F.C. Collins, A. Braun, A. Bormann: Finding geometric and topological similarities in building elements for large-scale pose updates in Scan-vs-BIM. Proc. of the Int. Conf. on Computing in Civil and Building Engineering (ICCCBE), 2022

Finding geometric and topological similarities in building elements for large-scale pose updates in Scan-vs-BIM

Fiona C. Collins[1 [0000−0001−5246−7727]], Alexander Braun[1 [0000−0003−1513−5111]], and André Borrmann[1 [0000−0003−2088−7254]]

Technical University of Munich, 80333 München, Germany, fiona.collins@tum.de

Abstract. Information-rich BIM models are rarely usable off-the-shelf for operations tasks. Change decisions made on the construction site can lead to significant differences between the as-designed and as-built state of buildings. The responsibility for keeping the digital representation in sync with its physical twin is not defined and will likely only fully be assigned when automatic methods facilitate the geometric update process. To this end, previous research succeeded in (1) identifying if an element was erected at the time and position it was initially designed, and (2) updating the parametric design geometry to fit its LiDAR-measured as-built state under a set of assumptions and threshold values.
The research presented in this paper aims at updating the as-designed model in case of significant pose differences between the as-designed and as-built state. The method leverages graphs to encode the topological connectivity between geometric elements, once for the as-designed BIM model and once for the as-built point cloud. A similarity metric, namely the cosine distance, allows for a quantitative comparison of the topologically enriched point cloud clusters and their corresponding BIM element. The results show that a convincing type-wise similarity can be found in the feature space between the as-built point cloud clusters and the BIM elements. This similarity score becomes meaningful once the element's topological arrangements are included. An instance-wise similarity score of above 90% is achieved for matching-pairs of free-standing columns and allows for a large-scale pose update in the as-designed BIM model.

Keywords: Scan-vs-BIM · knowledge-driven as-built reconstruction · GNN.

1 Introduction

Geometrically accurate and information-rich BIM models are indispensable in compliance-driven building projects. Whilst the information depth of such models increases with the growing adoption of digital planning methods, the risk of significant discrepancies during construction stays high. The slight geometric deviations accumulating during the on-site construction process can result in geometric conflicts [14] and require fast and reactive mitigation measures at the

time and place of construction [15]. In most cases, the need for a belated change request results in delays, exceeding costs, and material waste for matching compliance initially set out for the project. Although most geometrical discrepancies are expected to stay within a specific tolerance range, it happens that the as-built construction shows building elements with significant positional differences from their as-designed representation in the BIM model. Geometric deviations challenge the delivery of reliable as-built design documents [1] and lead to significant model rework times.

Spatial and visual data acquired by LiDAR scans and cameras offer an accurate yet unstructured representation of the as-built status. The process of using the resulting point clouds for updating the as-designed BIM model is currently supported by point cloud semantic segmentation (PCSS) results. The point cloud shows clusters of points assigned to a set of semantic labels. In current based practice, a manual workflow follows to move the as-designed geometries into place. The match between the as-designed and its matching as-built point representation is the modeller's responsibility. Existing approaches have investigated BIM parametrization and show to automatically update elements for which the deviations lie within a certain threshold. This work aims to present a method for automatically finding geometric and topological similarity-based matches between the point cloud data (PCD) and as-planned BIM data in an encoded vector space. The results show the first step in extending the current research body of automated dynamic BIM model updates to cover large-scale pose deviations. More precisely, the method aims to break the limitation of tolerance (threshold) values. At the method's core lies the geometric and topological encoding with graph structures for either input format into a comparable vector space.

2 Background

2.1 Updating the as-planned

Finding matches between the as-designed BIM models and as-built point clouds has been a topic in the research fields of progress track and Scan-vs-BIM related contributions. The first focuses on detecting an element at a given moment and location in time to verify the overall construction progress. The second explores methods for geometrical updates in the as-designed BIM model. Accurately capturing the on-site construction progress has been vigorously investigated with spatial and visual data acquisition methods such as LiDAR scans and photos. A point cloud is either generated by Time-of-Flight sensors or photogrammetry and offers an accurate 3D geometric representation of the visible as-built construction. A control-point-based acquisition [18, 13] or an Iterative Closest Point (ICP) approach allows for point cloud to BIM registration. To validate the presence of a building element at a given point in time, Tuttas et al. [17] compare a set of metrics between the triangulated planes from 4D BIM elements and the local fitted planes from the point cloud that lie in the distance d from the object. Braun et al. [2] measure the point density within a stage-dependant threshold of a building element. Turkan et al. [16] compare the scanned point

clouds with their orthogonal projection point on the as-planned surfaces. Braun et al. [3] enhance the detection by point surface comparison with an additional artificial intelligence (AI) approach involving a visibility analysis and element detection in images.

