildings.

Building	Accuracy with the “Not identified” label as correct prediction (%)	Accuracy considering the “Not identified” label as incorrect prediction (%)
Carpark building	96.18	94.45
Laboratory building	95.43	93.74
Educational building	94.76	92.52
Average	95.45	93.57



**Fig. 12.** The comparison of ground truth cross-section shape of structural members in (a) carpark building, (c) laboratory building, and (e) educational building with the obtained cross-section shape label from semantic segmentation step for (b) carpark building, (d) laboratory building, and (f) educational building.



**Fig. 13.** PVL for (a) carpark building, (b) laboratory building, and (c) educational building.

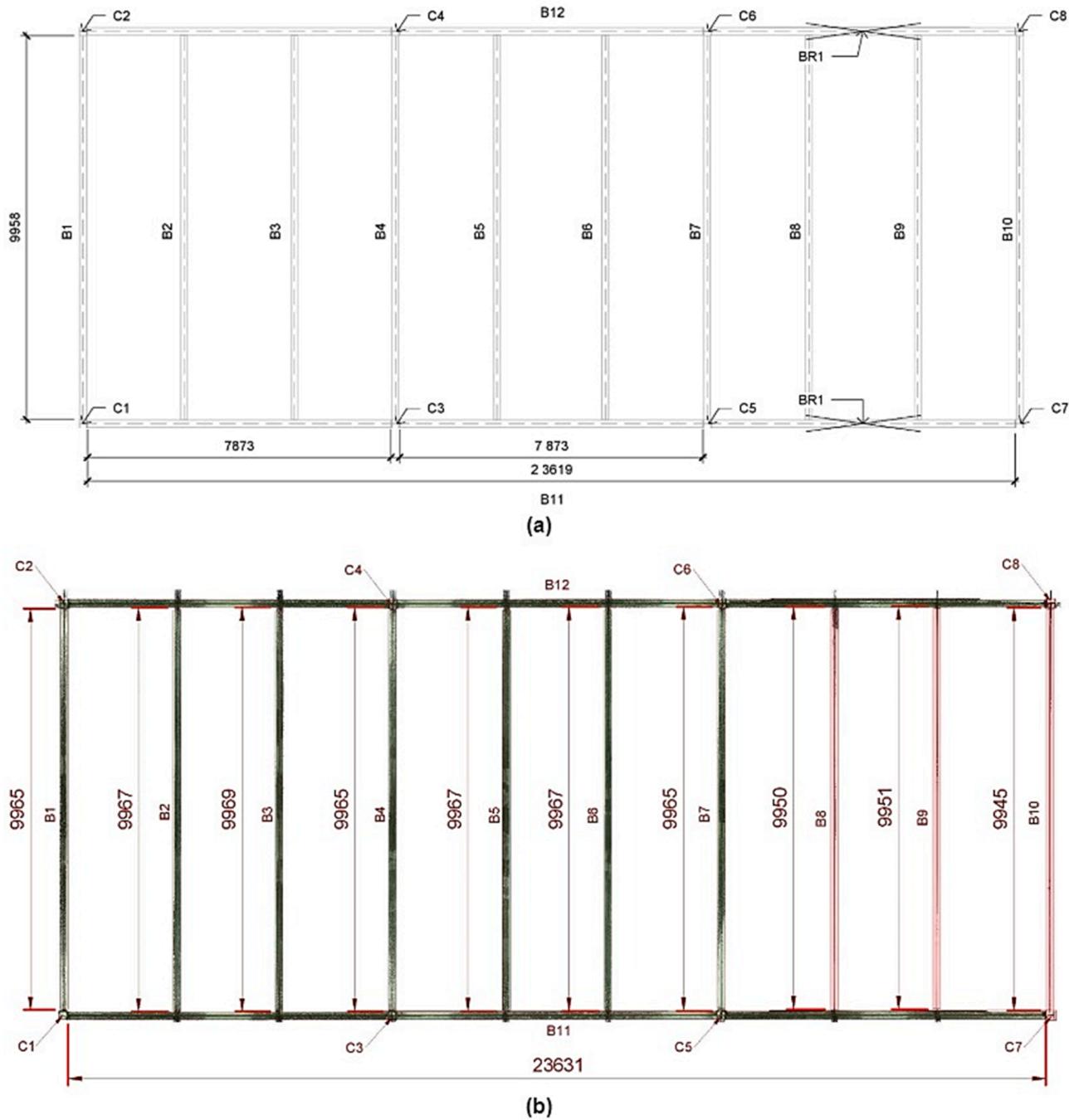
which is consistent with tolerance values reported in previous literature [82,93]. The ground truth models of carpark, laboratory, and educational building are shown in Fig. 14a, Fig. 15a, and Fig. 16a, respectively. Also, the models obtained from the proposed method along with the obtained dimensions are shown in Fig. 14b, Fig. 15b, and Fig. 16b, respectively. Details on the obtained dimensions of buildings and a comparison between ground truth data and obtained sizes are shown in Table 5, Table 6, and Table 7.

After implementing the proposed method on case study buildings, a tolerance of  $\pm 14.11$  mm for beams,  $\pm 14.13$  mm for columns, and  $\pm 54.46$  mm for bracing members in the carpark building was obtained. Also, the model had a tolerance of  $\pm 10.91$  mm for beams and  $\pm 11.11$  mm for columns in the laboratory building. Moreover, a tolerance of  $\pm 10.47$  mm for beams and  $\pm 13.73$  mm for columns was obtained in the educational building. It should be noted that the width and length of B8, B9, B10, C7, and C8 members in the carpark building and C5 members in

**Table 4**

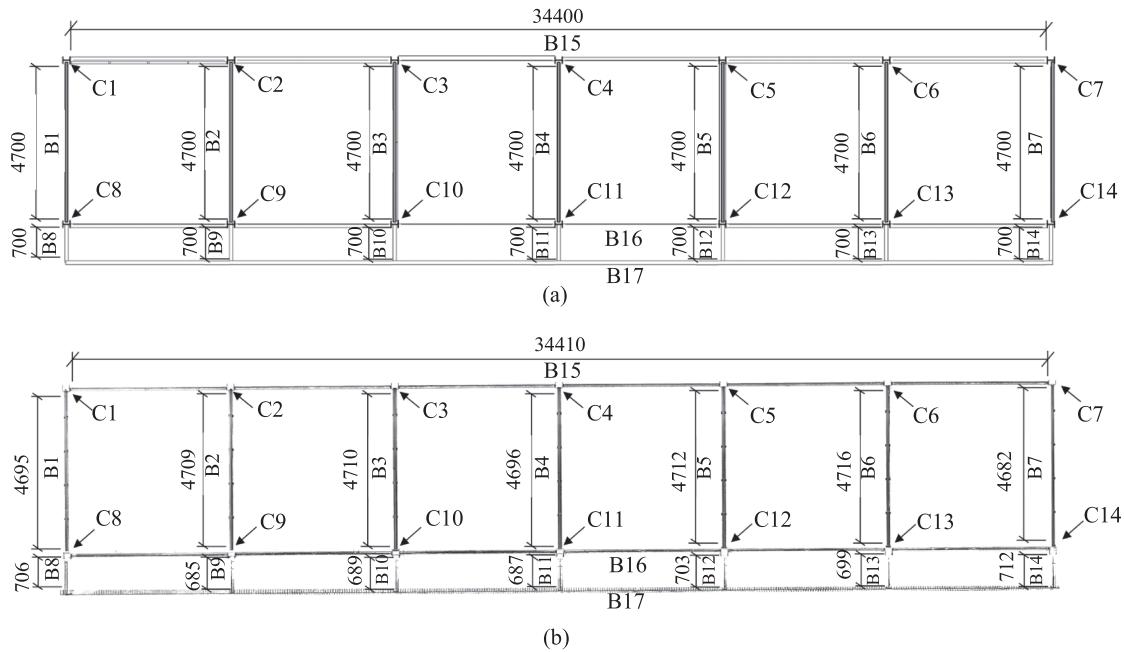
Point-wise evaluation of FL obtained from the proposed method.

