As-is Modelling of Deformed Linear Objects from 3D Point Clouds using Inverse Kinematics -based Iterative Closest Point

Jumana Dayeh[2] and Frédéric Bosché[1]

[1]School of Engineering, University of Edinburgh, UK

jumanadayeh2@gmail.com f.bosche@ed.ac.uk,

Abstract -

Accurate as-built BIM models are essential for construction monitoring, Quality Control (QC), and Facilities Management (FM). However, current Scan-to-BIM workflows struggle in reconstructing accurate as-built models. This paper presents the inverse kinematics -based iterative closest point (ICPIK) method for automated parametric modelling of deformed elements without manual intervention. Experimental validation across four case studies demonstrates that the algorithm achieves precise alignment (RMSE < 0.006m) for deflections up to 22.5°, handles multiple cross-sections and deformation locations, and maintains accuracy with up to 25% missing data.

Keywords -

Scan-vs-BIM; As-is Modelling; FM; BIM; Scan-to-BIM

1 Background

Many studies have shown that the presence of an accurate digital as-built model enables project managers to identify deviations and adjust schedules as needed during construction without having to rely on human intervention [1]. An as-built model also unlocks potential for use in Facilities Management (FM) and structural health monitoring, reducing costs and increasing efficiency [1, 2].Yet, to this date, the Architecture, Engineering, and Construction (AEC) industry has had to rely on manual labour to track the as-built conditions of buildings, lacking the digital tools necessary for accurate data collection and documentation [3]. The nature of the inspections carried out is susceptible to human judgement and error. As a result, there is a significant lack of reliable and accurate as-built information in construction projects [4].

The advent of reality capture technologies has significantly improved the state of data capture. The most popular tool used for collecting point cloud data (PCD) of an existing building or infrastructure asset is laser scanning, although alternative sensing technologies, such as structure-from-motion photogrammetry and 3D range cameras (e.g., the Microsoft Kinect), have also gained popularity [2].

Aside from data capture, *Scan-to-BIM* and *Scan-vs-BIM* software have seen advancements for detecting (as-designed) 3D BIM model elements in 3D point clouds, quantifying deviations, and creating BIM models from 3D point clouds [5].

Scan-to-BIM is the process of generating a Building Information Model (BIM) from point cloud data, typically obtained via laser scanning. It is currently the most widely used method to create as-built models. However, it relies on object recognition algorithms and necessitates manual interpretation [6]. Most *Scan-to-BIM* workflows rely on fitting idealised geometric primitives, which means that deformed components, such as a bent column or pipe, may be inaccurately reconstructed as perfect geometries. As a result, manual intervention is frequently required to interpret data correctly and capture deviations from the original design, a process that is far from fully automated according to the review by Son et al. [6]. At the core of *Scan-vs-BIM* processes is the registration of the captured data (typically point clouds) in the coordinate system of the project BIM model. This enables *Scan-vs-BIM* algorithms to automatically search for component matches in the vicinity of their designed locations and, if matches are found, to calculate deviations and make further inferences.

Scan-vs-BIM is the process used to match as-built data with as-designed models by using object detection and recognition. It is primarily used to compare a 3D as-built point cloud (typically generated from laser scanning) with a design BIM model to identify deviations [7]. fig. 1 summarizes a data processing workflow by Bosché et al. [1] for information management that integrates *Scan-vs-BIM* and *Scan-to-BIM*. But, most research on this topic stops at identifying the recognition, identification, and deviation of as-built conditions from as-planned conditions, lacking capability to generate the final as-built BIM model.

Among the available registration methods (which include traditional robust project survey point-based methods), optimisation-based methods are widely adopted, as rigorous mathematical theories can guarantee their convergence, they require no training data, and generalise well to unknown scenes [8]. Optimisation-based registration methods can be characterized into global, coarse, and fine registration. Coarse registration provides rough, approximate alignment as a computationally efficient initialization step. Fine registration refines an already roughly aligned

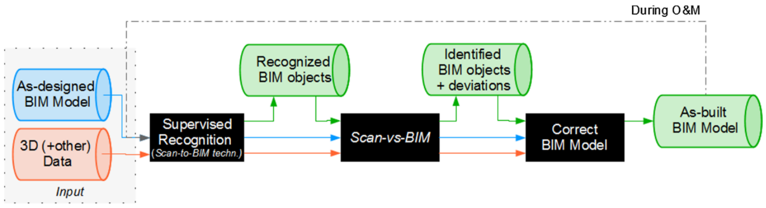


Figure 1. Data processing system for life-cycle BIM model dimensional information management (reproduced with permission from [1]

dataset to achieve precise alignment, but can fail if the initial alignment is poor. Global registration aims to achieve precision registration without initialisation, and is generally achieved by using coarse registration first, followed by fine registration for optimal results [8].