Having detected that an element is or is not present at a given time at its design location answers the use case of progress tracking. However, it might occur that the construction is well on time, yet the element is built with deviations from its designed geometric shape or pose. To that end, [13] introduce the concept of a Dyna-BIM (dynamic BIM) where both the element pose and the element shape in BIM are parameterized. Two metaheuristic optimization techniques show the ability of parametric slabs (in the article referred to as footings) to adapt to the as-built point cloud. In their approach, a previous point cloud processing crops the point cloud into segments of just the vicinity of each footing, thereby minimizing the effects of different footings on the optimization of one. For bridges, Mafipour et al. [11] further explore parametric shape fitting into point clouds previously processed with PCSS. All the above-explored automatic contributions are limited when it comes to large-scale deviations or rely on a previous point cloud slicing which induces assumptions about the element match. Bosché et al. [1] apply Circle Hough transform to compare as-planned piping systems to their as-built equivalent. Thanks to the shape prior they introduce, they can cover large-scale pose deviations to a certain degree. The described state-of-the-art is illustrated in Fig. 1 and Fig. 2.

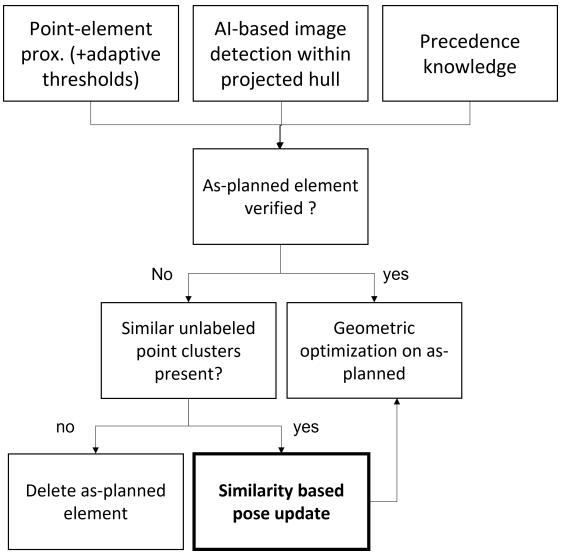


Fig. 1: State-of-the-art workflow for as-built progress track and design model fitting. The contribution of this work is highlighted in bolt.

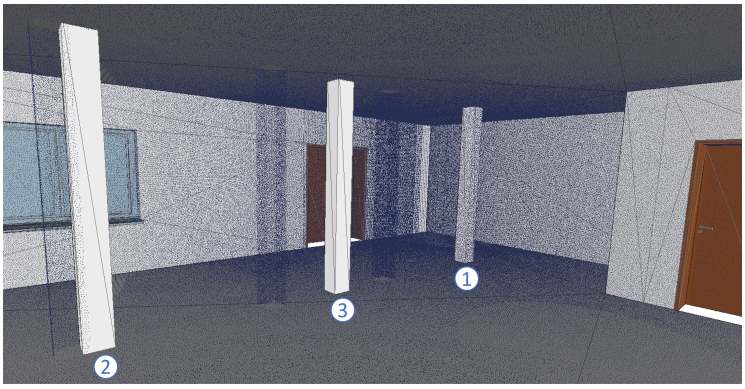


Fig. 2: As-built point cloud overlay with as-designed BIM model: 1) Column validated with point proximity, 2) Column validated with extended methods s.a. adaptive thresholds or with precedence knowledge 3) as-designed column not validated, two point clusters of potential columns remain.

2.2 Spatial relationships and dependencies

Using graphs to capture the relationship and dependencies between different building elements has been helpful as a concept. Characterized by their flexibility to represent complex relationships and heterogenous attributes, graphs have become the favored way of expressing information from BIM enriched with the insights of as-built point cloud acquisitions [22]. The connections (edges) allow for conditional semantic links between the elements (nodes), which host rich feature representations of the latter. Braun et al. [4], for instance, assemble a precedence relationship graph from as-planned BIM models where the technological temporal dependencies of an advancing construction are encoded as directed edges from an object (predecessor) to the depending object (successor). While recording the construction progress, the certainty of identifying a successor is significantly decreased if its corresponding predecessor has not been detected yet. Wang et al. [21] show that spatial encoding in a graph structure helps predict simple room layouts.

