

Newly-developed structural digital twin modeling method for monitoring deviations of prefabricated cable domes based on 3D laser scanning

Ailin Zhang ^{a, b}, Hao Ma ^a, Xi Zhao ^{a, d, *}, Yanxia Zhang ^{a, d}, Jie Wang^a, Meini Su ^c

^a School of Civil and Transportation Engineering, Beijing University of Civil Engineering and Architecture, Beijing, China

^b Beijing University of Technology, Beijing, China

^c School of Engineering, University of Manchester, Manchester M1 7JR, UK

^d Beijing Energy Conservation & Sustainable Urban and Rural Development Provincial and Ministry Co-construction Collaboration Innovation Center, Beijing, China

ABSTRACT

Structural performance of prefabricated cable domes is highly sensitive to structural deviations. An accurate and thorough detection of the deviations therefore is important. In this paper, a newly-developed structural digital twin (SDT) modeling method based on three-dimensional (3D) laser scanning is proposed for deviation monitoring. The point cloud data (PCD) is inventively segmented for further analysis according to structural features. A new structural skeleton point recognition algorithm is then developed where layering and clustering of PCD are processed. Members are recognized automatically with joints of connections sequentially identified. The SDT of cable domes are formed, therefore. Deviation monitoring of cable dome is provided where the geometry of SDT is compared with design model. The accuracy and efficiency of the method are discussed thoroughly. The novelty and benefits of the proposed method exhibit. The study laid the foundation for high-fidelity analysis approach based on SDT to predict structural performance in future.

Keywords: structural digital twin, cable dome structure, deviation monitoring, 3D laser scanning, skeleton point recognition

1. Introduction

With the ongoing advancements in tensegrity research and prestressing construction technology, cable dome structures have gained increasing popularity among researchers and engineers[1]. Their appeal lies in their large spans, lightweight structures, short construction periods, and aesthetically pleasing appearances. However, the complicated and challenging construction of a cable dome attributable to the geometric sensitivity of the structure[3]. The geometric imperfections of a structure are generally due to the manufacturing imperfections, residual deformation from welding, joint displacements during installations, base settlement, and slack of prestressed cables[4].

These deviations can alter the prestressing levels of a structure and consequently reduce its rigidity and loading capacity[5]. In extreme cases, these changes can result in structural instability, leading to safety concerns[4]. Given the potential risks associated with geometric deviations, continuously monitoring and predicting the mechanical properties of cable–dome structures throughout their life cycle is crucial.

* **Correspondence author at:** School of Civil and Transportation Engineering, Beijing University of Civil Engineering and Architecture, Beijing, 100044, China

Correspondence author. Email addresses: zhaoxi@bucea.edu.cn (X. Zhao).

For cable dome structures characterized by large spans and intricate contour curvatures, traditional inspection methods, which rely heavily on manual labor or total station techniques, are time consuming and imprecise[6]. Therefore, many scholars have attempted to develop a better solution that guarantees efficient and accurate structural inspections[7]. The development of digital-twin technology has provided new insights into this concern. This technology involves creating precise digital replicas of real-world entities and incorporating real-world data into these models to offer various services[8]. Initially, the aerospace sector used the digital twin technology for informed management and decision-making across the entire lifecycle of aerospace products[9]. Over time, its applications have broadened to various industries, such as manufacturing and healthcare, where automation enhances accuracy and efficiency[10]. Researchers in civil engineering initially used accelerometers and strain gauges for structural health monitoring, and their feedback has prompted the construction of digital twins[11]. However, the structural performance is attributed to both the materials and geometry of the structure. Point sensors cannot provide high-throughput and high-fidelity geometric information for inspection and prediction with sparsely distributed points[12]. A structural digital twin (SDT) comprising accurate and thorough global geometric information of a structure is required to better understand the constructional imperfections and numerical simulations[13]. The realization of the SDT is delayed primarily attributable to the conflict between the need for a substantial amount of high-precision data to build SDTs and the inefficiencies and inaccuracies inherent in existing data acquisition and processing techniques[14].

For efficient and accurate data acquisition, noncontact measurement technologies such as three-dimensional (3D) laser scanning are widely adopted[15]. This technology has numerous advantages in structural applications: (1) it reduces data acquisition labor time by over 90%[16], and (2) it ensures higher data accuracy than other methods[17]. Point cloud data (PCD) gathered via 3D laser scanning serve as vital foundational information for SDT development[18]. Consequently, semi or fully automatic model-building techniques that cater to high-precision PCD processing have become popular[19]. Initially, the structural geometric information was extracted by matching the PCD with the design model[20]. With the increased need for automation, several commercial software programs, such as Bentley Descartes[21], Realworks[22], and Cyclone[23], offer tools for digital model construction. These methods and programs extract the characteristic details from the PCD and lay a groundwork for the development of advanced models. However, these processes are only partially automated and require manual interventions for refining[24]. Moreover, general modeling processes may not readily extract specific structural data essential for intricate designs, demanding extra input[25].

With the aim of automatic modeling, many methods for automated feature recognition and extraction have been proposed, such as the Hough transform[26], region growing[27], and random sample consensus (RANSAC) algorithm[29]. For large-scale PCD of structures with large dimensions and complex geometric features, size estimation and feature recognition require structural skeleton extraction[30]. Related skeleton extraction methods include rotational symmetry axis (ROSA)[31], Laplace contraction[32], and algorithms grounded on geometric characteristics[33]. These methods can refine the skeleton through topological refinement or screening to improve the accuracy of the model and reduce the effects of missing point clouds and noise[34]. Although these methods have many advantages, challenges arise when integrating accurate geometric features with advanced structural analyses[35]. For a meaningful structural analysis, a structured model replete with detailed member and connection data is imperative[37]. However, current modeling methods do not provide the feasible geometric features that are used in the pre-stress calculations of cable dome structures[38]. Moreover, modeling efficiency affects the benefits of SDT in structural construction and health monitoring[39]. This requires development of a specialized data structure from scratch tailored specifically for structural modeling and performance evaluation.

In this study, a new automated SDT modeling method was proposed to reconstruct the SDT of a cable dome structure using PCD. Large-scale PCD are segmented into symmetric modules to reduce the computational effort required for better analysis and applications. A new structural skeleton point recognition method was proposed. The PCD clusters are obtained by layering and clustering, and their centers are contracted to obtain the skeleton points. This method effectively extracts structural features while significantly reducing the number of PCD. Members were automatically recognized from the extracted skeleton points based on a standard skeleton model. Therefore, the geometric information of the structural members was efficiently obtained for SDT reconstruction based on the member skeleton points. A new data structure was employed for storing the structural twin models. The structure is used for efficiently storing and determining member information, while providing a data interface for further structural performance analysis.

The remainder of this paper is organized as follows: The experimental model employed in the study and details the PCD preparation process are outlined in Section 2. The theory and implementation of the proposed SDT modeling method are described in Sections 3–6. In Section 7, the accuracy and efficiency of the proposed SDT model are discussed. Finally, the summary of the study is provided in Section 8.

2. Background

In this section, the background of this study is introduced. The structural formation and member geometry of this complex spatial structure are provided in Section 2.1, while the method of obtaining the structural PCD using a terrestrial laser scanning system is explained in Section 2.2. The pre-processing process that supports the fully automatic and highly efficient construction of the SDT is detailed in Section 2.3.

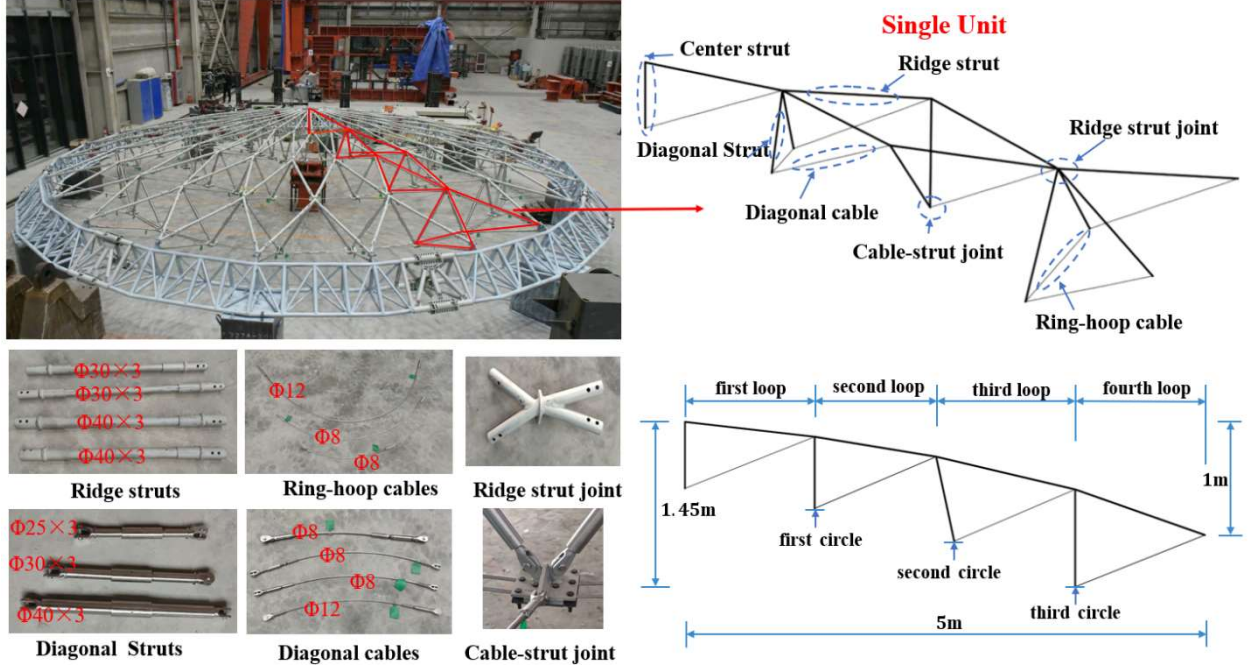


Fig. 1. Structural form and member information.

2.1. Structural information

The test model used in this study was a scaled-down version of a ridge-tube cable dome supplemented with annular struts, as shown in Fig. 1. The scale-down factor was set to 1:10. The cable dome had a diameter and height of 10 and 1 m, respectively. It was centrally symmetric and comprised 24 identical circumferential units. The unit featured three hoop cable loops and four additional loops for inclined cables.

2.2. 3D laser scanning

The structural PCD was obtained using a RIEGL laser-scanning system. This system encompasses the terrestrial laser scanner RIGEL-VZ1000 (as illustrated in Fig. 2) and PCD preprocessing software RiSCAN Pro. Panorama40 mode was selected for scanning. The measurement distance was 2.5–1400 m, the horizontal measurement angle was 360°, and the measurement accuracy reached 5 mm at 100 m.