3D registration methods can also be categorized into rigid, non-rigid, and semi-rigid types [9]. Each industry adopts the registration approach that best aligns with the characteristics of the objects being registered. For example, in medical imaging, non-rigid registration is often used because as it allows for deformation, which is essential for accurately modelling objects that undergo continuous shape changes, such as soft tissues [10]. On the other hand, the AEC industry typically relies on rigid registration, which aims to align point clouds through translation and rotation only, without altering the object's shape [8]. Semi-rigid registration offers a balance between rigid and non-rigid registration and is also known by various aliases, such as 'articulated registration'. It is mainly used in the field of robotics and motion tracking [11][9].

In the AEC, although rigid registration methods are effective for aligning scenes for scan-vs-BIM and to subsequently find the as-built pose of each objects, they may be inadequate for finding the as-built pose of some objects. This is because some objects can have deformations, which rigid methods cannot capture. Semi-rigid and non-rigid registration methods could potentially be implemented to fill this gap, but have yet to be explored in this context.

Many objects in construction scenes are linear elements, all of which can be subject to deformations (principally along their main axis). Therefore, accurately modelling the as-built pose of linear objects subject to such deformation is important and could be achieved using semi-rigid fine registration methods, in particular.

1.1 Contribution

This work aims to enhance the practice of as-built BIM modelling by implementing a semi-rigid fine registration algorithm within a Scan-vs-BIM framework, to automatically produce accurate parametric models of linear objects subject to geometric deformations. The study focuses only on the deformation of the extrusion path of a linear object; surface deformations are outside the scope.

Being able to produce as-built models that more correctly capture such deformations would directly support the following uses:

- Enhancing structural health monitoring, helping engineers assess safety and serviceability with improved precision.
- Improving dimensional quality control during construction by enabling the detection and quantification of deviations in shape, not just position, thus facilitating early corrective actions and reducing rework.
- Facilitating digital documentation of historical or ageing infrastructure, where deformation is often present, supporting preservation, retrofit, and restoration efforts through accurate digital replicas.

2 3D Fine Registration Methods

2.1 Iterative Closest point Algorithm (ICP)

The Iterative Closest Point Algorithm (ICP) is the most popular rigid fine registration algorithm, introduced by Besl and McKay [12]. It is also widely used in articulated registration and non-rigid workflows [9].

Point-to-point ICP revolutionised 3D shape registration at its time by iteratively establishing point correspondences and calculating rigid transformations to minimise mean squared errors. While efficient and independent of shape representation, it must be noted that ICP's reliance on accurate initial alignment limits its robustness, as poor estimates can lead to local optima or failure to converge [12, 9]. *Point-to-plane* ICP is a variant of the ICP algorithm that aligns point clouds by minimising the distance between points in one cloud and the planes defined by the normals of points in the other cloud [13]. To address issues in vanilla point-to-point and point-to-plane ICP, He et al. [14] introduced the Geometric Feature-based ICP (GFICP), which incorporates geometric features such as curvature and surface normals to enhance accuracy and speed. However, GF-ICP is sensitive to feature extraction errors in noisy environments, and surface normals computed must be accurate to prevent incorrect matches.

2.2 Non-Rigid Fine Registration

Non-rigid registration methods allow for more flexibility by enabling the deformation of point clouds to achieve alignment. They often employ sophisticated algorithms, such as Coherent Point Drift (CPD) [15] and Gaussian Mixture Models (GMM) [9], to handle the complexities of deformable shapes. Ge [16] proposed a method leveraging geodesic distances and the 4-Point Congruent Sets (4PCS) algorithm integrated with RANSAC, enabling robust alignment of deformed objects while preserving intrinsic surface properties. Similarly, Amberg et al. [17] extended ICP with locally affine transformations, effectively aligning flexible, non-rigid objects like human faces by ensuring smooth transitions across regions. However, these methods can be computationally intensive and cannot deal with large volume points, the case in AEC applications[10]. Also, they rely on distinctive points that must be matched in two sets of data. In the case of column, no such point really exists.