On the other hand, spatial encoding with graphs is also an important topic in AI research for performing PCSS. Graphs allow to encode both, local geometry [20] as well as topological element connections [10]. The super point graph approach suggested by [10] utilizes the hypothesis that a large amount of information needed for semantic element segmentation is contained in topological relationships.

2.3 Similarity-based object queries

Retrieving the most similar data object based on the content of the query object has found many applications in other areas, such as content-based image

retrieval (CBIR) in search engines or Graph similarity search for chemical compound retrieval. Retrieving entities informed with global scene information has yielded better results than entity-based retrievals alone. Maheshwari et al. [12], for instance, show that encoding images as scene graphs allow the image match to return more intuitive results – e.g., capturing object interactions such as human-animal interaction.

Some research has focused on retrieving the best matching CAD geometry from a point cloud object in the context of buildings. Bosché et al. [1], for instance, use a similarity-based criterion integrating location, radius, and orientation for retrieving cylindrical MEP components. Wang et al. [19] retrieve the most similar CAD models from a furniture shape database to align them with the input point cloud. A set of rotation-invariant key point features is used. The mentioned approaches have a shortcoming in that contextual information is largely neglected.

2.4 Contribution

Little work exists on adequately updating as-designed BIM models geometrically to their as-built equivalent while preserving initial model semantics [13]. More specifically, no automatic method can capture large-scale pose deviations between the as-planned model and the as-built point cloud. In this work, the described benefits of spatial relationships are used in formulating topological graphs to create more informed matches between the as-designed and as-built representations. The contribution of this article is summarized as follows:

- A method to perform similarity-based matches to provide pose updates for the outdated as-designed model
- The formulation of a graph matching problem for two fundamentally different yet often jointly used data sources (as-designed BIM geometry and as-built point clouds)

3 Method

The proposed approach to find correspondence between as-built and as-designed building elements across large-scale pose deviations is presented in three steps: 1. Graph formulations 2. Topological enrichment with GCNs 2. Similarity computation. The straightforward next step of updating the as-designed element is out of the scope of this work.

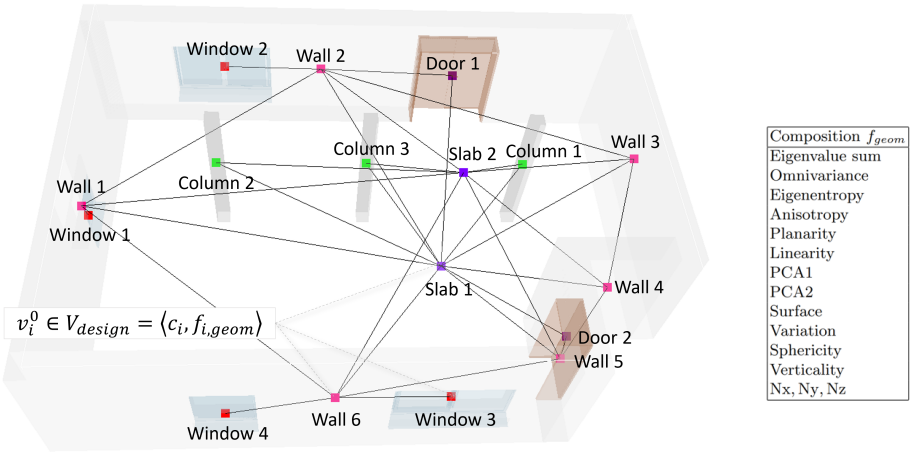
3.1 Graph formulation for as-designed and as built

The two corresponding yet different graphs are defined as G_{design} and G_{built} with the set of vertices V_{design}, V_{built} and edges E_{design}, E_{built} respectively.

***G_{design}*:** The building elements' shape characteristics and topological relationships are extracted from the BIM model represented as IFC[1] model to represent

[1] Industry Foundation Classes – A standardized BIM data exchange format

the as-designed facility in a graph format. The vertices $v_i = \langle c_i, f_{geom} \rangle \in V_{design}$ are defined as the elements geometrical centroids c_i and each include a set of features f_{geom}. The feature vector f_{geom} is a compact vector representation of the element's shape, summarized with a set of computed features as suggested by [7] (see Fig. 3b). The element's neighborhood information is extracted from the IFC files with the spatial query language QL4BIM suggested by [6] and the topological operator "touching". Accordingly, elements adjacent to each other will share an edge in G_{design}. The resulting graph is illustrated in Fig. 3a.