Case Building	Beam			Column			Bracing		
	Comp.	Corr.	Quality	Comp.	Corr.	Quality	Comp.	Corr.	Quality
Carpark building	97.6%	99.3%	98.4%	97.1%	98.2%	96.2%	85.4%	95.2%	84.3%
Laboratory building	96.0%	96.6%	93.8%	96.3%	98.1%	97.0%	NA	NA	NA
Educational building	95.5%	95.9%	95.6%	90.1%	97.8%	91.3%	NA	NA	NA

**Fig. 14.** (a) Ground truth dimensions of the carpark building, and (b) automatically obtained dimensions from the carpark building point cloud.

the educational building were not reported as they had not been correctly captured in the point cloud dataset. Also, member C12 in the educational building was not detected during the geometric segmentation step. Considering that the manual measurement tolerance for

structural members was  $\pm 15$  mm, the experiments showed that the proposed method could provide accurate geometric information for beams and columns. However, the obtained geometric information for bracing members in the carpark building is bigger than the manual



**Fig. 15.** (a) Ground truth dimensions of the laboratory building, and (b) automatically obtained dimensions from the laboratory building point cloud.

measurement tolerance, which will be further investigated in the Discussion and Limitations section.

#### 4.4. Discussion and limitations

The experiments assessed the performance of the proposed end-to-end dimensional quality inspection method in different case study buildings with varying geometric features, noise, occlusion, and outliers levels. Despite encouraging results for dimensional quality inspection, a few observations were noted to be further discussed and analyzed here.

The proposed method is capable of filtering out the non-structural objects in the dataset. However, issues could arise where non-structural objects are attached to the structural members. For instance, traffic signs attached to a column can cause variations between the obtained dimensions and ground truth data, which results in bigger section dimensions for some of the columns in the carpark building point cloud. Also, the effects of noises, outliers, and missing points are equally important, if not more important, to the geometric complexity of buildings. To quantify such factors' impact on the proposed method's performance, two spans of the laboratory building point cloud with different noise levels and outliers were compared. While the percentage of points within the acceptable PVL range ( $0.75 < \text{PVL} < 1$ ) is similar between the two spans, the classification network had a higher percentage of predicted points with high confidence ( $0.9 < \text{PVL} < 1$ ) in the span with lower levels of noises and occlusions  $P(\text{Low noise model})$  than the span with higher levels of noises and occlusions  $P(\text{High noise model})$  in the laboratory building. Fig. 17 depicts the impact of noises, outliers, and missing data by comparing the PVL of the  $P(\text{Low noise model})$  with  $P(\text{High noise model})$ . Considering the results of Table 3 and Fig. 17, it is concluded that having a high level of noise, missing data, and outliers in the point cloud results in lower confidence in the classification network; however, the overall classification accuracy of the proposed method remains the same.

PVL operates as a controlling mechanism for filtering out the classified points with low confidence. Fig. 18 depicts the impact of the minimum acceptable PVL value on the number of correctly classified points, incorrectly classified points, and "Not-identified" points. While setting a high value as the acceptable PVL threshold leads to picking points with the highest classification confidence and lowest number of incorrectly classified points, it also reduces the number of correctly

classified points. On the contrary, a low value as the acceptable PVL threshold leads to a high number of incorrectly classified points. The optimum value of PVL is identified by calculating the ratio of incorrectly classified points over the correctly predicted points. The trend found in Fig. 18b, Fig. 18d, and Fig. 18f demonstrated that the optimal value of the PVL threshold happens at a local minimum in the incorrectly classified points over the correctly predicted points graph. Though the global minima occurred at high PVL values, using such PVL threshold values leads to a low number of correctly classified points, as depicted in Fig. 18a, Fig. 18c, and Fig. 18e. Based on the data from Fig. 18, the optimum value for the acceptable PVL threshold depends on the level of noise, occlusions, and outliers in the point cloud dataset. While the optimum value for the acceptable PVL threshold for point clouds with lower levels of noises, occlusions, and outliers was obtained to be 0.775, as shown in Fig. 18b, point clouds with a higher level of noise, occlusions, and outliers required a stricter acceptable PVL threshold of 0.75 as depicted in Fig. 18d and Fig. 18f.

One crucial factor impacting the accuracy of the classification network is the number of samples in the training dataset. This study's proposed method demonstrated synthetically generated samples' capability to train the classification network. While it is believed that, ideally, increasing the number of training samples results in improvements in the overall performance of the classification network, it should be considered that generating redundant training samples is computationally demanding and might adversely impact the performance of the classification network. Thus, the classification network proposed in this study was trained with different sample numbers within the range of 100 to 5336 samples, and their performance in the carpark building was compared in terms of the percentage of points within the acceptable PVL range as depicted in Fig. 19. The performance of the classification network eventually saturated after using 800 training samples demonstrating the minimum number of training samples. Also, the maximum accuracy was obtained using 5336 training samples. Considering the obtained fixed trend of classification network's performance and impact of other factors such as noises and outliers in the point cloud, it can be concluded that adding more training samples is unlikely to yield better results.

Moreover, in all of the case study buildings, the obtained dimensions for beams show high accuracy. That is due to the low level of occlusion and attached non-structural objects for beams. However, compared to