2.3 Semi-Rigid Fine Registration

Semi-rigid or 'articulated' registration methods represent an intermediary approach that deals with articulated chains or bodies of rigid segments. Lin et al. [11] presents one of the few semi-rigid registration methods in the AEC industry for aligning 3D point clouds when rigid objects in the target scene have undergone positional or rotational changes from the source scene. The method starts by breaking down point clouds into rigid segments. It then determines segment correspondences between two point clouds using energy minimisation solved with integer programming, which establishes an initial rigid transformation for the entire point clouds. Finally, the rigid transformation for each pair of matching segments is calculated. This method effectively works, for aligning entire scenes in a Scan-vs-BIM context where items are misplaced or misaligned. However, it does not attempt to reconstruct as-built poses of individual objects.

A comprehensive survey on rigid to non-rigid registration methods by Tam et al. [9] reveals that, although significant advancements have been made, most of these advancements are in the fields of real-time hand tracking, post-estimation, and medical imaging, without any significant work associated to the AEC sector.

Within the array of articulated registration methods, Fleishman et al. [18] introduces a robust articulated ICP algorithm for matching a kinematic model of an articulated body to a point cloud for real-time hand tracking. The algorithm determines the joint transformation to align an articulated hand model to captured point clouds of different hand poses in real time, using an Inverse Kinematics (IK) approach and leveraging the iterative nature of ICP. Tests were performed using depth sensors and the performance of the algorithm was comparable to state-of-the-art methods, with failures occurring due to erroneous correspondences made or inaccuracies in the hand modelling. Tam et al. [9] predicted such failures, claiming that articulated registration requires more reliable correspondences to be found, which is a challenging task. Nonetheless, the method proposed by Fleishman et al. [18] shows significant potential in applications within the AEC industry.

3 Method

A computational method is implemented for the as-built modelling of linear BIM elements in the AEC industry, based on the ICPIK algorithm introduced by Fleishman et al. [18] for hand tracking. The ICPIK approach was tailored for AEC applications by creating a kinematic modelling algorithm, implementing weighting mechanisms, and specifying constraints for structural elements based on their realistic behaviour. fig. 2 illustrates the ICPIK algorithm implementation workflow.

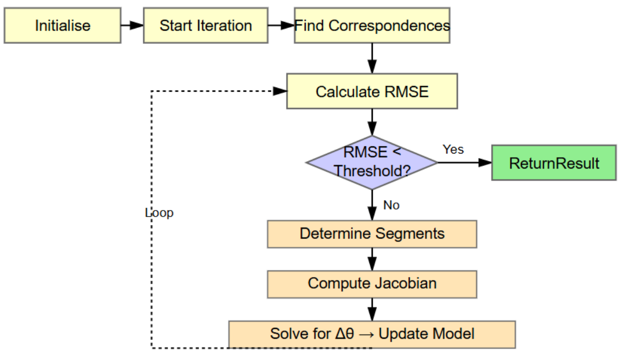


Figure 2. ICPIK Workflow

3.1 Kinematic Modelling of a Linear Object

To apply the ICPIK method, a linear element is modelled as an articulated chain of jointed segments that are then rotated to align with the deformed point cloud. Multiple segments form the object, connected via rotational joints with three degrees of freedom (X, Y, Z). Each segment maintains its world-space position and a local rotation relative to its parent, forming a parent-child hierarchy where movement of one segment affects its children. Accurate computation of the direction vector is essential for updating segment positions and generating 3D geometry. Each segment's local rotation is represented by three angles (x, y, z) in radians, used to construct rotation matrices R_x, R_y,

R_z, which are combined as local orientation $R = R_z R_y R_x$. The world orientation is obtained by multiplying the parent's orientation by the local orientation, and the direction vector is calculated by applying the world orientation to a unit Z-axis vector.

The dependent relationships between segments ensure transformations propagate correctly through the kinematic chain. When a parent rotates, its children maintain local orientations while moving and rotating in world space, preserving connectivity and allowing realistic deformations. This kinematic model forms the basis for the inverse kinematics-based registration algorithm (ICPIK) described in section 3.3.

3.2 Correspondence Finding: Point-to-Point ICP

Correspondence finding is performed using the ICP algorithm. Point-to-Point matching is chosen for its lower computational cost and suitability for simple, symmetric objects. To improve the quality of the matches, a distance threshold limits potential point pairs, reducing incorrect matches. Inverse distance weighting emphasises closer points with non-linear scaling, and segment-based weighting prioritises segments with more correspondences.