A meticulous scanning plan was prepared beforehand to ensure the precision and completeness of the data while avoiding unnecessary redundancy in PCD processing; 12 scanning sites were predetermined in circumferential order:

two sites inside the structure were designated for capturing its overall exterior details; the remaining sites outside the structure were responsible for recording the intricate details of the inner members[40].

2.3. Pre-processing of PCD

First, the PCD was preprocessed using RiSCAN Pro software. This enabled the fully automatic and highly efficient reconstruction of the SDT. PCD pre-processing includes PCD registration and noise elimination.

PCD registration is a two-step process: initial global registration followed by finer registration. The initial registration was conducted manually, which provided a rough registration of the scans. Fine registration was conducted using an embedded algorithm, the iterative closest point. The registration of multiple scanned pieces satisfactorily reconstructed the large-span cable dome with controlled accuracy.

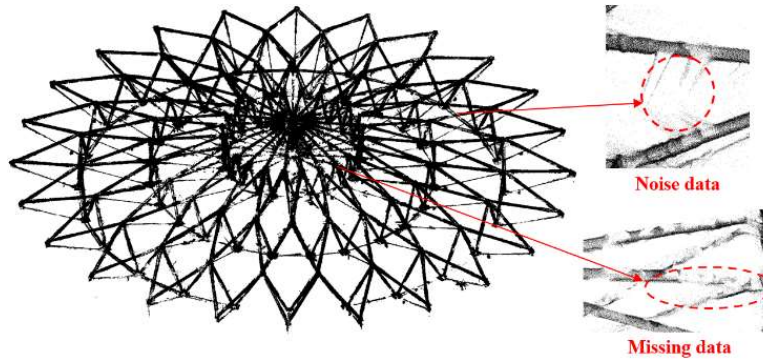


Fig. 2. PCD after denoising.

TABLE 1

PCD denoising information.

Parameter setting		Data volume	Time (s)
Number of neighboring points	6	15689612 (560.42 MB)	28.43
Standard deviation multiplier	1.00	12661192 (443.03 MB)	

After the completion of the PCD registration, PCD denoising was initiated. As shown in Table 1, statistical filtering was employed to remove noise points and change the number of neighboring points, and the standard deviation multiplier was employed to adjust the filtration intensity. As shown in Fig. 2, post denoising, PCD retained a minimal number of stubborn noise points with missing data for certain structural members.

3. PCD module segmentation

PCD module segmentation divides the preprocessed data of the entire structure into multiple modules that efficiently decrease computational effort for better analysis and applications.

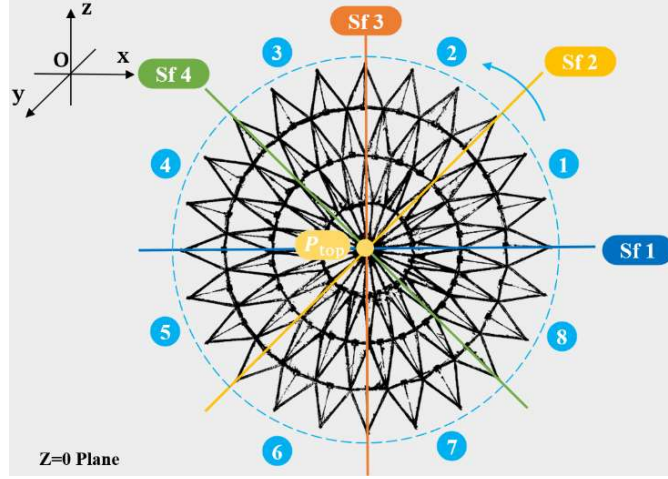


Fig. 3. Module segmentation process.

A module generally comprises several units smaller than the total number of units in a cable dome. The studied structure had 24 units that were segmented into eight modules with three units per module (Fig. 3). The segmentation process began after the scheme was set. The highest point, P_{top} , which is generally the top of the center strut, was set as a reference point at $(0,0,0)$ coordinate. The axis aligned with P_{top} and perpendicular to the X-Y plane was set as the Z-axis. A segmentation plane was generated perpendicular to the X-Y plane and at 45° increments towards the X-Z plane. Thus, four segmentation planes were generated, i.e., $Sf_i, i \in [1,4]$, and eight modules were obtained. For example, Module 1 was segmented into Sf_1 and Sf_2 , and it contained 85.86 MB of data, feasible for general computation and analysis. The data of the segmented modules were later stored in the form of a KD-tree, a k-dimensional binary tree for efficient PCD storage and searching.

Segmentation ensured no information loss; thus, the entity of structural data was guaranteed. The PCD of each ridge strut was confined to individual modules to eliminate redundancy.

4. Skeleton points recognition

A cable dome is a spatial structure, and its structural members cannot be simplified into a plane system for analysis. Therefore, its SDT must maintain a spatial configuration in which each cable and strut are located automatically and efficiently from the bulky PCD modules. The first step was to recognize the skeleton points (the centroids of the PCD segments) using the proposed layering and clustering algorithms.

4.1. Multi PCD layering

Layering is aimed at dividing the members into multiple length-determined segments (Fig. 4). The PCD layering algorithm comprises two rules for preserving the spatial geometrical features. In the first layering, the ridge struts are considered, where the PCD is searched under a polar coordinate system. In the second layering part, the remaining

cables and struts are considered, where the PCD segments are searched along the Z-axis in the Cartesian coordinate system.

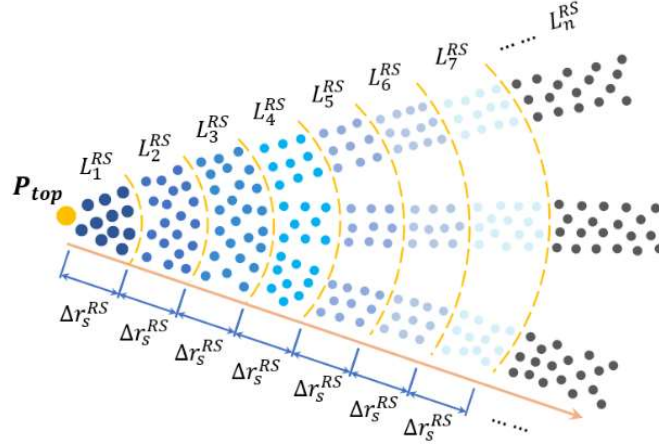


Fig. 4. PCD layering.

The ridge struts were spatially and circumferentially distributed outward by diffusion from the top of the center strut. The ridge-strut PCD was layered under the polar coordinate system, where the points of the layers were searched for under specific radii from P_{top} (Fig. 4). First, the total length R^{RS} , which is the distance between the peak P_{top} and farthest P_{far} , was calculated using Eq. (1). The total number of layering N^{RS} was determined in advance. This guaranteed both the computational efficiency and sufficient number of skeleton points in the recognition process. The searching radius increment (Δr_s^{RS}) was calculated using Eq. (2) and was assumed to be even. The superscript RS refers to the ridge strut.

$$R^{RS} = \sqrt{(x_b - x_t)^2 + (y_b - y_t)^2 + (z_b - z_t)^2} = \sqrt{x_b^2 + y_b^2 + z_b^2} \quad (1)$$

$$\Delta r_s^{RS} = R^{RS} / N^{RS} \quad (2)$$

where (x_t, y_t, z_t) and (x_b, y_b, z_b) are the coordinates of P_{top} and P_{far} , respectively. The P_{top} coordinates were (0,0,0). The N^{RS} was set to 250.

In layering, the radius of the points within a specific range are assumed to belong to the layer L^{RS} . The specific range was calculated using Eq. (3). The radius of point P, which is the distance between layered points P and P_{top} , was determined using Eq. (4). The layering window was then operated sequentially using the PCD of the ridge struts (Eq. (5)). The complete layering of ridge struts is shown in Fig. 5(a). Notably, other PCD, except of the ridge struts, were automatically deleted in the clustering section.

$$\begin{cases} R_{n-1}^{RS} = (n-1) \cdot \Delta r_s^{RS} \\ R_n^{RS} = n \cdot \Delta r_s^{RS} \end{cases}, n \in [1, N^{RS}] \quad (3)$$

$$r_p^{RS} = \sqrt{(x_p)^2 + (y_p)^2 + (z_p)^2} \quad (4)$$

$$L_n^{RS} = \{P | R_{n-1}^{RS} < r_p^{RS} < R_n^{RS}\}, n \in [1, N^{RS}] \quad (5)$$

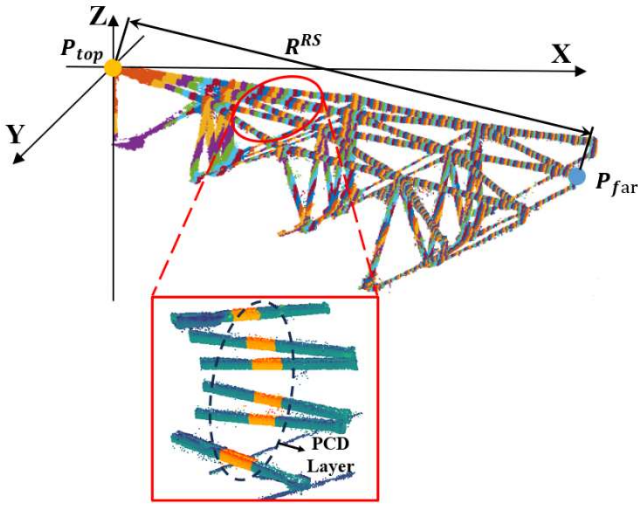
where r_p^{RS} represents the distance of point P from P_{top} , (x_p, y_p, z_p) represents the coordinates of point P.

The diagonal struts were vertically arranged in the structure. The diagonal-strut PCD was obtained considering the Cartesian coordinate system, where points of layers were searched under heights from the Z-axis projected points to the center P_{top} of the diagonal struts. The total height H^{DS} , which is the height between the peak P_{top} and lowest projected point P_{low} , was calculated using Eq. (6). The number of layering N^{DS} depends on the N^{RS} , where the ratio of N^{DS} to N^{RS} should align with the size of the structure. The skeleton points should be recognized at the same distance. The searching height increment was calculated using Eq. (7). The superscript DS denotes the diagonal strut.

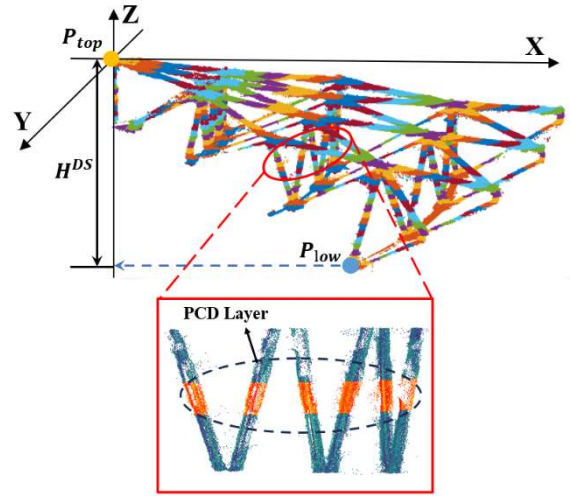
$$H^{DS} = |z_l - z_t| = |z_l| \quad (6)$$

$$\Delta h_s^{DS} = H^{DS} / N^{DS} \quad (7)$$

where z_l is the z coordinate of P_{low} . The N^{DS} was set to 50.