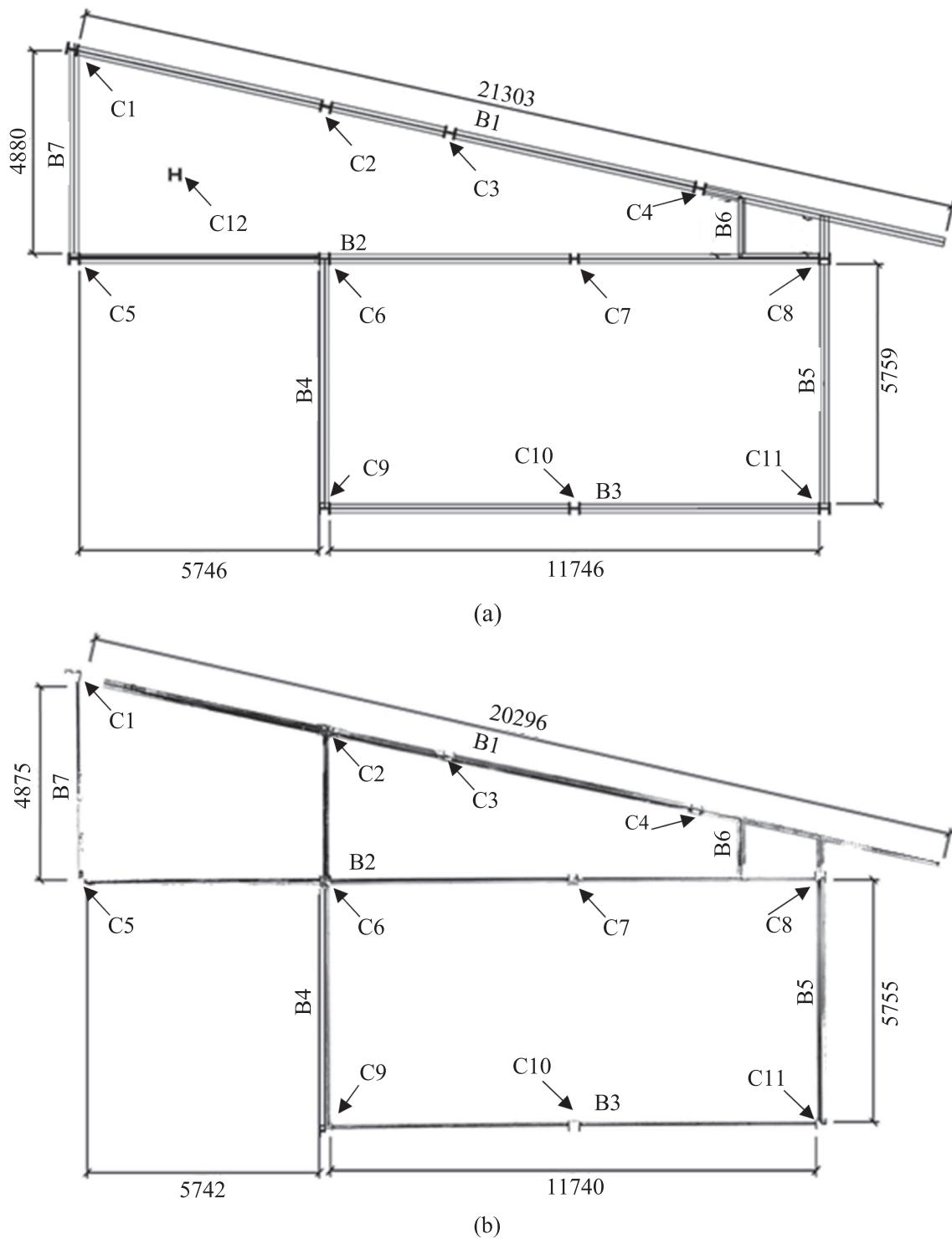


Fig. 16. (a) Ground truth dimensions of the educational building, and (b) automatically obtained dimensions from the educational building point cloud.

laboratory and educational building, the accuracy of beams in carpark case study buildings is lower. The main reason is that the top plate is missing in the point cloud; therefore, the beams' height is smaller in the point cloud than in the ground truth measurements. Also, this phenomenon caused some parts of beam sections to have a "T" section shape instead of a universal beam, which increased the confusion rate of the classification network. Moreover, the proposed method demonstrated lower confidence in classifying structural members in specific areas, such as beam/column connections and the end of structural members. During the manual measurements, it was noted that the end of structural

members is subjected to a gradual change in section size, which increased the confusion of the classification network. Also, due to the stepwise processing and removal of potential points during the geometric segmentation phase, beam/column connection points are removed from the point cloud as a part of possible beam points. Therefore, a gap is created in the middle of possible column points, which increases the confusion of the classification network. In addition, column C10 in the educational building and C12 in the laboratory building are incorrectly classified as a "Channel" section. After manual observations of the point cloud, it was noted that only half of these

**Table 5**

Comparison between the obtained dimensions and ground truth data for the carpark building.

Member mark	Width (mm)			Height (mm)			Length (mm)		
	Ground truth	Obtained	Variation	Ground truth	Obtained	Variation	Ground truth	Obtained	Variation
B1	171	182	11	356	330	26	9958	9965	7
B2	171	160	11	356	335	21	9958	9967	9
B3	171	185	14	356	340	16	9958	9969	11
B4	171	176	5	356	331	25	9958	9965	7
B5	171	181	10	356	340	16	9958	9967	9
B6	171	156	15	356	327	29	9958	9967	9
B7	171	145	26	356	340	16	9958	9965	7
B8	171	NA	NA	356	NA	NA	9958	NA	NA
B9	171	NA	NA	356	NA	NA	9958	NA	NA
B10	171	NA	NA	356	NA	NA	9958	NA	NA
B11	191	200	9	457	441	16	23,619	23,631	12
B12	191	170	21	457	430	27	23,619	23,631	12
C1	203	210	7	203	211	8	2200	2195	5
C2	203	195	8	203	194	9	2200	2189	11
C3	203	211	8	203	214	11	2200	2190	10
C4	203	210	7	203	210	7	2200	2196	4
C5	203	179	24	203	170	33	2200	2186	14
C6	203	182	21	203	152	51	2200	2168	32
C7	203	NA	NA	203	NA	NA	2200	NA	NA
C8	203	NA	NA	203	NA	NA	2200	NA	NA
BR1	193.7	153	40.7	193.7	144	49.7	6000	5910	90
BR2	193.7	167	26.7	193.7	151	42.7	6000	5923	77

**Table 6**

Comparison between the obtained dimensions and ground truth data for the laboratory building.

Member mark	Width (mm)			Height (mm)			Length (mm)		
	Ground truth	Obtained	Variation	Ground truth	Obtained	Variation	Ground truth	Obtained	Variation
B1	150	142	8	200	205	5	4700	4695	5
B2	150	162	12	200	207	7	4700	4709	9
B3	150	163	13	200	198	2	4700	4710	10
B4	150	164	14	200	196	4	4700	4696	4
B5	150	163	13	200	212	12	4700	4712	12
B6	150	148	2	200	214	14	4700	4716	16
B7	150	136	14	200	194	6	4700	4682	18
B8	150	162	12	200	185	15	700	706	6
B9	150	159	9	200	186	14	700	685	15
B10	150	130	20	200	196	6	700	689	11
B11	150	140	10	200	199	1	700	687	13
B12	150	163	13	200	180	20	700	703	3
B13	150	164	14	200	189	11	700	699	1
B14	150	153	3	200	191	9	700	712	12
B15	150	170	20	200	213	13	34,400	34,410	10
B16	150	167	17	200	216	16	34,400	34,423	23
B17	150	130	20	200	185	15	34,400	34,415	15
C1	300	288	12	300	309	9	2300	2312	12
C2	300	309	9	300	314	14	2300	2311	11
C3	300	310	10	300	289	11	2300	2314	14
C4	300	306	6	300	295	5	2300	2306	6
C5	300	297	3	300	309	9	2300	2298	2
C6	300	309	9	300	288	12	2300	2294	6
C7	300	280	20	300	280	20	2300	2315	15
C8	300	321	21	300	306	6	2300	2285	15
C9	300	290	10	300	304	4	2300	2306	6
C10	300	294	6	300	288	12	2300	2280	20
C11	300	316	16	300	316	16	2300	2309	9
C12	300	310	10	300	284	16	2300	2284	16
C13	300	290	10	300	309	9	2300	2321	21
C14	300	316	16	300	304	4	2300	2309	9

columns had been captured during the scanning, which was similar to a “Channel” section. Also, column C7 in the educational building was incorrectly classified as a “Box” section due to the wooden boards attached. Finally, column C12 in the educational building was not detected as a potential column member during the geometric segmentation step. After manual observations, it was noticed that column C12 in the educational building had been placed to support the ceiling load of another building and was not a part of the educational building framing system. Therefore, it did not satisfy the spatial relationships

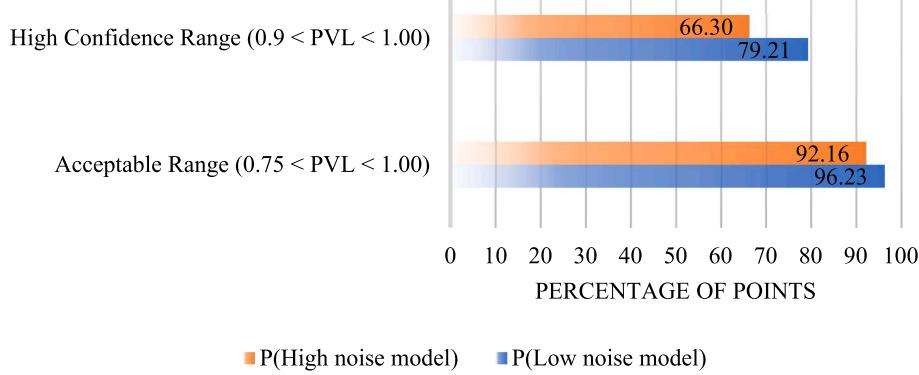
between columns and beams, as per Eq. (9), to be labeled as a potential column member.