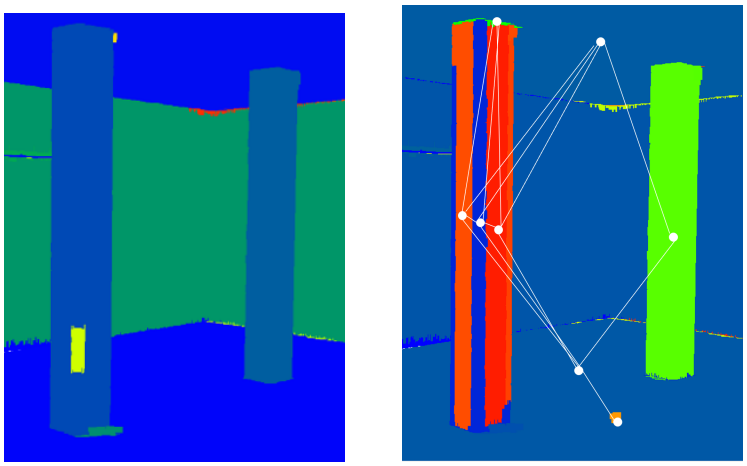


(a) (b)

Fig. 3: Topological connections in graph representation of the as-designed model G_{design}. The nodes include the generated shape representation, and the edges denote "touching" elements. For a better visibility, the top slab is not included in the illustration.

G_{built}: To generate the graph-based as-built representation, we assume a segmented point cloud as an input. The preceding point cloud segmentation (PCS) assigns points to semantically homogeneous clusters. This step can be performed with a conventional geometric clustering algorithm or a supervised (semantic) segmentation using Artificial Neural Networks such as suggested by [9]. An as-built element might be represented by one or several point clusters. Furthermore, depending on the regularisation strength of the clustering method as well as the geometric noise and occlusions in the source point cloud, these clusters might be more or less fine-grained. Figure 4a and b show results of such clustering with different regularization strengths. For our experiments the noise level is

set to 0.005 in the simulation, and the regularization strength in Landrieu and Simonovsky's [9] method to 0.08.



(a) Strong regularization

(b) Regularization r=0.08 and exemplary visualization of G_{built}.

Fig. 4: Point cloud geometric clustering results with the method by Landrieu and Simonovsky [10]

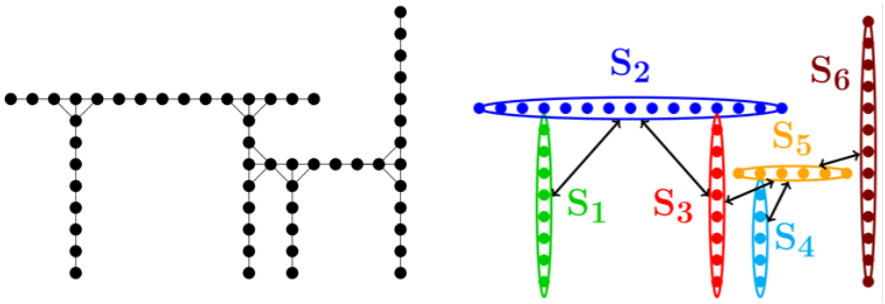
Similarly, as for the as-designed, the clusters' centroids c_i and the same set of features f_{geom} are computed for each cluster, forming the $v_i = \langle c_i, f_{geom} \rangle \in V_{built}$. To achieve the topological connectivity between the clusters, the graph formulation method from [10] is used. As the authors suggest, a symmetric Voronoi adjacency graph $G_{\text{voronoi}} = (P, E_{voronoi})$ is defined for the whole point cloud $P_{as-built}$ first. Two point clusters S_1 and S_2 are adjacent and marked with an edge in G_{built} if there is at least one edge in $E_{voronoi}$ with one end in S_1 and one end in S_2 (see Fig. 5).

The experimental set-up is structured according to two possible PCS results:

- One single point cluster represents one building element (idealistic setting)
- Several point clusters represent one building element (see Fig. 4b, left column).