(a) Layering for ridge struts



(b) Layering for diagonal struts

Fig. 5. Multi PCD layering.

Similar to the ridge-strut layering, in the diagonal-strut layering, the heights of the projected points within a specific range were assumed to belong to the layer L^{DS} . The specific range was calculated using Eq. (8). The height of point P , which is the projected distance between points P and P_{top} , was obtained from Eq. (9). The layering window was then operated sequentially using the diagonal-strut PCD (Eq. (10)). The complete layering of diagonal struts is shown in Fig. 5(b). Note that PCDs other than diagonal struts were automatically deleted in the clustering section.

$$\begin{cases} H_{n-1}^{DS} = (n-1) \cdot \Delta h_s^{DS} \\ H_n^{DS} = n \cdot \Delta h_s^{DS} \end{cases}, n \in [1, N^{DS}] \quad (8)$$

$$h_p^{DS} = |z_p - z_t| = |z_p| \quad (9)$$

$$L_n^{DS} = \{P | H_{n-1}^{DS} < h_p^{DS} < H_n^{DS}\}, n \in [1, N^{DS}] \quad (10)$$

where h_p^{DS} represents the projected distance to the P_{top} of point P , and z_p represents the z-coordinate of point P .

4.2. Clustering of PCD layers

The layering points were further clustered and noise points were eliminated using the density-based spatial clustering of applications with noise (DBSCAN) algorithm. The DBSCAN algorithm defines points in the window of a circle at a specified radius as a cluster (Fig. 6). The layered points are filtered through the window at the determined density of the points. The layers formed by the structural members exhibits a high density of points. In contrast, the layers polluted with noise exhibit a low density of points. Thus, the DBSCAN was an appropriate choice.

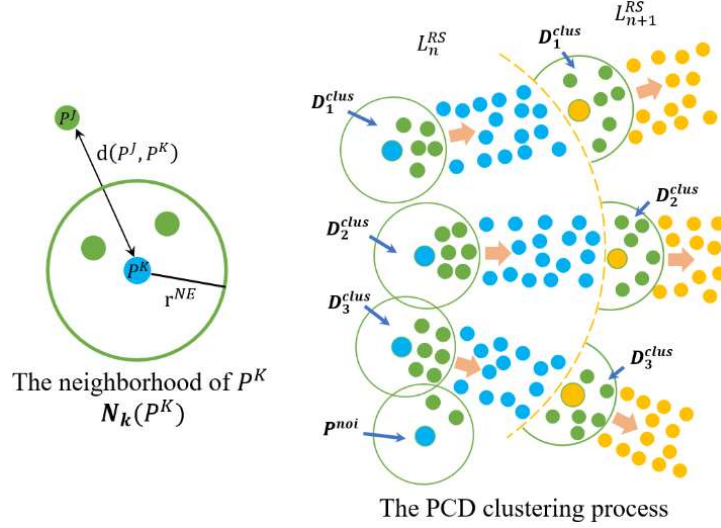


Fig. 6. Clustering of PCD layers.

Two parameters were determined in advance for the algorithm: 1) searching radius r^{NE} for searching the neighborhood for a point, and 2) threshold number M_{thr} of the points in a neighborhood for noise elimination. The r^{NE} was considered as 0.02 m (1.5 times the maximum radius of struts from trial and error). The M_{thr} was considered as $M_T/15000$, where M_T denotes the total number of PCD.

Points in a PCD layer were sorted by distance to P_{top} , from smallest to largest. First, a point P^K (a point nearest to P_{top}) was determined. The search algorithm then started computing the distance $d(P^K, P^J)$ between points P^K and other points P^J in a layer (Eq. (11)). The neighborhood $N_k(P^K)$ was composed of points with distances $d(P^K, P^J)$ within the searching radius r^{NE} , as shown in Eq. (12). The total number M_K of points in $N_k(P^K)$ was then counted. M_K was compared with the threshold M_{thr} (Eq. (13)). For $M_K < M_{thr}$, the points in the $N_k(P^K)$ were considered noise points D^{noi} and directly deleted. Otherwise, the points were considered valid member points, and a PCD cluster D^{clus} was formed for further analysis. The search was iterated until the new nearest P^K in the layer was found, excluding the points in $N_k(P^K)$ from the previous search.

$$d(P^K, P^J) = \sqrt{(x_K - x_J)^2 + (y_K - y_J)^2 + (z_K - z_J)^2}, J, K \in [1, M_T] \quad (11)$$

$$N_k(P^K) = \{P^J | d(P^K, P^J) \leq r^{NE}\} \quad (12)$$

$$\begin{cases} \mathbf{N}_k(P^K) = \mathbf{D}^{noi} & M_D < M_{th} \\ \mathbf{N}_k(P^K) = \mathbf{D}^{clus} & M_D > M_{th} \end{cases} \quad (13)$$

where (x_K, y_K, z_K) and (x_J, y_J, z_J) are the coordinates of points P^K and P^J , respectively.

Using PCD layering and clustering, the PCD clusters of members with different orientations were extracted uniformly and efficiently.

4.3. Central shrinkage for PCD clusters

The skeleton points were recognized by centrally shrinking the PCD clusters of the members, as detailed in this section. As shown in Fig. 7, the points in different colors represent the corresponding PCD clusters. Red points represent the target skeleton points.

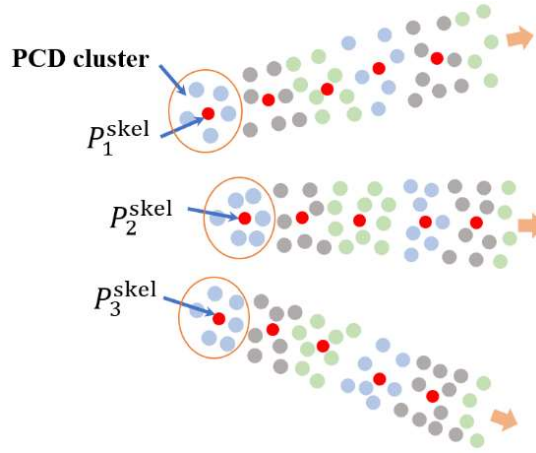


Fig. 7. Central shrinkage of PCD clusters.

The point density in the cluster members was similar to the last processing section. Thus, the centroid was obtained by averaging the coordinates of the points in a cluster, as shown in Eq. (14). The shrinking window was operated sequentially through the PCD clusters to recognize the skeleton point P^{skel} for each PCD cluster. The obtained skeleton points were aligned to produce the extracted skeleton point model $\mathbf{S}_{ca}$. The ridge and diagonal strut skeleton points are indicated using orange and blue colors, respectively. Sparse points were observed on the cables, which are discussed in Section 5.

$$\begin{cases} x^{skel} = \frac{1}{M^{clu}} \sum_{i=1}^{M^{clu}} x_i \\ y^{skel} = \frac{1}{M^{clu}} \sum_{i=1}^{M^{clu}} y_i, i \in [1, M^{clu}], \\ z^{skel} = \frac{1}{M^{clu}} \sum_{i=1}^{M^{clu}} z_i \end{cases} \quad (14)$$

where $(x^{skel}, y^{skel}, z^{skel})$ are the coordinates of point P^{skel} , (x_i, y_i, z_i) are the coordinates of a point in a PCD cluster; and M^{clu} is the number of points in the PCD cluster.

Compared with the original PCD, the data volume after the skeleton point recognition method was effectively reduced to 0.37% of the original volume. Most importantly, the key geometric features of all members were preserved. Further development was conducted to automatically recognize members from skeleton models.

5. Smart Member Recognition

The skeleton model obtained in the previous section maintains integrity. Therefore, automatic member recognition must be conducted to efficiently obtain the geometric information of structural members for SDT construction. Recognition is performed in two steps: the formation of a standard skeleton model and automatic member recognition, described in Sections 5.1 and 5.2, respectively.

5.1. Formation of a standard skeleton model

A standard skeleton model was required as a reference for member recognition. A skeletal model was developed using the design structure. The design structure was first segmented into the corresponding modules. The standard node information and node adjacency matrix (INAM) of a design module were identified, as shown in Fig. 8(a) and 8(b), respectively.

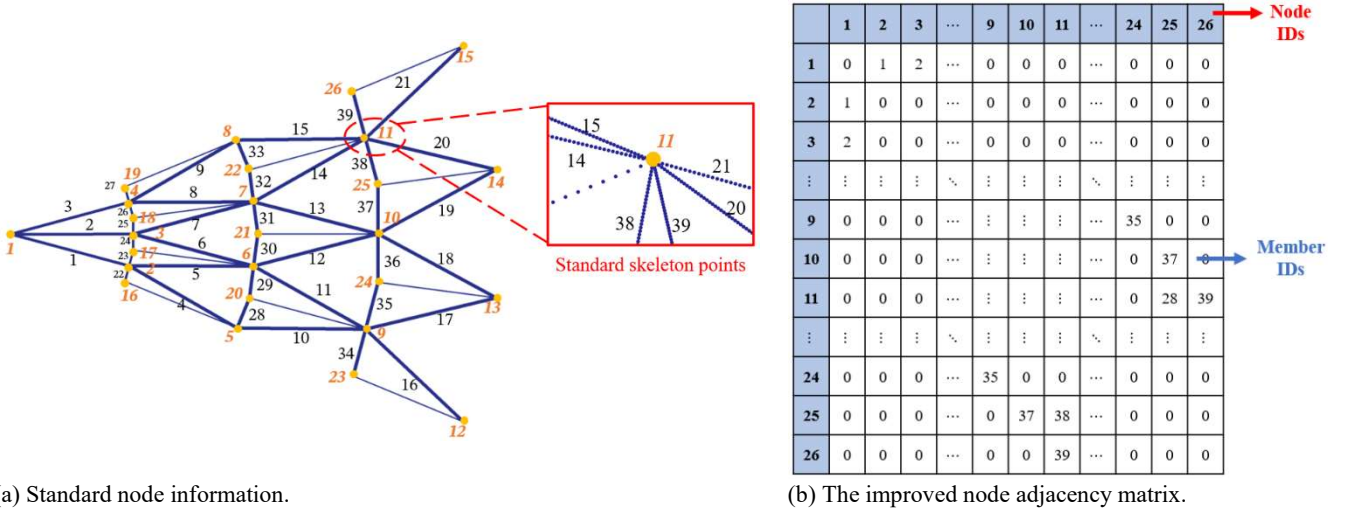


Fig. 8. Standard Skeleton Model.