The performance of the proposed method in obtaining dimensions was also affected by the attached non-structural objects along with changes in section sizes, such as the existence of column footing and pedestal. As a result, columns got a shorter length and larger section dimensions than those found in ground truth data in the carpark building. Moreover, C5 and C6 columns have a lower height value than the ground truth dimensions due to the lack of scanned data from the

**Table 7**

Comparison between the obtained dimensions and ground truth data for the educational building.

Member mark	Width (mm)			Height (mm)			Length (mm)		
	Ground truth	Obtained	Variation	Ground truth	Obtained	Variation	Ground truth	Obtained	Variation
B1	110	115	5	220	230	10	21,303	21,296	7
B2	110	112	2	220	236	16	17,692	17,682	10
B3	110	103	7	220	235	15	11,746	11,740	6
B4	110	100	10	220	205	15	5759	5755	4
B5	110	125	15	220	201	19	5759	5755	4
B6	110	123	13	220	236	16	1285	1289	4
B7	110	115	5	220	226	6	4880	4875	5
C1	250	240	10	250	243	7	2300	2295	5
C2	250	235	15	250	259	9	2300	2306	6
C3	250	254	4	250	257	7	2300	2289	11
C4	250	256	6	250	243	7	2300	2291	9
C5	NA	NA	NA	NA	NA	NA	NA	NA	NA
C6	250	242	8	250	246	4	2300	2296	4
C7	250	246	4	250	258	8	2300	2297	3
C8	250	256	6	250	256	6	2300	2307	7
C9	250	258	8	250	241	9	2300	2297	3
C10	250	241	9	250	253	3	2300	2296	4
C11	250	242	8	250	256	6	2300	2294	6
C12	NA	NA	NA	NA	NA	NA	NA	NA	NA

**Fig. 17.** Comparison of classification PVL between noiseless and noisy synthetic case study building.

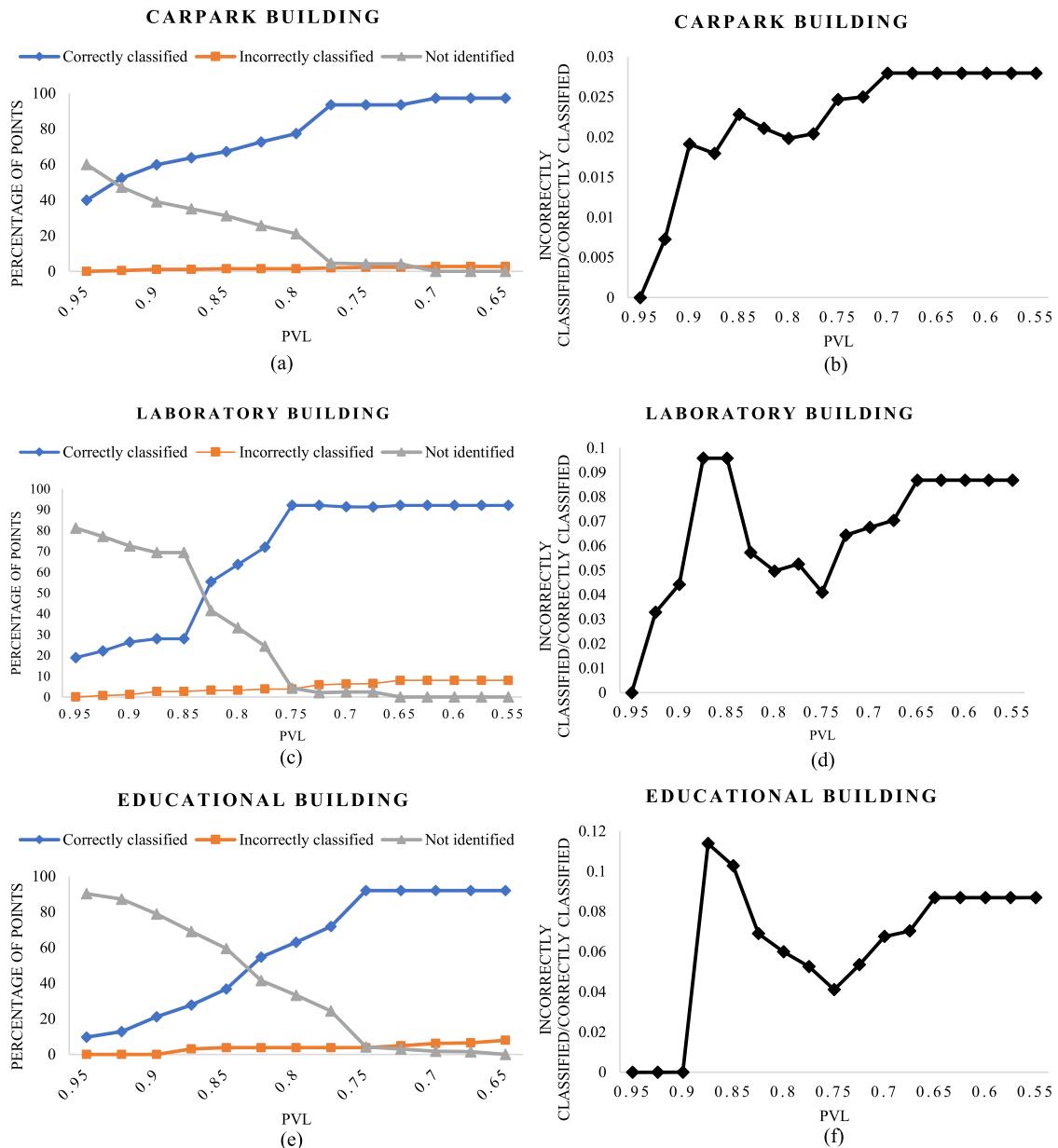
right span of the carpark building. Finally, bracing members demonstrated the highest level of dimensional variation compared to ground truth data. The manual observation of bracing members showed that the section type of these members is circular hollow sections. Due to the TLS device's location, laser beams could not reach the top and bottom parts of the circular hollow section in bracing members. Therefore, point clouds of these members possess a targeted occlusion that changes the overall geometry of the section into a couple of circular sectors with a high noise level. The targeted occlusion in bracing members, also, increased the confusion of the classification network for detecting bracing members. Bracing members are diagonal members that are further processed by applying the PCA algorithm to align with the main axes. As a result, the bounding box of bracing members can be distorted and not fully representative of the dimensions of the bracing member. Therefore, it can be concluded that a high occlusion level affects overall performance differently. Firstly, it has adverse effects on the classification network for semantic segmentation. Secondly, it changes the shape and dimensions of the member bounding box for dimension obtainment. Those issues demand an optimized scanning plan to reduce the level of occlusion in the point cloud.