The ICPIK algorithm employs two-level convergence. At the inner level, Open3D's ICP terminates when relative changes in fitness score and RMSE both fall below 1e-6. At the outer level, the main optimisation loop terminates when the maximum distance between corresponding points falls below 0.001m, or after 100 iterations. An adaptive distance threshold for correspondence matching is initialized at 0.3m and gradually decreases with iteration count, tightening matching criteria as the model converges toward the target geometry.

3.3 Inverse Kinematics Optimisation

The inverse kinematics optimisation represents the core solution to the problem. Once correspondences between the kinematic model and the simulated point cloud are established, the algorithm employs a damped least squares method to determine optimal joint angles that minimise the distance between the closest corresponding points [18].

The Inverse kinematics problem is stated as eq. (1), finding values for the joint rotations, $\boldsymbol{\theta}$, such that:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \|\mathbf{t} - \mathbf{s}(\boldsymbol{\theta})\|^2 \tag{1}$$

where $\boldsymbol{s}(\boldsymbol{\theta})$ are the positions of source points on the kinematic model as a function of the joint angle, $\boldsymbol{\theta}$, and $\boldsymbol{t}$ are the corresponding target points on the deformed point cloud. Unlike traditional IK formulations where a fixed set of end-effectors (eg:fingertips) are predefined, this method treats all correspondence points as optimisation targets. This approach is more suitable for linear structural elements since they dont have specific control points.

The problem can be solved by computing the Jacobian (eq. (2)) to linearly approximate the function.

$$\boldsymbol{J}(\boldsymbol{\theta}) = \left(\frac{\partial s_i}{\partial \theta_j}\right) \tag{2}$$

The entries of the Jacobian matrix $\boldsymbol{J}(\boldsymbol{\theta})$, are calculated using eq. (3). For each correspondence between a source point s and a target point t, the influence of a segment's rotation is captured using the cross product of the segment's rotation and the difference between the source points being handled at that iteration and their respective joint:

$$\frac{\partial s_i}{\partial \theta_j} = \mathbf{v}_j \times (\mathbf{s_i} - \mathbf{p}_j) \tag{3}$$

where $\mathbf{v}_j$ is the rotation in world coordinates for joint j, and $\mathbf{p}_j$ is position of the joint, j, and $\mathbf{s}_i$ represents the current position of source point i

The Jacobian matrix J and error vector e are assembled from the matched point pairs, and the optimal change in joint angles $\Delta\boldsymbol{\theta}$ is computed by solving eq. (4):

$$(J^\top J + \lambda I + C + Q)\Delta\boldsymbol{\theta} = J^\top \mathbf{e} \tag{4}$$

where $J^\top J$ is:

$$J^T J = \sum_{i=0}^{m} \frac{\partial s_i}{\partial \theta_j} \tag{5}$$

and $J^\top e$ is:

$$J^T \mathbf{e} = \sum_{i=0}^{m} \frac{\partial s_i}{\partial \theta_j} \cdot (t_i - s_i). \tag{6}$$

λI is a damping term that ensures numerical stability and prevents overshooting, C is a directional constraint matrix that favours natural bending along the dominant deformation axis while penalising motion in orthogonal directions, and Q is a continuity matrix that penalises sharp changes in rotation between neighbouring segments.

The algorithm incorporates additional mechanisms to improve performance. Weighted point correspondences assign higher importance to more reliable points, while segment prioritisation gives greater influence in the Jacobian to segments with more active correspondences. Joint angle limits constrain segment rotations to prevent unrealistic deformations, and adaptive damping adjusts the damping factor to ensure stability and convergence.

4 Experimental Validation

This method is implemented in Python using Open3D, PythonOCC, and NumPy. The following sections will outline features designed for experimental validation, the evaluation criteria, experimental set up, and discussion.

4.1 Point Cloud Simulation

To evaluate the method, a function was created for simulating realistic point clouds of the deformed kinematic models. It calculates the number of points to allocate to each segment by dividing the total number of points in the simulated point cloud (a user-defined parameter) by the number of segments in the model (also a user-defined parameter). Reproducible results are ensured by initialising the random number generator with a fixed seed.