3.2 Topological enrichment of both graphs with GCNs

In the graph formulation described above, the edges represent topology, yet the nodes' features contain no information about their neighborhood. In buildings, topology (here encoded as graph neighborhoods) is highly relevant for any type of scene interpretation. It is thus essential to include not only the elements shape characteristics but also what connectivity it has to its surroundings.



(a) $E_{voronoi}$ (b) G_{built}

Fig. 5: Graph formulation as proposed by Landrieu and Simonovsky [10]

Furthermore, in this work, we use a similarity score based solely on the node features (see Sec. 3.3 for further detail). It is, therefore, necessary to propagate the node features along the graph edges. Hence a 2-layered Graph Neural Network (GNN) as in [8] is assembled for message passing along the graph edges. The GCN learns to transform the input feature vector of node v_i together with an aggregate of the nodes neighboring messages to produce the predicted node label closest to the true label[2]. For a detailed description of GCNs, the reader is referred to [5]. Here the training is formulated as a fully-supervised node classification problem. Following an inductive training setting, the GCN is optimized on a set of 15 design graphs deduced from 15 different design models. Testing is performed on a different set of 4 design graphs. On average, the graphs contain 1300 nodes representing the classes Wall, Slab, Stairs, Door, Window, Furniture, Column, and Beam. Once the network achieves a balanced accuracy in node classification performance of over 85% on the test set, the network is used to perform inference on the showcase building presented in Section 4. The inference results are once computed for G_{design} and once for G_{built} without retraining, resulting in the node-wise feature vectors $f'_{i,design}$ and $f'_{i,built}$. The final node embedding vectors of the predictions can be regarded as an element representation, including shape characteristics and topology.

3.3 Similarity computation

To find matching building elements between the as-designed model and the as-built point cloud, we compute the similarity between the nodes of the respective graphs. Computing the similarity score between the initial feature vectors $f_{i,design}$ and $f_{i,built}$ hints at how many geometrical similarities the two elements have. Comparing the enriched vectors $f'_{i,design}$ with their corresponding $f'_{i,built}$, topological characteristics will complement the shape representation and thus influence the similarity score. The cosine similarity is computed between pairs

[2] Wall, Slab, Stairs, Door, Window, Furniture, Column, and Beam

of features as shown in Eq. 1. Figure 6 illustrates how nodes between the two graphs are matched for a case when one point cluster represents one building element. The analysis is conducted similarly when more than one point cloud clusters represent one building element.

In this research, we compute the node-wise similarity across all node pairs in the two graphs. Instead, in practice, the similarity computation could be limited to the unverified as-designed elements or the point clusters for which no design element was found in immediate proximity.

$$Sim(A, B) := \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}} \tag{1}$$

where A and B stand for $f^k_{i,design}$ and $f^k_{i,built}$; the two feature vectors to be compared. Note that when k=0, $f^k_{i,design}$ is simply the input feature vector $f_{i,design}$.

In this research, we compute the node-wise similarity across all node pairs in the two graphs, resulting in quadratic complexity of the algorithm. In practice, however, the similarity computation could be limited to the unverified as-designed elements or the point clusters for which no design element was found in immediate proximity.

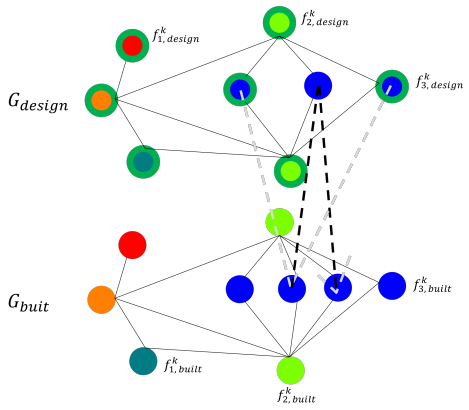


Fig. 6: Multi-fold graph correspondence between the as-designed and the as-built graph (schematic graphs follow the scenario outlined in Fig. 2). Nodes with green outlines are verified design elements according to state-of-the-art methods. For the unverified node, the most similar nodes in the as-built are found and indicated with black dashed lines. The other way around (grey dashed lines), the most similar design element is returned for each point cloud cluster.