Given the above discussions, the proposed method with synthetic data training has a few limitations. The synthetic training data for the classification network possess the same level of occlusion and noise. At the same time, it was noted that different members in the point cloud have different levels of noise and occlusion. A high discrepancy between the training and test datasets leads to a deficient performance for the

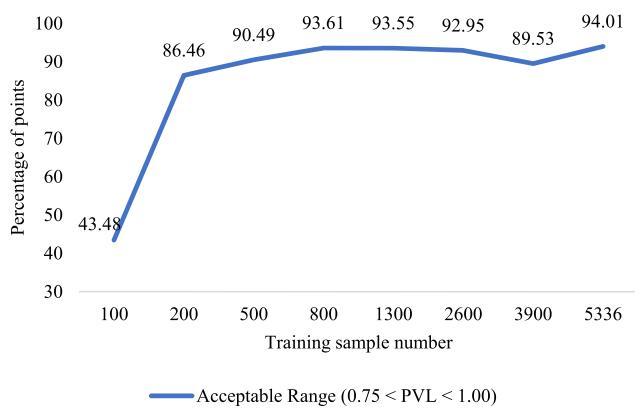
classification network. Also, the existence of non-structural members attached to structural members changes the captured cross-section shapes leading to incorrect cross-section classification results. Moreover, obtaining the dimensions of structural members via the proposed method mainly depends on the dimensions of the bounding box around each structural member. Therefore, a high noise level or the existence of attached non-structural objects could change the dimensions of the bounding box and reduce the accuracy of the proposed method. Also, construction projects are highly customizable and may not follow the common attributes of buildings stated in this work, such as sloped roofs or bent beams. While the proposed method can perform beyond the stated assumptions, i.e., sloped beams in the laboratory buildings and diagonal (zigzag) beams in the educational building, further geometric and spatial definitions may be required for non-conventional building designs.

## 5. Conclusion and future work

An automatic end-to-end dimensional quality inspection method for building structural members (beams, columns, and bracings) was introduced in this work. The proposed solution directly processed input point clouds that consist of an unorganized set of points defined by the coordinates (x,y,z). The method, first, segmented possible structural member points using geometric definitions and spatial relationships of each structural member. Potential beam points were segmented out based on the fact that beam points are often distributed horizontally in



**Fig. 18.** The impact of acceptable PVL threshold on the classification network performance for (a,b) carpark building, (c,d) laboratory building, and (e,f) educational building.



**Fig. 19.** The impact of training sample number on the performance of the classification network.

rectangular grids, while potential columns were segmented out as vertical members distributed beneath beams, and potential bracing members were defined as diagonal members distributed beneath beams. Next, a semantic segmentation step was performed to identify the cross-section shape of members found in the previous step. A framework for generating synthetic datasets was proposed to facilitate the adoption of classification networks and network training and validation purposes. The technique increased the repeatability and performance of the classification network for different scenarios. The labels given by geometric and semantic segmentation steps were grouped to detect structural members within the point cloud data. The proposed method could extract beams with completeness, correctness, and quality no smaller than 95.5%, 95.9%, and 93.8% across case buildings. Also, completeness, correctness, and quality no smaller than 90.1%, 97.8%, and 91.3% for column extraction and 85.4%, 95.2%, and 84.3% for bracing extraction across case buildings were obtained utilizing the proposed method. Finally, a bounding box was applied around each identified structural member to obtain the corresponding member length and

section dimensions. The proposed method demonstrated an acceptable performance for detecting different structural members within case study point clouds. The results indicated a tolerance of  $\pm 14.11$  mm for beams,  $\pm 14.13$  mm for columns, and  $\pm 54.46$  mm for bracing members in the carpark building. Also, the proposed model had a tolerance of  $\pm 10.91$  mm for beams and  $\pm 11.11$  mm for columns in the laboratory building. Moreover, a tolerance of  $\pm 10.47$  mm for beams and  $\pm 13.73$  mm for columns was obtained in the educational building. Overall, it is concluded that the proposed end-to-end method provides objective data for dimensional quality inspection of structural members using point clouds while standing out in terms of efficiency.

Future work further generalizes the proposed method by adding more geometric definitions, spatial relationships of structural members, and new section shape categories, such as "T" section shapes, to the classification network training dataset. Moreover, future work will consider an automated measure for adding variable levels of noise and occlusion to different categories in the training dataset based on the requirements of real-world applications. Also, a tolerance analysis will be conducted to predict the nature of occurred discrepancies and dimensional variability. Finally, different aspects of dimensional quality inspection, such as member positioning, will be investigated in future studies.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## Acknowledgements

The authors are grateful for support from the Australian Research Council (ARC) through the LIEF scheme (LE210100019). The assistance of the [ASCII Lab](#) members at Monash University is greatly appreciated.