The standard node information includes both the node ID number and corresponding coordinates of the design module. In contrast, the node adjacency matrix indicates the connectivity between the nodes of the design modules. The pivot rows and columns represent node ID. The other entries in the matrix represent the member ID that connect the corresponding nodes.

INAM is a symmetric and hollow matrix, that is all diagonal the elements are zero, indicating that a node does not connect to itself.

A standard skeleton model $\mathbf{S}_{st}$ was formed based on the INAM design. The members and connections were assumed to be perfect without defects. Interpolation was conducted among the nodes to form the skeleton points of the members. The density of $\mathbf{S}_{st}$ was 200 points/member, as estimated from the measurements $\mathbf{S}_{ca}$. The $\mathbf{S}_{st}$ data contained both the member IDs and spatial coordinates, which were used as the reference for member recognition in the following section.

5.2. Automatic member recognition from skeleton points

An automatic member recognition algorithm was developed using a point-to-point iterative closest point (ICP) method. The target skeleton model $\mathbf{S}_{ca}$ was registered in comparison with the reference standard model $\mathbf{S}_{st}$. The members in $\mathbf{S}_{st}$ provided a spatial position to the target such that they were automatically recognized.

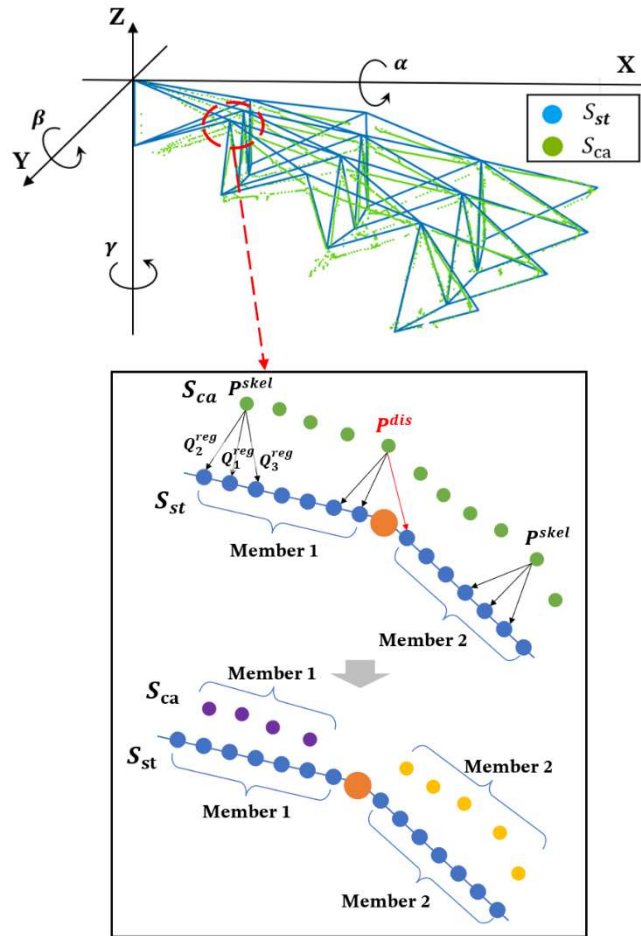


Fig. 9. Registration of skeleton models $\mathbf{S}_{st}$ and $\mathbf{S}_{ca}$.

The $\mathbf{S}_{st}$ and $\mathbf{S}_{ca}$ were first sorted using KD-Tree algorithms for efficient computation. A K-nearest neighbor method was implemented, where comparison pairs of points between $\mathbf{S}_{ca}$ and $\mathbf{S}_{st}$ were formed (p^{skel}, q^{skel}) . The K-nearest neighbor method uses the Euclidean distance as the target function in Eq. (15) to optimize the data registration of $\mathbf{S}_{ca}$, that is, the rotation matrix $\mathbf{R}^{REG}$ and translation vector $\vec{t}^{REG}$. The $\mathbf{R}^{REG}$ is a 3×3 matrix, with three angles α, β, γ rotating around the X, Y, Z axes as in Eq. (16). The $\vec{t}^{REG}$ is a 3×1 vector with three

displacements t_x, t_y, t_z along the X, Y, Z axes from the origin, as shown in Eq. (17). The optimization was repeated 50 times. The error of the registration from ICP algorithm was 0.2 mm, which required a very short time. Finally, points Q^{skel} in S_{st} were adjusted to the coordinate system of S_{ca} by rotation and translation, according to Eq. (18).

$$E(R^{REG}, \vec{t}^{REG}) = \operatorname{argmin} \|S_{ca} - S_{st}\|_2 = \operatorname{argmin} \sum_{i=1}^{M^{skel}} [P^{skel} - (R^{REG} \cdot Q^{skel} + \vec{t}^{REG})]^2 \quad (15)$$

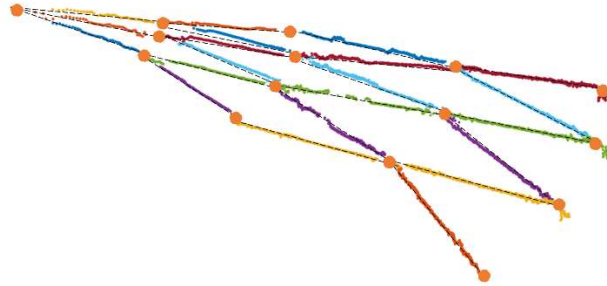
$$R^{REG} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (16)$$

$$\vec{t}^{REG} = [t_x \quad t_y \quad t_z] \quad (17)$$

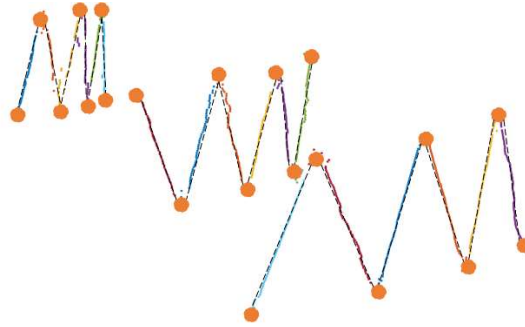
$$Q^{reg} = R^{REG} \cdot Q^{skel} + \vec{t}^{REG} \quad (18)$$

where M^{skel} represents the number of skeleton points in S_{ca} . Q^{reg} represents the adjusted point in S_{st} .

Recognition starts after the completion of registration, as shown in Fig. 9. The members between the two nodes in S_{st} were determined in advance. As shown in Fig. 9, a skeleton point P^{skel} in S_{ca} was first paired with the nearest point Q_1^{reg} in S_{ca} (P^{skel}, Q_1^{reg}). Two additional points, Q_2^{reg} and Q_3^{reg} adjacent to point Q_1^{reg} in S_{st} were then extracted from the set. If the two points are marked with the same member IDs, the skeleton point P^{skel} is considered a member point with the corresponding member ID. However, if one of the points (Q_2^{reg} and Q_3^{reg}) finds a position in another member, the skeleton point P^{skel} is considered to be the point near connection P^{dis} , which is deleted from the misleading information.



(a) Ridge strut skeleton points.



(b) Diagonal strut skeleton points.

Fig. 10. Skeleton points recognition result.

Line fitting was conducted on the recognized members with skeleton points in $\mathbf{S}_{ca}$. The interaction between the two fitting lines was considered to be the node of the module structure, as shown in Fig. 10. Interpolation was conducted to obtain the missing points between the members and nodes. Based on the skeleton points of the members, the next step was to extract the structural geometric features and automatically reconstruct the SDT.

6. Reconstruction of SDT

Member recognition provides the skeleton points of members, which can be further analyzed to determine the position and orientation of members. The nodes connecting the members were generated from a projection-reprojection member. The generation of the structural module and the entire system is realized sequentially.

6.1. Determination of Position and Orientation

The skeleton points of the recognized members were scattered in a structured space, and their positions and orientations were need to be determined. The centroids of the skeleton points were first calculated by averaging the coordinates of the points from Eq. (19). The centroids were set as the positions of the members used in the member generation, described in Section 6.2.

$$\begin{cases} x^{mem} = \frac{1}{M^{mem}} \sum_{i=1}^{M^{mem}} x_i \\ y^{mem} = \frac{1}{M^{mem}} \sum_{i=1}^{M^{mem}} y_i, i \in [1, M^{mem}] \\ z^{mem} = \frac{1}{M^{mem}} \sum_{i=1}^{M^{mem}} z_i \end{cases} \quad (19)$$

where $(x^{mem}, y^{mem}, z^{mem})$ and (x_i, y_i, z_i) are the coordinates of the centroid and skeleton points of member, respectively; M^{mem} is the number of skeleton points in the member; and $\mathbf{S}^{mem}$ represents the set of skeleton points of a member.

As mentioned in the previous section, the distribution of the skeleton points in a set $\mathbf{S}^{mem}$ was linearly aligned. Linear regression was conducted to determine the line segments of set $\mathbf{S}^{mem}$ where the orientation from p_1^{skel} to $p_{M^{mem}}^{skel}$ was determined using the singular value decomposition (SVD) method. The set $\mathbf{S}^{mem}$ was first normalized to improve the fitting accuracy[41].

$$\begin{cases} \sigma^x = \sqrt{\frac{\sum_1^{M^{mem}} (x_i - x^{mem})^2}{M^{mem}}} \\ \sigma^y = \sqrt{\frac{\sum_1^{M^{mem}} (y_i - y^{mem})^2}{M^{mem}}} \\ \sigma^z = \sqrt{\frac{\sum_1^{M^{mem}} (z_i - z^{mem})^2}{M^{mem}}} \end{cases} \quad (20)$$

$$\mathbf{S}_{nor}^{mem}: \begin{cases} x_i^{nor} = (x_i - x^{mem})/\sigma^x \\ y_i^{nor} = (y_i - y^{mem})/\sigma^y, i \in [1, M^{mem}] \\ z_i^{nor} = (z_i - z^{mem})/\sigma^z \end{cases} \quad (21)$$

where $(x_i^{nor}, y_i^{nor}, z_i^{nor})$ are the coordinates of a point in $\mathbf{S}_{nor}^{mem}$; and (x_i, y_i, z_i) are the coordinates of a point in the $\mathbf{S}^{mem}$.

A normalized member skeleton set $\mathbf{S}_{nor}^{mem}$ was obtained and further decomposed using Eq. (20) and (21). The set $\mathbf{S}_{nor}^{mem}$ is an $M^{mem} \times 3$ matrix, where the rows and columns represent the coordinates of the skeleton points and their corresponding serial numbers, respectively. The set $\mathbf{S}_{nor}^{mem}$ can be decomposed into three matrices (Eq. (22)), i.e., the left singular matrix $\mathbf{U}$, singular values matrix Σ , and right singular matrix $\mathbf{V}$. The orientation of the $\mathbf{S}^{mem}$ can be obtained through matrix $\mathbf{V}$, which is a no distorted rotational matrix.