4 Experiments

The method is demonstrated for a simplistic showcase building consisting of walls, slabs, windows, doors, and columns. The BIM model used for the image material of this article is the same as used for the experiments (see Fig. 3a) and 2. To demonstrate the effectiveness of our method, we introduce pose deviations to the column elements. The columns all have typical thicknesses of 250x250mm, and their pose deviations were all made in the aligning direction of all columns. For column 2, a discrepancy of 0.2m is set between the as-designed and as-built status. Column 3 is duplicated, and either as-built column equivalent is situated at a 1m distance from its as-planned equivalent. We will refer to the respective as-built column equivalent with a subscripted alphabetic letter after the number.

The graphs are constructed as described in Section 3.1 and 3.1, before the similarity score is computed as described in Section 3.3 with or without topological enrichment (see Section 3.2. In two experiments, we set out to show (1) the importance of topology for the similarity score and (2) the methods' generalization potential to apply to the state-of-the-art clustering methods.

For simplification, a simulated point cloud is used as input instead of a real point cloud, complementing the as-designed BIM model. The LiDAR scan simulation tool of Winiwarter et al. [23] is used. For simulating the as-built capturing process, a conventional Terrestrial Laser Scanner (TLS) is defined, important hardware parameters are set and scan positions are placed.

4.1 Importance of topology

In this experiment, we assume the point-cloud clusters to represent the BIM elements one-to-one. Some sophisticated point cloud clustering methods can achieve this when little noise and obstacles are present in scenes. The clustered as-built point cloud is compared to the as-designed BIM equivalent element-wise. Fig. 7 shows the pair-wise results of the similarity score, once for the computed node features f_i (a) and once for the topologically enriched node features f_i' (b). The heatmaps are not rectangular since the as-built point cloud includes one additional column, as shown in Fig. 2.

Without topological enrichment, the similarity scores for matching and congruent pairs are noticeably high, with similarity scores around 95%. For elements where the majority of element faces are captured by the LiDAR scanner (e.g. for columns 4 large faces are captured) the score lies higher than for such that have unilateral occluded faces (e.g. walls, slabs where the LiDAR scanner captures only one visible face). However, for the elements with deviations, the similarity score decrease with the distance the column has moved (e.g., the similarity score of 2_a - 2 is slightly higher than the one of 3_a - 3). The centroid features have the biggest influence on the similarity score. If the pose discrepancies between the as-designed and as-built are significant, as outlined in the paper introduction, these scores will not suffice for similarity-based updates across large-scale distances.

(a) Feature comparison between $f_{i,design}$ and $f_{i,built}$

(b) Topologically enriched feature comparison between $f^{'}_{i,design}$ and $f^{'}_{i,built}$ features

Fig. 7: Computed cosine similarity scores. For visualization purposes, only a part of the elements is displayed. The brighter the color, the higher the similarity score. Abbreviations COL: Column, WL: Wall, SLB: Slab, DR: Door, WIN: Window.

When the node features of both graphs are enriched with topological information via graph convolutions, the results look more promising. The topological enrichment increased the similarity score of the non-congruent, yet the same columns averaged 80 to 92%. Since all the columns have a rectangular format and are modelled identically, the high similarity score of e.g. column 2_a* with column 1 and 3 is also explicable. From the results it can be deduced that column 2_a is indeed the same as column 2a and column 3_a and 3_b the same as column 3. The similarity scores across types mostly decrease further and are less correlated to the exact location. In some cases the similarity score for wall - column pairs increases. The common connection to a slab element could cause this behaviour.

As suggested by the results above, it can be said that the topological most likely decreases the importance of the location-based features and increases the influence of the shape and topological features. In our experiments this happens in a favorable way such that the as-built columns can be matched with their as-designed equivalent.

4.2 Sub-element clusters

For this experiment, a common output of a point clustering algorithm is used to formulate the graph. The used 2-layered GCN is capable of propagating feature information as far as two clusters. For evaluation, the similarity score was computed between all cluster embeddings (f_i or $f^{'}_i$) matching to one as-built ground

truth and their matching as-designed element. Depending on how many clusters the algorithm made, the mean of all the similarity scores was computed. For the left column in Fig. 4b for example, the mean was calculated, whereas for the right column it was not neaded. The average per-type matching scores for each element cluster with it's matching as-designed embedding is reported in Table 1.

Table 1: Averaged similarity scores over all elements of the type based on shape alone and the addition of topological information (*).