## References

- [1] N.Z.S. Iso, ISO 8402:1994 Australian/New Zealand Standard Quality management and quality assurance—Vocabulary, 1994. [Online]. Available: <https://www.saiglobal.com/pdftemp/previews/osh/as/as8000/8400/8402.pdf>.
- [2] N. Johnston, S. Reid, An Examination of Building Defects in Residential Multi-owned Properties, 2019, pp. 1–62.
- [3] Infrastructure, A comprehensive assessment of America's Infrastructure, Asce, 2017, pp. 1-112. [Online]. Available: <https://www.infrastructurereportcard.org/>.
- [4] J.J. Lin, A. Ibrahim, S. Sarwade, M. Golparvar-Fard, Bridge Inspection with Aerial Robots: Automating the Entire Pipeline of Visual Data Capture, 3D Mapping, Defect Detection, Analysis, and Reporting, *J. Comput. Civ. Eng.* 35 (2) (2021) 04020064, [https://doi.org/10.1061/\(asce\)cp.1943-5487.0000954](https://doi.org/10.1061/(asce)cp.1943-5487.0000954).
- [5] M. Arashpour, A. Heidarpour, A. Akbar Nezhad, Z. Hosseini, Performance-based control of variability and tolerance in off-site manufacture and assembly: optimization of penalty on poor production quality, *Constr. Manag. Econ.* (2019) 1–13, <https://doi.org/10.1080/01446193.2019.1616789>.
- [6] M. Tavakolan, F. Mostafazadeh, S. Jalilzadeh Eirdmousa, A. Safari, K. Mirzaei, "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 (2022), 103485, <https://doi.org/10.1016/j.jobe.2021.103485>.
- [7] M.-K. Kim, Q. Wang, H. Li, Non-contact sensing based geometric quality assessment of buildings and civil structures: A review, *Autom. Constr.* 100 (2019) 163–179, <https://doi.org/10.1016/j.autcon.2019.01.002>.
- [8] L.-C. Wang, Enhancing construction quality inspection and management using RFID technology, *Autom. Constr.* 17 (4) (2008) 467–479, <https://doi.org/10.1016/j.autcon.2007.08.005>.
- [9] B.M. Phares, G.A. Washer, D.D. Rolander, B.A. Graybeal, M. Moore, Routine Highway Bridge Inspection Condition Documentation Accuracy and Reliability, *J. Bridge Eng.* 9(4) (2004) 403–413, doi: 10.1061/(asce)1084-0702(2004)9:4(403).
- [10] S. Paneru, I. Jeelani, Computer vision applications in construction: Current state, opportunities & challenges, *Autom. Constr.* 132 (2021), 103940, <https://doi.org/10.1016/j.autcon.2021.103940>.
- [11] Q. Wang, M.-K. Kim, Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018, *Adv. Eng. Inf.* 39 (2019) 306–319, <https://doi.org/10.1016/j.aei.2019.02.007>.
- [12] L. Yu, R. Tao, G. Lubineau, "Accurate 3D Shape, Displacement and Deformation Measurement Using a Smartphone," (in eng), *Sens.* (Basel, Switzerland) 19 (3) (2019) 719, <https://doi.org/10.3390/s19030719>.
- [13] V. Abolhasannejad, H. Xiaoming, N. Namazi, "Developing an Optical Image-Based Method for Bridge Deformation Measurement Considering Camera Motion," (in eng), *Sens.* (Basel, Switzerland) 18 (9) (2018) 2754, <https://doi.org/10.3390/s18092754>.
- [14] M. Wojtkowska, M. Kedzierski, P. Delis, Validation of terrestrial laser scanning and artificial intelligence for measuring deformations of cultural heritage structures, *Measurement* 167 (2021), 108291, <https://doi.org/10.1016/j.measurement.2020.108291>.
- [15] C. Poullis, A Framework for Automatic Modeling from Point Cloud Data, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (11) (2013) 2563–2575, <https://doi.org/10.1109/tpami.2013.64>.
- [16] K. Mirzaei, M. Arashpour, E. Asadi, H. Masoumi, Y. Bai, A. Behnood, 3D point cloud data processing with machine learning for construction and infrastructure applications: A comprehensive review, *Adv. Eng. Inf.* 51 (2022), 101501, <https://doi.org/10.1016/j.aei.2021.101501>.
- [17] Y. Arayici, An approach for real world data modelling with the 3D terrestrial laser scanner for built environment, *Autom. Constr.* 16 (6) (2007) 816–829, <https://doi.org/10.1016/j.autcon.2007.02.008>.
- [18] V. Pătrăucean, I. Armeni, M. Nahangi, J. Yeung, I. Brilakis, C. Haas, State of research in automatic as-built modelling, *Adv. Eng. Inf.* 29 (2) (2015) 162–171, <https://doi.org/10.1016/j.aei.2015.01.001>.
- [19] M. Arashpour, R. Wakefield, N. Blismas, E.W.M. Lee, A new approach for modelling variability in residential construction projects, *Australasian J. Constr. Econom. Build.* 13 (2) (2013) 83–92, <https://doi.org/10.5130/AJCEB.v13i2.3120>.
- [20] J. Jung, et al., Productive modeling for development of as-built BIM of existing indoor structures, *Autom. Constr.* 42 (2014) 68–77, <https://doi.org/10.1016/j.autcon.2014.02.021>.
- [21] M. Arashpour, J. Lamborn, P. Farzanehfar, Optimising collaborative learning and group work amongst tertiary students, in: 10th International Structural Engineering and Construction Conference, ISEC 2019, 2019: ISEC Press, doi: <https://doi.org/10.14455/ISEC.res.2019.121>.
- [22] M. Arashpour, J. Lamborn, P. Farzanehfar, Group Dynamics in Higher Education: Impacts of Gender Inclusiveness and Selection Interventions on Collaborative Learning, in: Claiming Identity Through Redefined Teaching in Construction Programs: IGI Global, 2020, pp. 42–60.
- [23] S.M. Iman Zolanvari, D.F. Laerer, Slicing Method for curved façade and window extraction from point clouds, *ISPRS J. Photogramm. Remote Sens.* 119 (2016) 334–346, <https://doi.org/10.1016/j.isprsjprs.2016.06.011>.
- [24] Y. Xiao, C. Feng, Y. Taguchi, V.R. Kamat, User-Guided Dimensional Analysis of Indoor Building Environments from Single Frames of RGB-D Sensors, *J. Comput. Civ. Eng.* 31 (4) (2017) 04017006, [https://doi.org/10.1061/\(asce\)cp.1943-5487.0000648](https://doi.org/10.1061/(asce)cp.1943-5487.0000648).
- [25] M. Arashpour, R. Wakefield, N. Blismas, E.W.M. Lee, "Framework for improving workflow stability: Deployment of optimized capacity buffers in a synchronized construction production," (in English), *Can. J. Civ. Eng.* 41 (12) (2014) 995–1004, <https://doi.org/10.1139/cjce-2014-0199>.
- [26] B. Riveiro, H. González-Jorge, M. Varela, D.V. Jauregui, Validation of terrestrial laser scanning and photogrammetry techniques for the measurement of vertical underclearance and beam geometry in structural inspection of bridges, *Measurement* 46 (1) (2013) 784–794, <https://doi.org/10.1016/j.measurement.2012.09.018>.
- [27] F. Dai, S. Dong, V.R. Kamat, M. Lu, Photogrammetry Assisted Measurement of Interstory Drift for Rapid Post-disaster Building Damage Reconnaissance, *J. Nondestr. Eval.* 30 (3) (2011) 201–212, <https://doi.org/10.1007/s10921-011-0108-6>.
- [28] L. Truong-Hong, R. Lindenberg, Extracting structural components of concrete buildings from laser scanning point clouds from construction sites, *Adv. Eng. Inf.* 51 (2022), 101490, <https://doi.org/10.1016/j.aei.2021.101490>.
- [29] Y. Kardovskyi, S. Moon, Artificial intelligence quality inspection of steel bars installation by integrating mask R-CNN and stereo vision, *Autom. Constr.* 130 (2021), 103850, <https://doi.org/10.1016/j.autcon.2021.103850>.
- [30] O.-S. Kwon, C.-S. Park, C.-R. Lim, A defect management system for reinforced concrete work utilizing BIM, image-matching and augmented reality, *Autom. Constr.* 46 (2014) 74–81, <https://doi.org/10.1016/j.autcon.2014.05.005>.
- [31] M. Arashpour, T. Ngo, H. Li, Scene understanding in construction and buildings using image processing methods: A comprehensive review and a case study, *J. Build. Eng.* 33 (2021), 101672, <https://doi.org/10.1016/j.jobe.2020.101672>.
- [32] C. Iglesias, J. Martínez, J. Taboada, Automated vision system for quality inspection of slate slabs, *Comput Ind* 99 (2018) 119–129, <https://doi.org/10.1016/j.compind.2018.03.030>.
- [33] E.B. Anil, P. Tang, B. Akinci, D. Huber, Deviation analysis method for the assessment of the quality of the as-is Building Information Models generated from point cloud data, *Autom. Constr.* 35 (2013) 507–516, <https://doi.org/10.1016/j.autcon.2013.06.003>.
- [34] M.-K. Kim, Q. Wang, J.-W. Park, J.C.P. Cheng, H. Sohn, C.-C. Chang, Automated dimensional quality assurance of full-scale precast concrete elements using laser scanning and BIM, *Autom. Constr.* 72 (2016) 102–114, <https://doi.org/10.1016/j.autcon.2016.08.035>.
- [35] M.-K. Kim, Q. Wang, S. Yoon, H. Sohn, A mirror-aided laser scanning system for geometric quality inspection of side surfaces of precast concrete elements,