Different cross-section handling is implemented for generating point clouds from the kinematic model. For example, in circular cross-sections, the radius is extracted from the segment's size property, and points are distributed on the cylinder surface using a grid approach along theta (circumference) and z (length) coordinates. Small random perturbations (i.e. noise) are added to avoid perfectly regular patterns, and a parameter is included allowing the user to exclude faces in order to simulate occlusions. Figure 3 shows an example of simulated point cloud.

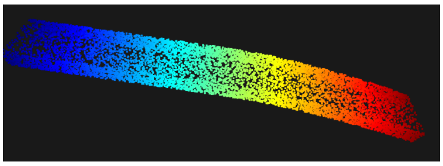


Figure 3. Simulated Complete Point Cloud: Circular cross-section and 10000 total points

4.2 Evaluation Criteria

The effectiveness of registration methods in as-built modelling is typically evaluated based on accuracy, computational efficiency, and robustness to occlusions. For articulated registration method, we recommend evaluating capacity at handling different degrees of deformation,locations of the deformation, and cross-sections.

Accuracy is commonly measured using the Root Mean Square Error (RMSE) between the registered and target point clouds. This quantitative metric is complemented by qualitative visual inspection to reveal alignment issues not fully captured by RMSE alone. For articulated models specifically, the Euclidean distance between corresponding nodes in the kinematic chain can also be computed to assess alignment at discrete points along the linear element, identifying which segments are better aligned.

In this paper, the ICPIK algorithm is evaluated against these criteria using simulated point clouds with controlled deformations, allowing for precise measurement of performance under various conditions.

4.3 Experimental Setup

Four case studies are considered to evaluate the performance of the ICPIK algorithm. They include:

- **Case Study 1**: To evaluate ability to handle different degrees of deformation. Conducted with a 0.1 x 0.3m beam cantilever with a rectangular cross-section.
- **Case Study 2**: To evaluate ability to handle deformations at different locations. Conducted with a circular column with a 0.15m radius, fixed at one end.
- **Case Study 3**: To evaluate ability to handle different cross-sections. Conducted with circular, box, and H-section columns.
- **Case Study 4**: To evaluate ability to handle different levels of PCD completeness. Conducted with the same beam as Case Study 1, with missing PCD from (1) one side parallel to the deformation axis, (2) one side perpendicular to the deformation axis, and (3) jointly two sides perpendicular and parallel to the deformation axis.

Eleven experiments are conducted using the four case studies. table 1 summarises details of each test. All experiments employ kinematic models with linear elements of 2.5m in length, decomposed into 10 segments with each segment having a 0.25 m length and apply deformations beyond realistic deflections to test the limits of the ICPIK.

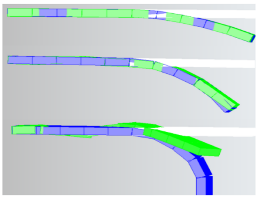
Table 1. Overview of Experimental Tests

Case Study	Test	Description
1	a	9° rotation at seg. 8 (small)
	b	22.5° rotation at seg. 8 (moderate)
	c	45° rotation at seg. 8 (large)
2	a	9° rotation at seg. 8 (top)
	b	9° rotation at seg. 4 (mid)
3	a	Circular (0.15m & 0.2m)
	b	Box (0.35m × 0.35m)
	c	H-section
4	a	Missing side parallel
	b	Missing side perpendicular
	c	Two missing sides (50%)

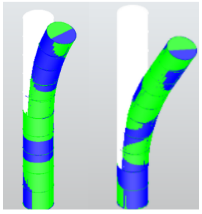
4.4 Results and Discussion

The experimental results are summarised in fig. 4 for visual evaluation, in fig. 5 reporting the RMSE, and in fig. 6 reporting the final error between segment nodes.

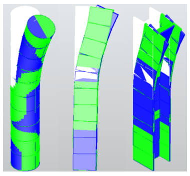
Case Study 1 (fig. 4a) reveals that the algorithm's success is highly dependent on deformation magnitude. Deflections below 22.5°achieved excellent alignment with RMSE values below 0.006m, while deformations at 45° resulted in complete failure. This limitation stems from the point-to-point ICP correspondence finding step, which struggles when source and target points are initially far apart [12]. For structural BIM elements operating within serviceability limits, this threshold is adequate. However,