Class	mean similarity score	mean similarity score*
Slab	0.78	0.91
Column	0.45	0.78
Wall	0.62	0.9
Window	0.32	0.52
Door	0.56	0.61

Again it becomes visible that including topological information propagated across the edges of the graphs increases the similarity scores. With topological enrichment, the average scores for columns are slightly lower than for the other well-performing types, such as slabs and walls. Especially slabs and walls and, to a lesser degree also, columns are typically clustered in large point clusters due to their rather homogenous flat surfaces. This allows our 2-layered GCN to inform each cluster about the relevant across-building-element typologies.

Likewise windows and doors would typically show as multiple point clusters because of their frames which is most likely the reason for the worse results. Deeper graph networks might be able to increase the results in cases where finer segments are present in the clustered point cloud.

5 Conclusion and outlook

The detection of the deviations between as-designed and as-built status to update the digital representation of the building under construction promises significant added value while remaining one of the main challenges of automation. However, the difficulty of reconstructing accurate information-rich BIM models from as-built point clouds can impede the usability of such reconstructed models. A combination of large-scale similarity-based pose updates and metaheuristic geometric optimization techniques could prove top-down approaches favorable over bottom-up reconstruction in terms of the information depth of the resulting models.

Our suggested method shows the first evidence that similarity-based geometry retrieval is very promising for solving Scan-vs-BIM problems with significant pose deviations and is apt to complement state-of-the-art methods. We propose using graph representations of both the as-designed model and the as-built

representation, where node feature vectors reflect the individual element's characteristics and their adjacency relationships to other building elements. Based on this, we apply a cosine similarity metric to assess the similarity between the nodes of both graphs. This enables us to find matching pairs among the elements having deviations between as-designed and as-built, which subsequently allows updating the BIM model correspondingly. The method shows stable similarity scores above 90% for columns in simulated as-built representations. The method entirely relies on propagating topological information in a GCN inference scenario. Formulating the similarity score to include edge features instead of only features could further make the method more robust.

To fully prove the method as applicable, the experiments will be conducted on real PCD in the future. Given the extensive application fields of a similarity-based object (as presented in Section 2.3) retrieval from queries, the authors suggest formulating the graph matching as a learning problem. Thereby, even less obvious matching elements, such as in the case of severe occlusions, might be automized.

6 Acknowledgements

The work presented in this paper is funded by a Georg Nemetschek scholarship which is gratefully acknowledged.

References

1. Bosché, F., Ahmed, M., Turkan, Y., Haas, C.T., Haas, R.: 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). https://doi.org/10.1016/j.autcon.2014.05.014, http://dx.doi.org/10.1016/j.autcon.2014.05.014
2. Braun, A., Tuttas, S., Stilla, U., Borrmann, A.: Incorporating knowledge on construction methods into automated progress monitoring techniques. 23rd International Workshop of the European Group for Intelligent Computing in Engineering, EG-ICE 2016 pp. 1–11 (2016)
3. Braun, A., Tuttas, S., Borrmann, A., Stilla, U.: Improving progress monitoring by fusing point clouds, semantic data and computer vision. Automation in Construction **116**(March), 103210 (2020). https://doi.org/10.1016/j.autcon.2020.103210, https://doi.org/10.1016/j.autcon.2020.103210
4. Braun, A., Tuttas, S., Borrmann, A., Stilla, U.: A concept for automated construction progress monitoring using BIM-based geometric constraints and photogrammetric point clouds. Journal of Information Technology in Construction **20**(November 2014), 68–79 (2015)
5. Collins, F.C., Ringsquandl, M., Braun, A., Hall, D.M., Borrmann, A.: Shape encoding for semantic healing of design models and knowledge transfer to Scan-to-BIM. Proceedings of the Institution of Civil Engineers - Smart Infrastructure and Construction pp. 1–21 (2022). https://doi.org/10.1680/jsmic.21.00032