$\mathbf{S}_{nor}^{memT} \cdot \mathbf{S}_{nor}^{mem}$ provide a symmetric property that can be expanded as in Eq. (23). The eigenvector $\vec{v}_1$ corresponding to the largest eigenvalue λ_1 represents the major directions of skeleton points. It was solved thereafter. Based on the orthogonality of the different eigenvectors, both sides of Eq. (23) were simultaneously multiplied by $\vec{v}_1$ to obtain Eq. (24). The right-hand side of the Eq. was shifted to the left-hand side to obtain Eq. (25). Since the vector $\vec{v}_1$ is a non-zero vector, the λ_1 was obtained by solving the determinant for solving for $\vec{v}_1$ in Eq. (26).

$$\mathbf{S}_{nor}^{mem} M^{mem \times 3} = \mathbf{U}_{M^{mem} \times M^{mem}} \cdot \Sigma_{M^{mem} \times 3} \cdot \mathbf{V}_{3 \times 3}^T \quad (22)$$

$$\mathbf{S}_{nor}^{memT} \cdot \mathbf{S}_{nor}^{mem} = \mathbf{V} \cdot \Sigma^2 \cdot \mathbf{V}^T = \begin{bmatrix} \vec{v}_1 \\ \vec{v}_2 \\ \vec{v}_3 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \vec{v}_3 \end{bmatrix} \quad (23)$$

$$(\mathbf{S}_{nor}^{memT} \cdot \mathbf{S}_{nor}^{mem}) \cdot \vec{v}_1 = \begin{bmatrix} \vec{v}_1 \\ \vec{v}_2 \\ \vec{v}_3 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \lambda_1 \cdot \vec{v}_1 \quad (24)$$

$$(\mathbf{S}_{nor}^{memT} \cdot \mathbf{S}_{nor}^{mem} - \lambda_1 \cdot \mathbf{I}) \vec{v}_1 = \vec{0} \quad (25)$$

$$|\mathbf{S}_{nor}^{memT} \cdot \mathbf{S}_{nor}^{mem} - \lambda_1 \cdot \mathbf{I}| = 0 \quad (26)$$

where $(x_i^{nor}, y_i^{nor}, z_i^{nor})$ and (x_i, y_i, z_i) are the coordinates of a point in $\mathbf{S}_{nor}^{mem}$ and $\mathbf{S}^{mem}$.

6.2. Determination of nodes

The 3D geometries of the struts connected by joints have different dimensions. Although the simplified line segments demonstrate the orientations and positions of the struts, the member nodes cannot be directly determined from the lines. In general, these lines do not spatially intersect. However, an intersection P^{NID} was observed when these spatial lines were projected onto the X-Y plane. A projection-reprojection method was used to determine node P^{INT} . The x and y coordinates of the nodes were determined from the intersection of the projected lines. The z-coordinates of the nodes can be determined using the reprojection method.

The nodes are categorized into three types based on the connection relationship of the lines: a two-member node (TMN) refers to a node that connects two members; a multi-member node (MMN) refers to a node that connects

multiple members; and a single-member node (SMN) refers to a node that connects multiple members, as depicted in Fig. 11.

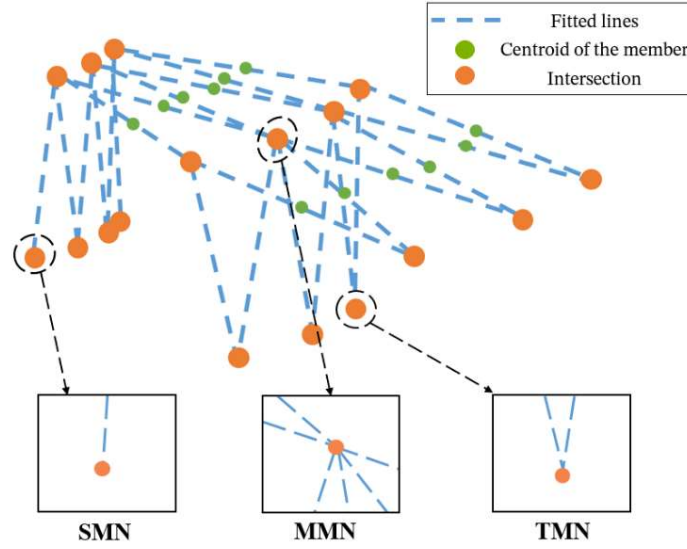


Fig. 11. Classification of nodes.

Two-member nodes (TMN):

First, lines a and b connected by a node were denoted as l_a and l_b , respectively (Eq. (27)). The coordinates of P^{NID} were obtained by solving corresponding Δ^{NID} . First, the two lines were projected onto the $Z=0$ plane and downscaled to two-dimensional straight lines, as shown in Fig. 15(a). The intersection of two-dimensional lines was solved to obtain Δ^{NID} using Eq. (28).

$$l_k^{mem} = \begin{bmatrix} x_k^{mem} \\ y_k^{mem} \\ z_k^{mem} \end{bmatrix} + \begin{bmatrix} v_{xk}^{mem} \\ v_{yk}^{mem} \\ v_{zk}^{mem} \end{bmatrix} \cdot \Delta_k, k = a \text{ or } b \quad (27)$$

$$\Delta^{NID} = \frac{x_a^{mem} - x_b^{mem}}{v_{xb}^{mem} - v_{xa}^{mem}} \quad (28)$$

where the l_k^{mem} represents lines l_a^{mem} and l_b^{mem} . The Δ_k and Δ^{NID} represent the parameter of the two lines and node, respectively. The NID represents the node ID. $[x_k^{mem}, y_k^{mem}, z_k^{mem}]$ and $[v_{xk}^{mem}, v_{yk}^{mem}, v_{zk}^{mem}]$ are the central points and direction vectors of the lines l_a^{mem} and l_b^{mem} , respectively, as mentioned in the previous section.

The coordinates of P^{NID} , x_{ab}^{NID} , y_{ab}^{NID} , and z_{ab}^{NID} were solved after obtaining Δ^{NID} . x_{ab}^{NID} and y_{ab}^{NID} were solved using Eq. (29) and (30), respectively. The z_{ab}^{NID} was reprojected to realize Eq. (31). The dimensional differences between the two members were assumed to be negligible. Therefore, z_{ab}^{NID} was obtained by averaging z values corresponding to the x_{ab}^{NID} and y_{ab}^{NID} at the two spatial lines.

$$x_{ab}^{NID} = x_a^{mem} + v_{xa}^{mem} \cdot \Delta^{NID} \quad (29)$$

$$y_{ab}^{NID} = y_a^{mem} + v_{ya}^{mem} \cdot \Delta^{NID} \quad (30)$$

$$z_{ab}^{NID} = \frac{1}{2} \sum z_i^{NID}, i = a \text{ or } b \quad (31)$$

$$\text{where } z_a^{NID} = z_a^{mem} + v_{za}^{mem} \cdot \Delta^{NID}, z_b^{NID} = z_b^{mem} + v_{zb}^{mem} \cdot \Delta^{NID}$$

Multi-member nodes (MMN):

For the MMN, node coordinate fitting was solved using a similar method. The members connected by the node were grouped two by two, with a total of $M^{NID} = M^{con} \cdot (M^{con} + 1)/2$ groups, where M^{con} represents number of members. The coordinates of $P_1^{NID} - P_M^{NID}$ of the groups were obtained by solving them according to the projection-reflection method. The final coordinates were the centers of the $P_1^{NID} - P_M^{NID}$ obtained using Eq. (32).

$$\begin{aligned} x_M^{NID} &= \frac{1}{M^{NID}} \sum_{i=1}^{M^{NID}} x_i^{NID} \\ y_M^{NID} &= \frac{1}{M^{NID}} \sum_{i=1}^{M^{NID}} y_i^{NID}, i \in [1, M^{NID}] \\ z_M^{NID} &= \frac{1}{M^{NID}} \sum_{i=1}^{M^{NID}} z_i^{NID} \end{aligned} \quad (32)$$

where $(x_M^{NID}, y_M^{NID}, z_M^{NID})$ and $(x_i^{NID}, y_i^{NID}, z_i^{NID})$ are the coordinates of the final intersection and intersection obtained in a group, respectively. The M represents the member IDs intersected by the node.

Single-member nodes (SMN):

An SMN-type node is generally located at the edge of the structural module. The position of the node adjacent to the SMN type node was determined as described above. In addition, the position of the member related to the SMN-type node was known. The distance of the adjacent node to the member position point was assumed to be the same as that of the SMN-type node to the member position point. Therefore, the coordinates of the SMN-type node were obtained by extrapolation using Eq. (33).

$$\begin{bmatrix} x_S^{NID} \\ y_S^{NID} \\ z_S^{NID} \end{bmatrix} = \begin{bmatrix} x_s^{mem} \\ y_s^{mem} \\ z_s^{mem} \end{bmatrix} + \begin{bmatrix} v_{xs}^{mem} \\ v_{ys}^{mem} \\ v_{zs}^{mem} \end{bmatrix} \cdot \Delta^{NID}, \Delta^{NID} = \frac{x_s^{mem} - x_a^{NID}}{v_{xs}^{mem}} \quad (33)$$

where $x_S^{NID}, y_S^{NID}, z_S^{NID}$ are the coordinates of P^{NID} . $[x_s^{mem}, y_s^{mem}, z_s^{mem}]$ and $[v_{xs}^{mem}, v_{ys}^{mem}, v_{zs}^{mem}]$ are the central points and direction vectors of the line intersecting the node, respectively. S represents the line ID. x_a^{NID} represents the x coordinate of the node at the other end of the line.

The nodes were located where the skeletons of the struts and cable can be sketched as linear lines connecting the nodes. The ID numbers of the nodes and skeleton members strictly follow the INAM order. SDT is described in the following section.

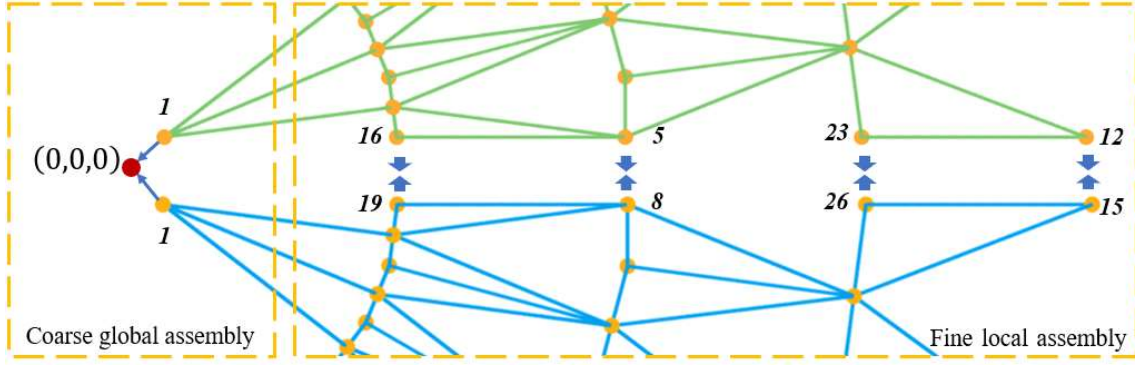


Fig. 12. Coarse global assembly and fine local assembly.