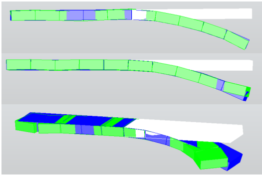
(a) Case Study 1



(b) Case Study 2



(c) Case study 3



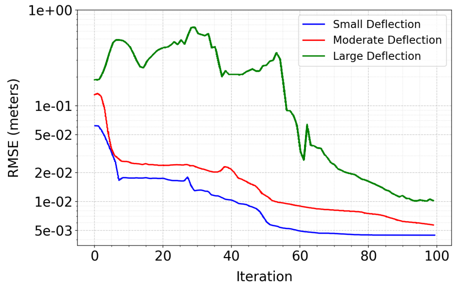
(d) Case Study 4

Figure 4. 3D visual of original (white), ground truth (blue), and ICPIK result (green)

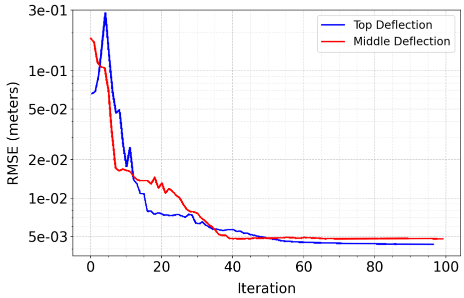
for applications involving non-structural elements (e.g., MEP components) that may experience larger deformations, alternative correspondence methods such as GFICP should be explored.

Case Study 2 (fig. 4b) confirms that deformation location does not significantly impact the algorithm's success for cantilevers, with both tip and midpoint deflections achieving comparable accuracy. The observed difference in RMSE convergence patterns simply reflects better initial conditions rather than fundamental performance differences. However, this experiment is limited to elements with one fixed end, and the lack of tests on doubly-constrained elements represents a gap that should be addressed in future work.

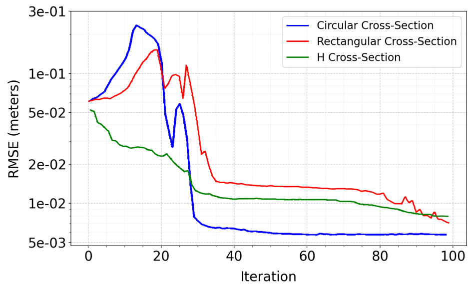
Case Study 3 (fig. 4c) reveals superior performance for circular cross-sections. Circular sections converges 20-40 iterations earlier than rectangular, box, or H-sections and achieved lower final RMSE values (0.004-0.005m vs 0.006-0.009m). This is likely because point-to-point ICP lacks directional information, making it suboptimal for geometries with multiple orthogonal planes. When matching points on planar surfaces, the algorithm cannot leverage surface normal information, increasing the likelihood of erroneous correspondences. Since the weighting mechanism prioritises high-quality correspondences, planar cross-sections received fewer heavily weighted matches, affecting overall accuracy. Future work should explore point-to-plane ICP with surface normal computation, which could significantly improve performance for non-circular sections. In terms of overall performance, point-to-plane converges faster than point-to-point. While it has a slightly higher computational cost per iteration due to calculating surface normals, the fewer total iterations required lead to overall faster registration. Thus, it



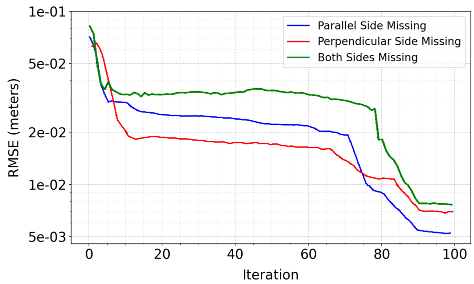
(a) Case Study 1



(b) Case Study 2



(c) Case Study 3

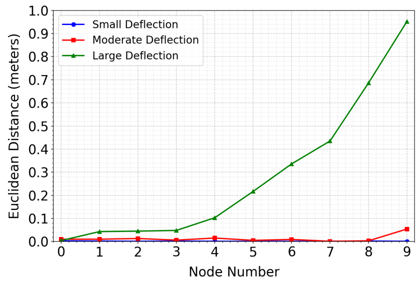


(d) Case Study 4

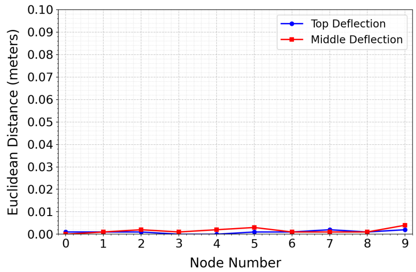
Figure 5. Root Mean square Error (RMSE) at Each Iteration

would increase the overall speed and accuracy for planar sections, but could reduce accuracy due to the sensitivity of the surface normal computation to noise.