- Measurement 141 (2019) 420–428, <https://doi.org/10.1016/j.measurement.2019.04.060>.
- [36] Q. Wang, Y. Tan, Z. Mei, Computational Methods of Acquisition and Processing of 3D Point Cloud Data for Construction Applications, *Arch. Comput. Meth. Eng.* 27 (2) (2019) 479–499, <https://doi.org/10.1007/s11831-019-09320-4>.
- [37] R. Maalek, D.D. Lichten, J.Y. Ruwanpura, Automatic Recognition of Common Structural Elements from Point Clouds for Automated Progress Monitoring and Dimensional Quality Control in Reinforced Concrete Construction, *Remote Sens. (Basel)* 11 (9) (2019) 1102, <https://doi.org/10.3390/rs11091102>.
- [38] R. Romero-Jarén, J.J. Arranz, Automatic segmentation and classification of BIM elements from point clouds, *Autom. Constr.* 124 (2021), 103576, <https://doi.org/10.1016/j.autcon.2021.103576>.
- [39] C. Fotsing, N. Menadjou, C. Bobda, Iterative closest point for accurate plane detection in unorganized point clouds, *Autom. Constr.* 125 (2021), 103610, <https://doi.org/10.1016/j.autcon.2021.103610>.
- [40] M.A. Fischler, R.C. Bolles, Random sample consensus, *Commun. ACM* 24 (6) (1981) 381–395, <https://doi.org/10.1145/358669.358692>.
- [41] K. Aitelkadi, D. Tahiri, E. Simonetto, I. Sebari, L. Polidori, Segmentation of heritage building by means of geometrical and radiometric components from terrestrial laser scanning, *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci. II-5/W1* (2013) 1–6, <https://doi.org/10.5194/isprsaannals-ii-5-w1-1-2013>.
- [42] B. Xu, W. Jiang, J. Shan, J. Zhang, L. Li, Investigation on the Weighted RANSAC Approaches for Building Roof Plane Segmentation from LiDAR Point Clouds, *Remote Sens. (Basel)* 8 (1) (2015) 5, <https://doi.org/10.3390/rs8010005>.
- [43] M. Previtali, et al., Automatic façade modelling using point cloud data for energy-efficient retrofitting, *Appl. Geomatics* 6 (2) (2014) 95–113, <https://doi.org/10.1007/s12518-014-0129-9>.
- [44] A.P. Dal Poz, M.S. Yano, Ransac-Based Segmentation for Building Roof Face Detection in Lidar Point Cloud, in: presented at the IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium, 2018/07, 2018. [Online]. Available: <https://doi.org/10.1109/igarss.2018.8518502>.
- [45] C. Miller, Z. Nagy, A. Schlueter, A review of unsupervised statistical learning and visual analytics techniques applied to performance analysis of non-residential buildings, *Renew. Sustain. Energy Rev.* 81 (2018) 1365–1377, <https://doi.org/10.1016/j.rser.2017.05.124>.
- [46] A. Aldoma, et al., Tutorial: Point Cloud Library: Three-Dimensional Object Recognition and 6 DOF Pose Estimation, *IEEE Rob. Autom. Mag.* 19 (3) (2012) 80–91, <https://doi.org/10.1109/mra.2012.2206675>.
- [47] T. Czerniawski, B. Sankaran, M. Nahangi, C. Haas, F. Leite, 6D DBSCAN-based segmentation of building point clouds for planar object classification, *Autom. Constr.* 88 (2018) 44–58, <https://doi.org/10.1016/j.autcon.2017.12.029>.
- [48] H. Aljumaily, D.F. Laefer, D. Cuadra, Urban Point Cloud Mining Based on Density Clustering and MapReduce, *J. Comput. Civ. Eng.* 31 (5) (2017) 04017021, [https://doi.org/10.1061/\(asce\)cp.1943-5487.0000674](https://doi.org/10.1061/(asce)cp.1943-5487.0000674).
- [49] R. Lu, I. Brilakis, C.R. Middleton, Detection of Structural Components in Point Clouds of Existing RC Bridges, *Comput. Aided Civ. Inf. Eng.* 34 (3) (2018) 191–212, <https://doi.org/10.1111/mice.12407>.
- [50] M.F. Ahmed, C.T. Haas, R. Haas, Automatic Detection of Cylindrical Objects in Built Facilities, *J. Comput. Civ. Eng.* 28 (3) (2014) 04014009, [https://doi.org/10.1061/\(asce\)cp.1943-5487.0000329](https://doi.org/10.1061/(asce)cp.1943-5487.0000329).
- [51] C. Mura, O. Mattausch, R. Pajarola, Piecewise-planar Reconstruction of Multi-room Interiors with Arbitrary Wall Arrangements, *Comput. Graphics Forum* 35 (7) (2016) 179–188, <https://doi.org/10.1111/cgf.13015>.
- [52] A. Dimitrov, M. Golparvar-Fard, Segmentation of building point cloud models including detailed architectural/structural features and MEP systems, *Autom. Constr.* 51 (2015) 32–45, <https://doi.org/10.1016/j.autcon.2014.12.015>.
- [53] A. Khaloo, D. Lattanzi, Robust normal estimation and region growing segmentation of infrastructure 3D point cloud models, *Adv. Eng. Inf.* 34 (2017) 1–16, <https://doi.org/10.1016/j.aei.2017.07.002>.
- [54] B. Koo, R. Jung, Y. Yu, Automatic classification of wall and door BIM element subtypes using 3D geometric deep neural networks, *Adv. Eng. Inf.* 47 (2021), 101200, <https://doi.org/10.1016/j.aei.2020.101200>.
- [55] M. Arashpour, V. Kamat, A. Heidarpour, M.R. Hosseini, P. Gill, Computer vision for anatomical analysis of equipment in civil infrastructure projects: Theorizing the development of regression-based deep neural networks, *Autom. Constr.* 137 (2022), 104193, <https://doi.org/10.1016/j.autcon.2022.104193>.
- [56] M. Arashpour, et al., Predicting individual learning performance using machine-learning hybridized with the teaching-learning-based optimization, *Comput. Appl. Eng. Educ.* (2022), <https://doi.org/10.1002/cae.22572>.
- [57] S. Wang, S. Suo, W.-C. Ma, A. Pokrovsky, R. Urtasun, Deep Parametric Continuous Convolutional Neural Networks, in: presented at the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018/06, 2018. [Online]. Available: <https://doi.org/10.1109/cvpr.2018.00274>.
- [58] Y. LeCun, et al., Backpropagation Applied to Handwritten Zip Code Recognition, *Neural Comput.* 1 (4) (1989) 541–551, <https://doi.org/10.1162/neco.1989.1.4.541>.
- [59] S.A. Bello, S. Yu, C. Wang, J.M. Adam, J. Li, Review: Deep Learning on 3D Point Clouds, *Remote Sens. (Basel)* 12 (11) (2020) 1729, <https://doi.org/10.3390/rs12111729>.
- [60] I. Coudron, S. Puttemans, T. Goedemé, P. Vandewalle, “Semantic Extraction of Permanent Structures for the Reconstruction of Building Interiors from Point Clouds,” in (eng), *Sensors (Basel, Switzerland)* 20 (23) (2020) 6916, <https://doi.org/10.3390/s20236916>.
- [61] W. Zhirong, et al., 3D ShapeNets: A deep representation for volumetric shapes,“ presented at the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015/06, 2015. [Online]. Available: <https://doi.org/10.1109/cvpr.2015.7298801>.
- [62] C.R. Qi, L. Yi, H. Su, L.J. Guibas, PointNet++: Deep hierarchical feature learning on point sets in a metric space, *Advances in Neural Information Processing Systems*, vol. 2017-Decem, pp. 5100–5109, 2017.
- [63] X. Wang, S. Liu, X. Shen, C. Shen, J. Jia, Associatively Segmenting Instances and Semantics in Point Clouds, in: presented at the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019/06, 2019. [Online]. Available: <https://doi.org/10.1109/cvpr.2019.00422>.
- [64] D. Maturana, S. Scherer, VoxNet: A 3D Convolutional Neural Network for real-time object recognition, in: presented at the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015/09, 2015. [Online]. Available: <https://doi.org/10.1109/iros.2015.7353481>.
- [65] C. Zhao, L. Sun, P. Purkait, T. Duckett, R. Stolk, “Dense RGB-D Semantic Mapping with Pixel-Voxel Neural Network,” (in eng), *Sensors (Basel, Switzerland)* 18 (9) (2018) 3099, <https://doi.org/10.3390/s18093099>.
- [66] B.-S. Kim, P. Kohli, S. Savarese, 3D Scene Understanding by Voxel-CRF, in: presented at the 2013 IEEE International Conference on Computer Vision, 2013/12, 2013. [Online]. Available: <https://doi.org/10.1109/iccv.2013.180>.
- [67] Y. Xu, W. Yao, L. Hoegner, U. Stilla, Segmentation of building roofs from airborne LiDAR point clouds using robust voxel-based region growing, *Remote Sens. Lett.* 8 (11) (2017) 1062–1071, <https://doi.org/10.1080/2150704x.2017.1349961>.
- [68] L. Wang, Y. Xu, Y. Li, Y. Zhao, “Voxel segmentation-based 3D building detection algorithm for airborne LiDAR data,” (in eng), *PLoS One* 13 (12) (2018) e0208996-e, <https://doi.org/10.1371/journal.pone.0208996>.
- [69] H. Su, S. Maji, E. Kalogerakis, E. Learned-Miller, Multi-view Convolutional Neural Networks for 3D Shape Recognition, in: presented at the 2015 IEEE International Conference on Computer Vision (ICCV), 2015/12, 2015. [Online]. Available: <https://doi.org/10.1109/iccv.2015.114>.
- [70] R. Q. Charles, H. Su, M. Kaichun, L. J. Guibas, PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation, in: presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017/07, 2017. [Online]. Available: <https://doi.org/10.1109/cvpr.2017.16>.
- [71] C. Xiang, C.R. Qi, B. Li, Generating 3D Adversarial Point Clouds, in: presented at the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019. [Online]. Available: <https://doi.org/10.1109/cvpr.2019.00935>.
- [72] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, B. Chen, PointCNN: Convolution on X-Transformed Points, 2018. [Online]. Available: <http://arxiv.org/abs/1801.07791>.
- [73] Y. Wang, Y. Sun, Z. Liu, S.E. Sarma, M.M. Bronstein, J.M. Solomon, Dynamic Graph CNN for Learning on Point Clouds, *ACM Trans. Graph.* 38 (5) (2019) 1–12, <https://doi.org/10.1145/3326362>.
- [74] E. Agapaki, I. Brilakis, CLOI-NET: Class segmentation of industrial facilities’ point cloud datasets, *Adv. Eng. Inform.* 45 (2020) p. 101121, doi: 10.1016/j.aei.2020.101121.
- [75] J.W. Ma, T. Czerniawski, F. Leite, Semantic segmentation of point clouds of building interiors with deep learning: Augmenting training datasets with synthetic BIM-based point clouds, *Autom. Constr.* 113 (2020), 103144, <https://doi.org/10.1016/j.autcon.2020.103144>.
- [76] Y. Perez-Perez, M. Golparvar-Fard, K. El-Rayes, Scan2BIM-NET: Deep Learning Method for Segmentation of Point Clouds for Scan-to-BIM, *J. Constr. Eng. Manag.* 147 (9) (2021) 04021107, [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0002132](https://doi.org/10.1061/(ASCE)CO.1943-7862.0002132).
- [77] A. Smith, R. Sarlo, Automated extraction of structural beam lines and connections from point clouds of steel buildings, *Computer-Aided Civil and Infrastructure Engineering*, <https://doi.org/10.1111/mice.12699> vol. 37, no. 1, pp. 110–125, 2022/01/01 2022, doi: <https://doi.org/10.1111/mice.12699>.
- [78] B. Wang, C. Yin, H. Luo, J.C.P. Cheng, Q. Wang, Fully automated generation of parametric BIM for MEP scenes based on terrestrial laser scanning data, *Autom. Constr.* 125 (2021), 103615, <https://doi.org/10.1016/j.autcon.2021.103615>.
- [79] M.-K. Kim, J.P.P. Thedja, Q. Wang, Automated dimensional quality assessment for formwork and rebar of reinforced concrete components using 3D point cloud data, *Autom. Constr.* 112 (2020), 103077, <https://doi.org/10.1016/j.autcon.2020.103077>.
- [80] C. Rausch, M. Nahangi, C. Haas, W. Liang, Monte Carlo simulation for tolerance analysis in prefabrication and offsite construction, *Autom. Constr.* 103 (2019) 300–314, <https://doi.org/10.1016/j.autcon.2019.03.026>.
- [81] N. Puri, E. Valero, Y. Turkan, F. Bosché, Assessment of compliance of dimensional tolerances in concrete slabs using TLS data and the 2D continuous wavelet transform, *Autom. Constr.* 94 (2018) 62–72, <https://doi.org/10.1016/j.autcon.2018.06.004>.
- [82] M.-K. Kim, H. Sohn, C.-C. Chang, Automated dimensional quality assessment of precast concrete panels using terrestrial laser scanning, *Autom. Constr.* 45 (2014) 163–177, <https://doi.org/10.1016/j.autcon.2014.05.015>.
- [83] Q. Wang, J.C.P. Cheng, H. Sohn, Automated Estimation of Reinforced Precast Concrete Rebar Positions Using Colored Laser Scan Data, *Comput. Aided Civ. Inf. Eng.* 32 (9) (2017) 787–802, <https://doi.org/10.1111/mice.12293>.
- [84] M. Nahangi, J. Yeung, C.T. Haas, S. Walbridge, J. West, Automated assembly discrepancy feedback using 3D imaging and forward kinematics, *Autom. Constr.* 56 (2015) 36–46, <https://doi.org/10.1016/j.autcon.2015.04.005>.
- [85] F. Bosché, Automated recognition of 3D CAD model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction, *Adv. Eng. Inf.* 24 (1) (2010) 107–118, <https://doi.org/10.1016/j.aei.2009.08.006>.
- [86] L. Li, F. Yang, H. Zhu, D. Li, Y. Li, L. Tang, An Improved RANSAC for 3D Point Cloud Plane Segmentation Based on Normal Distribution Transformation Cells, *Remote Sens. (Basel)* 9 (5) (2017) 433, <https://doi.org/10.3390/rs9050433>.