6.3. Assembly of module SDT

The reconstruction of SDT was commenced by sorting the data structures of the modules. The nodes and skeletons of the members were identified using their spatial positions and connectivity. The member dimensions were obtained using the standard model S_{st} as listed in Table 2.

First, the SDT module was inherently determined. The global SDT was formed by the sequential registration of modules. Node 1 in all the modules was used as the initial registration point, as shown in Fig. 12. However, the other edge nodes may not coincide with each other owing to manual module segmentation. Therefore, the common nodes between adjacent modules were estimated by averaging the coordinates of the pairs of edge nodes. Table 2 The data structure of the module SDT.

Member ID	Nodes ID ^a	Node 1 (m)			Node 2 (m)			Class ^b	Cross-section ^c (mm)	Length (mm)
		x	y	z	x	y	z			
1	1/2	0	0	0	1.1880	0.1583	-0.1826	L	$\Phi 30 \times 3$	1212.3
2	1/3	0	0	0	1.1435	0.4822	-0.1954	L	$\Phi 30 \times 3$	1256.3
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
11	6/9	2.3580	0.6161	-0.4089	3.6593	0.4399	-0.6574	L	$\Phi 40 \times 3$	1336.5
12	6/10	2.3580	0.6161	-0.4089	3.4254	1.3631	-0.6439	L	$\Phi 40 \times 3$	1323.8
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
35	9/24	3.6593	0.4399	-0.6574	3.5571	0.9164	-1.4542	L	$\Phi 40 \times 3$	934.0
36	10/24	3.4254	1.3631	-0.6439	3.5571	0.9164	-1.4542	L	$\Phi 40 \times 3$	934.5
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
60	24/25	3.5571	0.9164	-1.4542	3.0587	1.7491	-1.4277	C	$\Phi 12$	970.4
61	25/26	3.0587	1.7491	-1.4277	2.4837	2.5106	-1.4473	C	$\Phi 12$	954.2

a: Node ID represents The Node ID at the two ends of the members: node a/node b.

b: The member form consists of straight lines (L) and curves (C) dictating the respective member-modeling methods.

c: The cross-section of struts are traffic circles, with the dimension of Φ radius of the outer circle $\times$ tube thickness. The cross sections of the cables are circles with dimensions of the radius of the circle.

The reconstructed SDT is shown in Fig. 13. The ridge struts, diagonal struts, diagonal cables, and ring-loop cables are colored green, blue, orange, and yellow, respectively. The SDT demonstrated the actual geometric features of the cable dome structure. The structured data of SDT could be easily used for the further development of numerical simulations or Building Information Modeling (BIM) design.

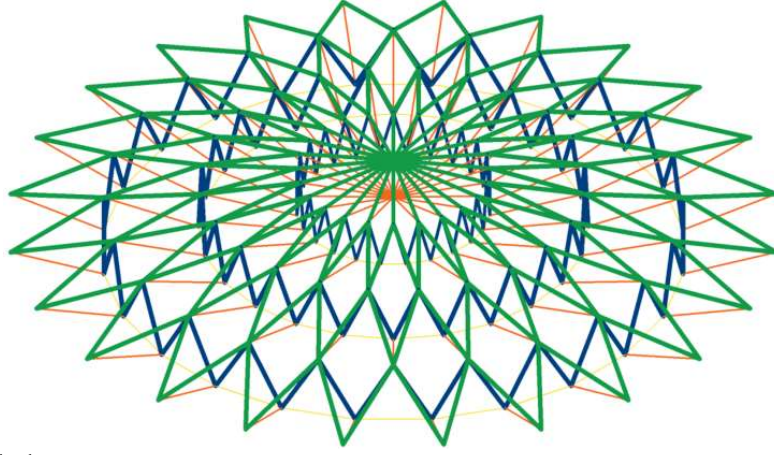


Fig. 13. The SDT of the cable dome structure.

7. Discussion

In this section, the validation of SDT modeling method, application on deviation monitoring, and comparison with other algorithms involved in the SDT modeling method is discussed.

7.1. Validation of SDT modeling method

The validation of the SDT modeling method included precision and efficiency validations, which are technical concerns in structural monitoring with laser scanning.

Precision validation:

In the precision validation of the SDT modeling method, the cable dome SDT was compared with a reference model from Geomagic Wrap. The reference model was prepared with manually marked nodes, where the coordinates of the nodes were known. The target model was a reconstructed SDT cable dome. The precision can be determined from the errors of the pairs of nodes in comparison using Eq. (34).

$$\delta^{NID} = \|P_{ref}^{NID} - P^{NID}\|_2 = \sqrt{(x_{ref}^{NID} - x^{NID})^2 + (y_{ref}^{NID} - y^{NID})^2 + (z_{ref}^{NID} - z^{NID})^2} \quad (34)$$

where P_{ref}^{NID} and P^{NID} are the nodes manually calibrated in the software and fitted from the SDT modeling method, respectively. δ^{NID} represents the fitting error of P^{NID} . $(x_{ref}^{NID}, y_{ref}^{NID}, z_{ref}^{NID})$ and $(x^{NID}, y^{NID}, z^{NID})$ are the coordinates of P_{ref}^{NID} and P^{NID} , respectively.

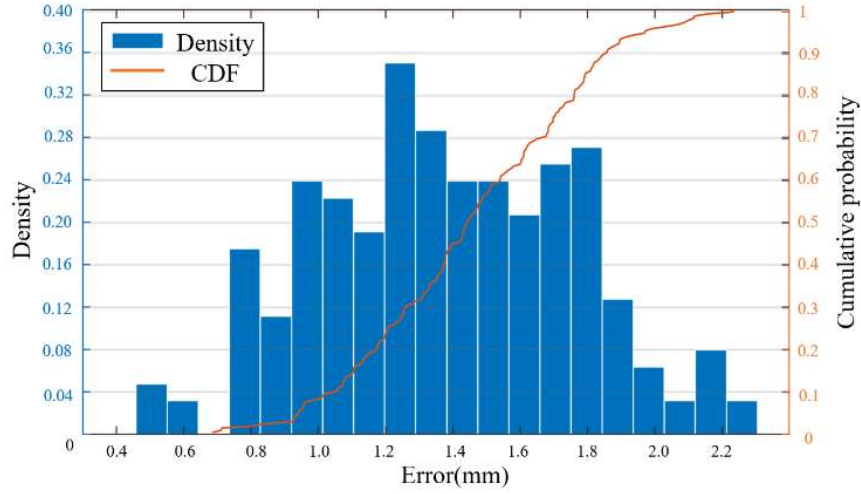


Fig. 14. Node fitting errors.

The comparison results were statistically analyzed by plotting a histogram of the errors between the SDT and reference, as shown in Fig. 14. The cumulative distribution function (CDF) of the errors was also summarized. The error followed the Gaussian distribution. The maximum and mean errors were 2.305 mm and <2 mm, respectively. Considering the span of the cable dome, the precision of the SDT modeling was controlled to within 0.02%. SDT modeling can maintain high fidelity and major features of a structure under 3D laser scanning.

Efficiency validation:

First, the PCD was manually segmented into eight modules. The SDT modeling of each module was accomplished sequentially with automatic assembly of the entire SDT cable dome. A runtime test was performed on a personal computer equipped with an AMD Ryzen7 5800H @3.20GHz processor to test the efficiency of the SDT modeling. The time consumed by each step in the modeling process was recorded in MATLAB software using the tic-toc algorithm, as listed in Table 3.

TABLE 3

SDT modeling time.

Step	Time	
Module segmentation	160.45 s	
Module SDT modeling	Module	Total
Skeleton points recognition	2.87 s	18.75 s
Member recognition	0.41 s	2.43 s
Module data structure sorting	4.63 s	37.97 s
Assembly of module SDT	2.35 s	
Total	221.95 s	
BIM reconstruction method	28931.72 s	
3DPM ^b	2331.6 s	
SDT modeling method v.s. BIM reconstruction method (%)	0.76%	
SDT modeling method v.s. 3D pipeline modeling method (%)	9.52%	

Manual module segmentation required maximum time of 160.45 s. SDT modeling of a single module required a total of 7.91 s. Skeleton points recognition required 2.87 s/module. Member recognition required 0.41 s/module. Module data structure sorting required 4.63 s/module. The overall modeling time was approximately 4 min.

SDT modeling efficiently improved the processing time of volumetric PCD obtained using 3D laser scanning. Two laser-point-cloud SDT modeling methods were compared by considering the equivalent data volume and type of structures: 1) the BIM reconstruction method (BRM), which is applied to full-scale complex tubular engineering structures, with a data volume of ~22 million; the reconstruction required 28931.72 s[34]. 2) the 3D pipeline modeling method (3DPM), which is applied to an as-built pipeline structure, with a data volume of ~135 thousand; the reconstruction required 2331.6 s[43]. The data volume of the SDT cable dome exceeded 12 million. The overall modeling time was 221.95 s. Thus, it required only 0.76% and 9.52% of the processing time of the BRM and 3DPM methods, respectively. The SDT modeling method facilitates real-time deviation monitoring with feasible structural data.

7.1. Application in deviation monitoring

Construction deviation monitoring is currently concerned with the most important structural projects. SDT modeling provides a better access to deviation monitoring than a structural design model. The SDT cable dome was aligned with the design model with the centroids of the two structural models coinciding. The deviations can be determined from the distances between the node pairs using Eq. (35).

$$D^{NID} = \|P_{design}^{NID} - P^{NID}\|_2 = \sqrt{(x_{design}^{NID} - x^{NID})^2 + (y_{design}^{NID} - y^{NID})^2 + (z_{design}^{NID} - z^{NID})^2} \quad (35)$$

where P_{design}^{NID} and P^{NID} are the nodes from the design model and those fitted from the SDT modeling method, respectively. D^{NID} represents the deviation of P^{NID} . $(x_{design}^{NID}, y_{design}^{NID}, z_{design}^{NID})$ and $(x^{NID}, y^{NID}, z^{NID})$ are the coordinates of P_{des}^{NID} and P^{NID} , respectively.