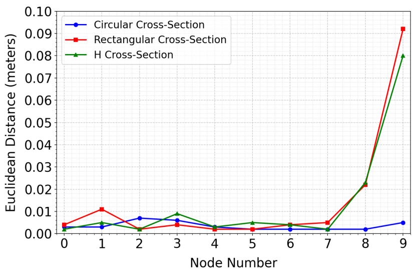
Case Study 4 (fig. 4d) demonstrates that the algorithm maintains reasonable accuracy when up to 25% of the point cloud is missing, with tip deviations remaining below 0.05m. Interestingly, removing the side perpendicular to deformation results in marginally larger errors (0.048m)



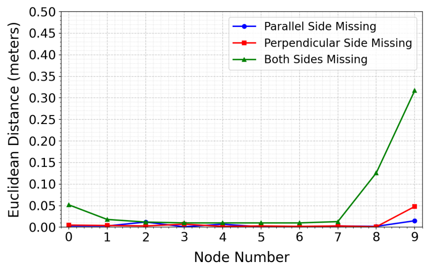
(a) Case Study 1



(b) Case Study 2



(c) Case Study 3



(d) Case Study 4

Figure 6. Euclidean Distance Between Nodes in Ground Truth and ICPIK Result

than removing the parallel side (0.015m), despite the parallel side having greater surface area. This suggests that geometric information perpendicular to the deformation axis plays a more critical role in constraining the solution space. However, when 50% of the point cloud is missing, the algorithm fails completely, bending toward incorrect axes with deviations exceeding 0.3m. This failure highlights two issues: insufficient bending direction constraints in the kinematic model and the importance of adequate correspondence coverage. Preprocessing techniques should be explored to improve robustness in occluded scenarios.

Across all experiments, 82% of tests (9 out of 11) achieve successful alignment despite the fact that the applied deformations are beyond realistic deflections, demonstrating the algorithm's practical viability for as-built modelling within its operational envelope. The convergence patterns observed—rapid initial improvement, brief plateau, then gradual refinement—reflect the algorithm's structure, where adjustments propagate through the kinematic chain. For typical structural monitoring applications where deflections remain well below the 45° threshold, the ICPIK method provides an effective automated solution that eliminates the manual intervention required by traditional Scan-to-BIM workflows.

While the simulations incorporate random perturbations to approximate noise, captured PCDs present considerably more complex characteristics, such as spatially uneven density and mixed noise. These factors could affect performance at two key stages. First, uneven point density would introduce bias into the segment-based weighting mechanism, causing the Jacobian matrix to over-represent well-sampled regions of the element while under-constraining sparsely scanned segments, potentially skewing the computed joint angle updates and degrading alignment accuracy. Second, real noise differs fundamentally from random Gaussian perturbations and could corrupt the correspondence-finding step, producing erroneous point pairs that propagate through the IK optimisation. To address these challenges, future work should evaluate the algorithm on captured PCD, and investigate preprocessing strategies such as statistical outlier removal.

5 Conclusions

This paper presented the ICPIK algorithm for automated as-built modelling of deformed linear structural elements from 3D point clouds. The method integrates kinematic modelling, the ICP algorithm, and Inverse Kinematics-based optimisation to create parametric models that accurately reflect existing geometric deformations.

The algorithm demonstrated robust performance across multiple stress-testing scenarios. For deflections under 45°, ICPIK achieved precise alignment. Alignment was achieved for circular, rectangular and H cross-sections, with better performance achieved for circular cross-sections requiring 40-60 iterations compared to 80+ iterations for the other two. The method successfully handled point clouds with up to 25% missing data when occlusion occurred on a single face, though performance degraded significantly with 50% data loss.

The ICPIK method addresses a critical gap in existing Scan-to-BIM and Scan-vs-BIM workflows by enabling automated parametric modelling of deformed elements without manual intervention. This capability has immediate applications in structural health monitoring, digital twin development, and facility management where accurate as-built documentation is essential. Current limitations include the assumption of rigid segments without material (surface) deformation and reliance on point-to-point ICP for correspondence finding. Future work should (1) incorporate point-to-plane ICP for improved correspondence matching, (2) extend the framework to account for surface deformations, (3) extend validation on real-world captured point clouds. The potential extension to MEP components, particularly deformed piping systems, represents a promising direction given their frequent deformation and current reliance on manual inspection processes.