- [87] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, in: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), 1996. [Online]. Available: [www.aaai.org](http://www.aaai.org). [Online]. Available: [www.aaai.org](http://www.aaai.org).
- [88] Z. Akbari, R. Unland, Automated Determination of the Input Parameter of DBSCAN Based on Outlier Detection, in: L.I. Cham, I. Maglogiannis (Eds.), Artificial Intelligence Applications and Innovations, Springer International Publishing, 2016// 2016;, pp. 280–291.
- [89] Z. Yu, T. Wang, T. Guo, H. Li, J. Dong, Robust point cloud normal estimation via neighborhood reconstruction, *Adv. Mech. Eng.* 11(4) (2019) p. 1687814019836043, doi: 10.1177/1687814019836043.
- [90] in AS/NZS 5131:2016 - Structural steelwork - Fabrication and erection, ed: Standards Australia / Standards New Zealand, 2016.
- [91] L. Truong-Hong, D.F. Laefer, Quantitative evaluation strategies for urban 3D model generation from remote sensing data, *Comput. Graph.* 49 (2015) 82–91, <https://doi.org/10.1016/j.cag.2015.03.001>.
- [92] M. Rutzinger, F. Rottensteiner, N. Pfeifer, A Comparison of Evaluation Techniques for Building Extraction From Airborne Laser Scanning, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 2 (1) (2009) 11–20, <https://doi.org/10.1109/JSTARS.2009.2012488>.
- [93] J. Guo, Q. Wang, J.-H. Park, Geometric quality inspection of prefabricated MEP modules with 3D laser scanning, *Autom. Constr.* 111 (2020), 103053, <https://doi.org/10.1016/j.autcon.2019.103053>.