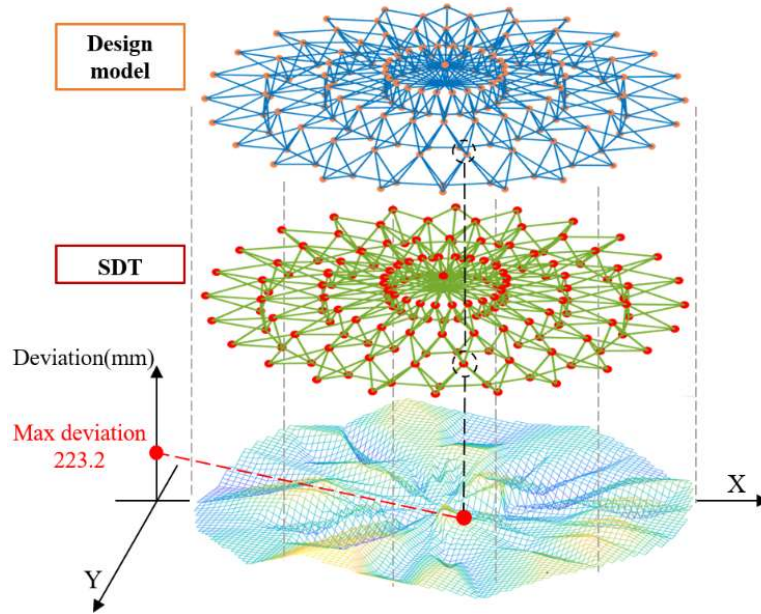


Fig. 15. Nodes deviation.

The comparison results are shown in Fig. 15. An upper cable dome was used as the design model. The middle part was the SDT cable dome. Fig. 15 shows the deviations between the SDT and design model in a 3D illustration. All deviations can be instantly identified. The maximum and average deviations of the nodes were 223.2 and ~ 70 mm, respectively. This deviation was attributed to the deviation of the actual construction from the ideal state. Owing to the stiffness of the ridge struts, lifting the ridge struts to the position of the design state through tensioning is difficult.

7.2. Comparisons

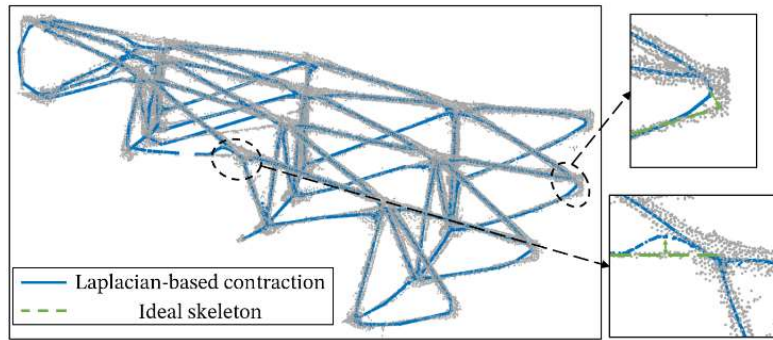
SDT modeling comprises four steps: module segmentation, skeleton point recognition, smart member recognition, and SDT reconstruction. Developing algorithms of skeleton point and member recognitions and each step function is the most important and challenging in the process of SDT modeling. The benefits of the two algorithms were demonstrated by comparing them popular existing methods.

Comparison of skeleton point recognition methods:

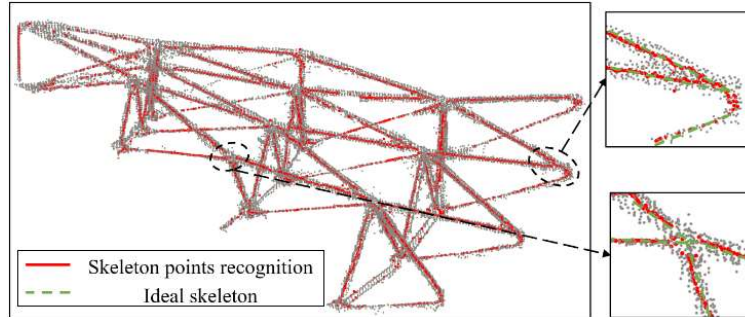
In this study, a novel central shrinkage algorithm was developed for skeleton-point recognition. This algorithm processes denoised and layered PCD in which the centroids of the clusters were identified sequentially. The centroids of the clusters were considered as skeleton points, as described in Section 4.3. The central shrinkage algorithm was compared with a popular skeleton point recognition method, i.e., Laplacian-based contraction method[32][42]. The Laplacian matrix comprises the neighborhood of the target point according to the distance relationship. The eigenvalues of the Laplacian matrix are obtained from eigen decomposition, where the small and large eigenvalues are separated into two groups. Points with small eigenvalues are considered contraction points and discarded.

Conversely, points with large eigenvalues are considered attraction points that are maintained. The procedure is iterated until minimal errors are achieved in the optimization function.

Both methods were applied to the PCD of a module structure for better comparison. Fig. 16 illustrates the results. Fig. 16 (a) shows the results of the Laplacian-based contraction method. The green and blues lines represent the reference and Laplacian-based contraction skeletons, respectively. This shows that the Laplacian-based contraction method performs poorly in corner regions. This discrepancy arose from the varying cross-sectional dimensions of the members. However, the Laplacian-based contraction method conducts uniform contractions in which distortion occurs around abrupt variations. However, the proposed skeleton point recognition method functioned well as the algorithm implicitly adapted to geometric variations (Fig. 16(b)).



(a) Skeleton points from Laplacian-based contraction method.



(b) Result of skeleton points recognition method.

Fig. 16. Comparison of skeleton recognition method.

TABLE 4

Comparison of skeleton extraction methods.

	Time(s)	Shrinkage accuracy			
		Noise point	Percentage	Offset point	Percentage
Method in this study	20.05	65	1.74%	118	3.16%
Laplace shrinkage	90.33	332	1.55%	2514	11.74%

The efficiencies and accuracies of the two methods were compared as summarized in Table 4. The Laplacian-based contraction methods required longer time than the skeleton points recognition method, that is, 90.34 s against 2.87 s per structural module. The numbers of skeleton points disturbed by noise were similar: 1.74% and 1.55 for the Laplacian-based contraction and skeleton point recognition methods, respectively. However, a large skeleton

distortion was observed in the corner regions, accounting for 11.74% of the PCD of the structural module. Conversely, in the skeleton point recognition method, only 3.16% distorted PCD points were observed.

Comparison of member recognition methods:

The novel smart member recognition (SMR) method proposed in this paper automatically recognizes members composed of skeleton points according to a standard model. The details are explained in Section 5.2. The smart member recognition algorithm was compared with the region-growing method, which is popular in member recognition area[27]. The region-growing (RG) method applies a test to the curvatures and angles of a random point and its neighbors in the structural module. The thresholds were set for the curvatures and angles. Points within the thresholds were assumed to belong to a member. The procedure was iterated until all points were processed. However, the region growing method may not function well in cable domes, as shown in Fig. 17. The fixed threshold failed to adapt to the intersections of multiple members.

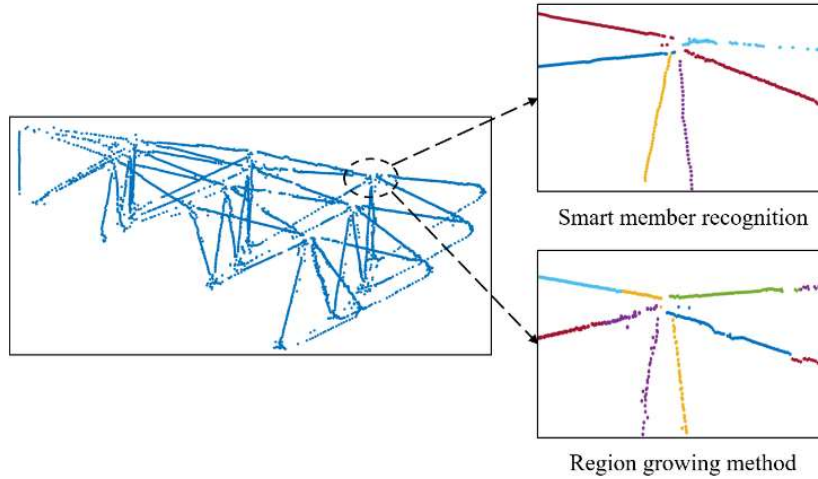


Fig. 17. Comparison of skeleton recognition method.

The efficiencies and accuracies of the two methods were compared summarized in Table 5. In the region-growing method, most members were not fully recognized. The correctness of the recognition was low, and only four out of 39 members were recognized perfectly. However, the smart member recognition method yielded satisfactory recognition results. The false rate was 7.69%. The time consumptions of the two methods were also compared. This indicates that the smart member recognition method was more efficient than the region-growing methods.

TABLE 5

Comparison of member recognition methods.

	Time	Recognition error	
		Wrong Member	Percentage
SMR	0.41s	3	7.69%
RG	2.57s	35	89.74%

8. Conclusion

A high-precision SDT of the cable-dome structure was reconstructed using the proposed SDT modeling method based on 3D laser scanning. The SDT was used for deviation monitoring to improve the efficiency and accuracy of structural detection for the reliability and safety of cable dome structures. The following main conclusions were drawn for the proposed SDT modeling method:

1. New structural skeleton point recognition and member recognition methods were proposed. These methods effectively reduced the time required for feature recognition in large-scale PCD with high accuracy.
2. According to accuracy and runtime test results, the proposed SDT modeling method showed high accuracy, with a fitting error of ~ 2 mm, and high efficiency, requiring $<10\%$ of the time compared with other methods.
3. The SDT was compared to a structural design model. The deviations of all the nodes were extracted synchronously. This illustrated the practicability of the method for structural deviation detection and monitoring.
4. To address the computational efficiency challenges posed by the structural dimensions and PCD data volume, this study employed a dimensionless computational method with various parameters selected based on the structural component dimensions and PCD volume. This flexibility makes it suitable for applications in large-cable dome structures.

This study proposed a technique to reconstruct SDT from large volume of point clouds. The application of reconstructed SDT was aimed at accomplishing automatic deviation monitoring. The reconstructed SDT in future can be further applied in structural performance predictions where numerical simulations will be involved with the high-fidelity SDT models.

Acknowledgments

This study was fully supported by the Key Program of the National Natural Science Foundation of China (52130809).

References

- [1] A. L. Zhang, G. H. Shangguan, Y. X. Zhang, M. Zou, J. Wang, Experimental study on static performance of fully assembled ridge-tube threading cable with annular-struts cable dome, *Eng. Struct.* 288 (2023) 116194, <https://doi.org/10.1016/j.engstruct.2023.116194>
- [2] A. L. Zhang, G. H. Shangguan, Y. X. Zhang, M. Zou, C. Y. Luo, Structural behavior of a fully assembled cable-piercing ridge-tube cable dome with modular assembly ring truss, *J CONSTR STEEL RES.* 211 (2023) 108193, <https://doi.org/10.1016/j.jcsr.2023.108193>
- [3] A. L. Zhang, X. C. Liu, J. Li, J. Q. Ge, G. J. Zhang, Static experimental study on large-span cable dome structure, *J. Build. Eng.* 33 (04) (2012) 54–59.