6. Daum, S., Borrmann, A.: Processing of topological BIM queries using boundary representation based methods. Advanced Engineering Informatics **28**(4), 272–286 (2014). https://doi.org/10.1016/j.aei.2014.06.001, http://dx.doi.org/10.1016/j.aei.2014.06.001
7. Hackel, T., Wegner, J.D., Schindler, K.: Contour detection in unstructured 3D point clouds. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2016-Decem**, 1610–1618 (2016). https://doi.org/10.1109/CVPR.2016.178
8. Kipf, T.N., Welling, M.: Semi-Supervised Classification with Graph Convolutional Networks pp. 1–14 (2016), http://arxiv.org/abs/1609.02907
9. Landrieu, L., Boussaha, M.: Point cloud oversegmentation with graph-structured deep metric learning. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2019-June**, 7432–7441 (2019). https://doi.org/10.1109/CVPR.2019.00762
10. Landrieu, L., Simonovsky, M.: Large-Scale Point Cloud Semantic Segmentation with Superpoint Graphs. CoRR **abs/1711.0** (2017). https://doi.org/10.1109/CVPR.2018.00479
11. Mafipour, M.S., Vilgertshofer, S., Borrmann, A.: Deriving Digital Twin Models of Existing Bridges from Point Cloud Data Using Parametric Models and Metaheuristic Algorithms. EG-ICE 2021 Workshop on Intelligent Computing in Engineering, Proceedings pp. 464–474 (2021)
12. Maheshwari, P., Chaudhry, R., Vinay, V.: Scene Graph Embeddings Using Relative Similarity Supervision. 35th AAAI Conference on Artificial Intelligence, AAAI 2021 **3B**, 2328–2336 (2021)
13. Rausch, C., Haas, C.: Automated shape and pose updating of building information model elements from 3D point clouds. Automation in Construction **124**(August 2020), 103561 (2021). https://doi.org/10.1016/j.autcon.2021.103561, https://doi.org/10.1016/j.autcon.2021.103561
14. Rausch, C., Nahangi, M., Haas, C., Liang, W.: Monte Carlo simulation for tolerance analysis in prefabrication and offsite construction. Automation in Construction **103**(November 2018), 300–314 (2019). https://doi.org/10.1016/j.autcon.2019.03.026, https://doi.org/10.1016/j.autcon.2019.03.026
15. Talebi, S., Koskela, L., Tzortzopoulos, P., Kagioglou, M., Rausch, C., Elghaish, F., Poshdar, M.: Causes of Defects Associated with Tolerances in Construction: A Case Study. Journal of Management in Engineering **37**(4) (2021). https://doi.org/10.1061/(asce)me.1943-5479.0000914
16. Turkan, Y., Bosche, F., Haas, C.T., Haas, R.: Automated progress tracking using 4D schedule and 3D sensing technologies. Automation in Construction **22**, 414–421 (2012). https://doi.org/10.1016/j.autcon.2011.10.003, http://dx.doi.org/10.1016/j.autcon.2011.10.003
17. Tuttas, S., Braun, A., Borrmann, A., Stilla, U.: Validation of BIM components by photogrammetric point clouds for construction site monitoring. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **2**(3W4), 231–237 (2015). https://doi.org/10.5194/isprsannals-II-3-W4-231-2015
18. Tuttas, S., Braun, A., Borrmann, A., Stilla, U.: Acquisition and Consecutive Registration of Photogrammetric Point Clouds for Construction Progress Monitoring Using a 4D BIM. Photogrammetrie, Fernerkundung, Geoinformation **85**(1), 3–15 (2017). https://doi.org/10.1007/s41064-016-0002-z

19. Wang, J., Wu, Q., Remil, O., Yi, C., Guo, Y., Wei, M.: Modeling indoor scenes with repetitions from 3D raw point data. CAD Computer Aided Design **94**, 1–15 (2018). https://doi.org/10.1016/j.cad.2017.09.001, http://dx.doi.org/10.1016/j.cad.2017.09.001
20. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph Cnn for learning on point clouds. ACM Transactions on Graphics **38**(5) (2019). https://doi.org/10.1145/3326362
21. Wang, Z., Sacks, R., Yeung, T.: Exploring graph neural networks for semantic enrichment: Room type classification. Automation in Construction (December), 104039 (2021). https://doi.org/10.1016/j.autcon.2021.104039
22. Werbrouck, J., Pauwels, P., Bonduel, M., Beetz, J., Bekers, W.: Scan-to-graph: Semantic enrichment of existing building geometry. Automation in Construction **119**(2020), 103286 (2020). https://doi.org/10.1016/j.autcon.2020.103286, https://doi.org/10.1016/j.autcon.2020.103286
23. Winiwarter, L., Esmorís Pena, A.M., Weiser, H., Anders, K., Martínez Sánchez, J., Searle, M., Höfle, B.: Virtual laser scanning with HELIOS++: A novel take on ray tracing-based simulation of topographic full-waveform 3D laser scanning. Remote Sensing of Environment **269**(November 2021) (2022). https://doi.org/10.1016/j.rse.2021.112772