References

[1] F. Bosché, A. Guillemet, Y. Turkan, C.T. Haas, and R. Haas. Tracking the built status of mep works: Assessing the value of a scan-vs-bim system. Journal of Computing in Civil Engineering, 28(4):05014004, 2014. doi:10.1061/(ASCE)CP.1943-5487.0000343.

[2] J. Xu, K. Chen, F. Xue, and W. Lu. 3D Point Cloud Data Enabled Facility Management: A Critical Review, pages 641–657. 02 2021. doi:10.1007/978-981-15-3977-0_49.

[3] K. Saidi, A. Lytle, and W. Stone. Report of the nist workshop on data exchange standards at the construction job site. 09 2003. doi:10.22260/ISARC2003/0095.

[4] R. Navon and R. Sacks. Assessing research issues in automated project performance control (appc). Automation in Construction, 16(4):474–484, 2007. doi:https://doi.org/10.1016/j.autcon.2006.08.001.

[5] F. Bosché. Automated recognition of 3d cad model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction. Advanced Engineering Informatics, 24(1):107–118, 2010. doi:https://doi.org/10.1016/j.aei.2009.08.006.

[6] H. Son, C. Kim, and Y. Turkan. Scan-to-bim-an overview of the current state of the art and a look ahead. In ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction, volume 32, page 1, 2015.

[7] F. Bosché, M. Ahmed, Y. Turkan, C.T. Haas, and R. Haas. The value of integrating scan-to-bim and scan-vs-bim techniques for construction monitoring using laser scanning and bim: The case of cylindrical mep components. Automation in Construction, 49:201–213, 2015. doi:https://doi.org/10.1016/j.autcon.2014.05.014.

[8] X. Huang, G. Mei, J. Zhang, and R. Abbas. A comprehensive survey on point cloud registration. ArXiv, abs/2103.02690, 2021.

[9] G.K.L. Tam, Z.-Q. Cheng, Y.-K. Lai, F.C. Langbein, Y. Liu, D. Marshall, R.R. Martin, X.-F. Sun, and P.L. Rosin. Registration of 3d point clouds and meshes: A survey from rigid to nonrigid. IEEE Transactions on Visualization and Computer Graphics, 19(7):1199–1217, 2013. doi:10.1109/TVCG.2012.310.

[10] B. Deng, Y. Yao, R.M. Dyke, and J. Zhang. A survey of non-rigid 3d registration. Computer Graphics Forum, 41(2):559–589, 2022. doi:https://doi.org/10.1111/cgf.14502.

[11] B. Lin, K. Sakai, T. Tamaki, B. Raytchev, K. Kaneda, and K. Ichii. Semi-rigid registration of 3d points. Journal of Signal and Information Processing, 04: 25–29, 2013. doi:10.4236/jsip.2013.43B005.

[12] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2):239–256, 1992. doi:10.1109/34.121791.

[13] S.-Y. Park and M. Subbarao. An accurate and fast point-to-plane registration technique. Pattern Recognition Letters, 24:2967–2976, 12 2003. doi:10.1016/S0167-8655(03)00157-0.

[14] Y. He, B. Liang, J. Yang, S. Li, and J. He. An iterative closest points algorithm for registration of 3d laser scanner point clouds with geometric features. Sensors, 17(8), 2017. ISSN 1424-8220. doi:10.3390/s17081862.

[15] P. Wang, P. Wang, Z. Qu, Y. Gao, and Z. Shen. A refined coherent point drift (cpd) algorithm for point set registration. Science China Information Sciences, 54:2639–2646, 2011.

[16] X. Ge. Non-rigid registration of 3d point clouds under isometric deformation. ISPRS Journal of Photogrammetry and Remote Sensing, 121:192–202, 2016. doi:https://doi.org/10.1016/j.isprsjprs.2016.09.009.

[17] B. Amberg, S. Romdhani, and T. Vetter. Optimal step nonrigid icp algorithms for surface registration. In CVPR, 06 2007. doi:10.1109/CVPR.2007.383165.

[18] S. Fleishman, M. Kliger, A. Lerner, and G. Kutliroff. Icpik: Inverse kinematics based articulated-icp. In 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 28–35, 2015. doi:10.1109/CVPRW.2015.7301345.