- [4] L. M. Zhang, W. J. Chen, S. L. Dong, Manufacture error and its effects on the initial pre-stress of the Geiger cable domes, *Int. J. Space Struct.* 21(3) (2006) 141–147, <https://doi.org/10.1260/026635106779380467>
- [5] L. M. Chen, H. Deng, Y. H. Cui, S. L. Dong, Y. Y. Zhou, Theoretical analysis and experimental study on sensitivity of element-length error in cable-strut tensile structures, *Adv. Struct. Eng.* 19(9) (2016) 1463–1471. <https://doi.org/10.1177/1369433216643245>
- [6] H. S. Park, H. M. Lee, A New Approach for Health Monitoring of Structures: Terrestrial Laser Scanning, *Comput-Aided. Civ. Inf.* 22 (2007) 19–30. <https://doi.org/10.1111/j.1467-8667.2006.00466.x>
- [7] X. S. Wu, S. D. Xu, J. Ma, F. Miao, Measuring full static displacements of cable domes based only on limited tested locations, *Appl. Math. Model.* 77 (2020) 1054–1064. <https://doi.org/10.1016/j.apm.2019.08.018>
- [8] W. M. Kritzing, M. Karner, G. Traar, J. Henjes, W. Sih, Digital Twin in manufacturing: A categorical literature review and classification, *IFAC-PapersOnLine*. 51(11) (2018), 1016–1022, <https://doi.org/10.1016/j.ifacol.2018.08.474>
- [9] M. Shafto, M. Conroy, R. Doyle, E. Gleassgen, C. Kemp, J. LeMoigne, L. Wang, Draft modelling, simulation, information technology & processing roadmap, National Aeronautics and Space Administration (2010).
- [10] B. R. Barricelli, E. Casiraghi, D. Fogli, (2019). A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications. *IEEE Access*. 7 (2019) 167653–167671, <https://doi.org/10.1109/ACCESS.2019.2953499>
- [11] L. X. Wang, H. B. Liu, Z. H. Chen, F. Zhang, L. L. Guo, Combined digital twin and hierarchical deep learning approach for intelligent damage identification in cable dome structure, *Eng. Struct.* 274 (2023) 115172, <https://doi.org/10.1016/j.engstruct.2022.115172>
- [12] Y. M. Ye, Q. Yang, J. G. Zhang, S. H. Meng, J. Wang, A dynamic data driven reliability prognosis method for structural digital twin and experimental validation, *Reliab. Eng. Syst. Safe.* 240 (2023) 109543, <https://doi.org/10.1016/j.res.2023.109543>
- [13] J. H. Xu, X. Shu, P. Qiao, S. S. Li, J. G. Xu, Developing a digital twin model for monitoring building structural health by combining a building information model and a real-scene 3D model, *Measurement*. 217 (2023) 112955, <https://doi.org/10.1016/j.measurement.2023.112955>
- [14] V. Vanessa, J. Handibry, F. Henry, Technologies for digital twin applications in construction, *Automat. Constr.* 152 (2023) 104931, <https://doi.org/10.1016/j.autcon.2023.104931>
- [15] Q. Wang, M. K. Kim, Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018, *Adv. Struct. Eng.* 39 (2019) 306–319, <https://doi.org/10.1016/j.aei.2019.02.007>
- [16] D. B. Zhang, T. Q. Liu, S. H. Li, H. Q. Luo, G. Z. Ding, Z. H. Su, C. F. Xie, S. J. Zhang, P. Feng, Three-dimensional laser scanning for large-scale as-built surveying of 2022 Beijing Winter Olympic Speed Skating Stadium: A case study, *J. Build. Eng.* 59 (2022) 105075, <https://doi.org/10.1016/j.job.2022.105075>
- [17] X. C. Wei, J. S. Fan, Y. F. Liu, J. X. Zhang, X. G. Liu, S. Y. Kong, Automated inspection and monitoring of member deformation in grid structures, *Comput-Aided. Civ. Inf.* 37(10) (2021) 1277–1297, <https://doi.org/10.1111/mice.12766>
- [18] Y. Turkan, F. T. Bosché, C. Haas, R. Haas, Tracking of secondary and temporary objects in structural concrete work, *Constr. Innov-Engl.* 14(2) (2014) 145–167, <https://doi.org/10.1108/CI-12-2012-0063>
- [19] M. Kim, D. Lee, T. Kim, S. Oh, H. Cho, Automated extraction of geometric primitives with solid lines from unstructured point clouds for creating digital buildings models, *Automat. Constr.* 145 (2023) 104642, <https://doi.org/10.1016/j.autcon.2022.104642>
- [20] F. Bosche, C. T. Haas, Automated retrieval of 3D CAD model objects in construction range images, *Automat. Constr.* 17(4) (2008) 499–512, <https://doi.org/10.1016/j.autcon.2007.09.001>
- [21] Bentley, URL: <https://www.bentley.com/software/bentley-descartes/>, accessed on 8 January 2024.
- [22] Trimble, URL: <https://cn.geospatial.trimble.com/products-and-solutions/trimble-realworks>, accessed on 8 January 2024.
- [23] Leica, URL: <https://www.leica-geosystems.com.cn/>, accessed on 8 January 2024.
- [24] Z. L. Ma, S. L. Liu, A review of 3D reconstruction techniques in civil engineering and their applications, *Adv. Eng. Inform.* 37 (2018) 163–174, <https://doi.org/10.1016/j.aei.2018.05.005>

- [25] I. Heinz, F. Härtl, C. Fröhlich, Semi-automatic 3D CAD model generation of as-built conditions of real environments using a visual laser radar, 2001 IEEE International Workshop on Robot and Human Interactive Communication, 2001, 400–406. <https://doi.org/10.1109/ROMAN.2001.981937>
- [26] D. Borrmann, J. Elseberg, K. Lingemann, A. Nüchter,. The 3D Hough Transform for plane detection in point clouds: A review and a new accumulator design, 3D Research. 02 (2011) 02003, [https://doi.org/10.1007/3DRes.02\(2011\)3](https://doi.org/10.1007/3DRes.02(2011)3)
- [27] T. Rabbani, F. Van Den Heuvel, G. Vosselman, Segmentation of point clouds using smoothness constraint, Int. Archives Photogrammetry Remote Sensing Spatial Inform. Sci. 36 (2006). 248–253.
- [28] D. S. Li, J. P. Liu, L. Feng, Y. Zhou, H. T. Qi, Y. F. Chen,. Automatic modeling of prefabricated components with laser-scanned data for virtual trial assembly, Comput-Aided. Civ. Inf. 36(4) (2020) 453–471, <https://doi.org/10.1111/mice.12627>
- [29] R. Schnabel, R. Wahl, R. Klein,. Efficient RANSAC for Point-Cloud Shape Detection, Comput. Graph. Forum. 26(2) (2007) 214–226. <https://doi.org/10.1111/j.1467-8659.2007.01016.x>
- [30] Y. H. Jin, W. H. Lee, Fast Cylinder Shape Matching Using Random Sample Consensus in Large Scale Point Cloud, Appl. Sci. 9(5) (2019) 974, <https://doi.org/10.3390/app9050974>
- [31] A. Tagliasacchi, H. Zhang, D. Cohen-Or, Curve skeleton extraction from incomplete point cloud, ACM Trans. Graph. 28(3) (2009) 1–9, <https://doi.org/10.1145/1576246.1531377>
- [32] H. Huang, S. H. Wu, D. Cohen-Or, M. L. Gong, H. Zhang, G. Q. Li, B. Q. Chen, L1-medial skeleton of point cloud, ACM Trans. Graph. 32(4) (2013) 1–8, <https://doi.org/10.1145/2461912.2461913>
- [33] J. Yang, X. R. Wen, Q. L. Wang, J. S. Ye, Y. L. Zhang, Y. Sun, A Novel Algorithm Based on Geometric Characteristics for Tree Branch Skeleton Extraction from LiDAR Point Cloud, Forests, 13(10) (2022) 1534, <https://doi.org/10.3390/f13101534>
- [34] J. Liu, L. Fu, G. Cheng, Automated BIM Reconstruction of Full-Scale Complex Tubular Engineering Structures Using Terrestrial Laser Scanning, Remote. Sens.-Basel. 14(7) (2022) 1659, <https://doi.org/10.3390/rs14071659>
- [35] G. Castellazzi, C. Carloni, F. Ubertini, An innovative numerical modeling strategy for the structural analysis of historical monumental buildings, Eng. Struct. 132 (2017) 229–248, <https://doi.org/10.1016/j.engstruct.2016.11.032>
- [36] G. Castellazzi, N. Lo Presti, A. M. D’Altri, S. de Miranda, Cloud2FEM: A finite element mesh generator based on point clouds of existing/historical structures, SoftwareX. 18 (2022) 101099, <https://doi.org/10.1016/j.softx.2022.101099>
- [37] L. Yang, J. C. P. Cheng, Q. Wang, Semi-automated generation of parametric BIM for steel structures based on terrestrial laser scanning data, Automat. Constr. 112 (2020) 103037, <https://doi.org/10.1016/j.autcon.2019.103037>
- [38] Y. Zhao, J. M. Guo, Z. Y. Jiang, W. G. Chen, G. G. Zhou, Control method for determining feasible pre-stresses of cable-stayed structure, Thin. Wall. Struct. 174 (2022) 109159, <https://doi.org/10.1016/j.tws.2022.109159>
- [39] E. Lucchi, Digital twins for the automation of the heritage construction sector, Automat. Constr. 156 (2023) 105073, <https://doi.org/10.1016/j.autcon.2023.105073>
- [40] H. Ma, A. L. Zhang, X. Zhao, Y. X. Zhang, G. Du, Research on digital twin technology for large-span steel structures based on 3d laser scanning, 2023 13th Pacific Structural Steel Conference, 102.
- [41] L. A. Shalabi, Z. Shaaban, B. Kasasbeh, Data Mining: A Preprocessing Engine, J. Comput. Sci. 2(9) (2006) 735–739.
- [42] O. K. Au, C. L. Tai, H. K. Chu, D. Cohen-Or, T. Y. Lee, Skeleton extraction by mesh contraction, ACM Trans. Graph. 27(3) (2008) 1–10, <https://doi.org/10.1145/1360612.1360643>
- [43] J. Lee, H. Son, C. Kim, C. Kim. Skeleton-based 3D reconstruction of as-built pipelines from laser-scan data, Automat. Constr. 35 (2013) 199–207, <https://doi.org/10.1016/j.autcon.2013.05.009>

文件名: Newly-developed structural digital twin modeling
method for monitoring deviations of prefabricated cable domes based on
3D laser scanning

目录: C:\Users\马浩\Documents

模板: C:\Users\马浩
\AppData\Roaming\Microsoft\Templates\Normal.dotm

标题:

主题:

作者: 浩 马

关键词:

备注:

创建日期: 2024/1/22 15:54:00

修订号: 6

上次保存日期: 2024/1/22 16:02:00

上次保存者: 浩 马

编辑时间总计: 2 分钟

上次打印时间: 2024/1/22 16:03:00

打印最终结果

页数: 27

字数: 9,980 (约)

字符数: 56,887 (约)