Graph-based linking of point cloud and BIM elements for extended DT integration

Collins F.C., Braun A., Borrmann A.
Technical University of Munich, Germany
fiona.collins@tum.de

A Digital Twin (DT) of a building is marked by a regular inflow of spatial and visual data to keep its digital representation up to date. A seamless integration of this data with the existing geometric semantic DT is key. To link the two data sources on building element-level, existing methods use 1. a proximity-based approach which relies on good spatial registration, or, 2. a key-point matching approach using characteristic geometry for specific elements. Positional and geometric deviations between the two data sources can be a challenge. In this paper, a method is proposed to link semantic instances in point cloud data to their DT equivalent using high-level geometric features and topological relationships in the building. The method is demonstrated using a small-scale case study that contains positional and geometric deviations. It is demonstrated that the method is robust to positional deviations of doors and deviating geometry.

1 Introduction

The digital Twin (DT) for buildings is a promising concept for improving transparency and informed decision making in the built environment. A continuous data flow between the physical building and its digital counterpart lies at its core (Kritzinger *et al.*, 2018). In practice however, the complete monitoring of the physical building across lifecycle phases and disciplines is hard to achieve because of the vast nature of activities and poor digitalization of processes. The domain has thus adopted an approach of monitoring the physical building selectively and "frequently-enough" (Brilakis *et al.*, 2020). The specifics are defined by the use case in question.

A common and highly precise way of monitoring and documenting the changing appearance of the physical building is to acquire coloured Point Cloud Data (PCD) by LiDAR sensors along the way. A four-step process takes place: 1. PCD of an area-of-interest is acquired, 2. the raw data is processed (filtered, semantically enriched etc.), 3. the PCD is integrated with the existing, potentially outdated, DT model, and 4. potential differences are analysed for decision-making. In this paper, we focus on step 3, data integration. Integration describes the process of fusing data from multiple streams (Sacks *et al.*, 2020) and creating links between heterogeneous representations of the same building elements. For instance, we can link the acquired as-built point cloud representation of elements to their historic element representation (e.g., an as-designed geometry from BIM). This link enables conclusions about the timeliness of construction or the adherence to codes regarding the position or product type of a specific element.

In the common integration process, PCD data is spatially registered with the DT model by a manual approach or the use of the Iterative Closest Point (ICP) algorithm. The points are matched with their geometric representation from the DT by a measure of spatial proximity. This process of registration and matching at point-level is time consuming, especially if many separate scans of areas of interest must be processed. In the context of reoccurring acquisitions for DTs, the need for robust automation becomes apparent. However, the automated matching mainly relying on spatial proximity can be technically challenging in case of poor registration

results or undocumented positional deviations of elements (Chuang and Yang, 2023). Meanwhile, meaningful processing of PCD such as semantic- and instance segmentation is becoming a mature field of research for major structural elements. This motivates the data integration to make use of semantic information and move from a pointwise linking to a semantically guided one.

In summary, a comprehensive and automated integration of acquired PCD into an existing geometric DT model is key to analysing the evolution of a building and using it for down-stream tasks such as model update or compliance checks. In this paper we aim to abstract the element linking process with the help of geometric-topologic graphs. We thereby circumvent the need of spatial proximity of the building elements and instead use a combination of geometry and the topologic relationships to establish the link between the element representations. With our graph-based method individual scans of areas of interest can be linked to their DT counterpart elementwise despite slight positional and geometric element deviations.

2 Related work

Comparing two different data streams can lead to important conclusions about a building. On construction sites for example the progress of assembly (Braun *et al.*, 2020) and demolition (Meyer, Brunn and Stilla, 2022) is documented with images and point clouds scans. By comparing this data with the as planned 4D BIM model, reports on the project timeliness, productivity, or quality control can be made. In the context of the DT such reports are the result of an integrated database from which additional insights can be gained from a vast knowledge base. Several authors have emphasized the use of graphs to represent highly connected data with frequent update cycles. Graphs offer a convenient representation for building data by organizing it as a set of nodes and edges and are effective to update and change operations. In this section, we present relevant works from literature integrating spatial and visual data with existing (usually BIM models). Then we show approaches that use graphs to encode building data and review the graph matching problem as a tool for linking two data sources.

2.1 Element-matching without topologic information

Linking geometric-semantic models and as-built PCD or images has been integrated into the BIM process and has shown its benefits in terms of increased process transparency and reliability (Meyer, Brunn and Stilla, 2022). By aligning the two data sources in a common spatial reference system, the spatial proximity allows for a direct comparison. For photogrammetric point clouds of construction sites, the immediate vicinity of an element can be analysed by 3D-projections or occupancy-grids. The methods are made more robust and accurate with the use of process knowledge, object detection and visibility analysis (Braun *et al.*, 2020; Vincke and Vergauwen, 2022). Monitoring indoor scenes is however a more challenging task because of limited observability and weak georeferencing. Furthermore, uncertainty is attached to the geometric-semantic model with respect to Level of Geometry (LoG) and Level of Accuracy (LoA) (Meyer, Brunn and Stilla, 2022). To tackle positional uncertainty between the two data sources Chuang and Yang (2023) propose a framework incorporating geometric key points and plane matches to perform the linking of building elements despite deviating locations within a room. The geometric uncertainty coupled with positional uncertainty as for example with differing levels of LoG, remains a challenge.

2.2 Encoding building relations in graphs

Preceding works treat the individual building elements independently and mostly out of context of their spatial and topological surroundings. In buildings however the information included in the spatial and semantic neighbourhood is key to the domain expert's work. Encoding those relations explicitly in property graphs to use by machine learning and/or pattern matching applications has been suggested by Sacks, Girolami and Brilakis (2020) and is gaining momentum ever since. Esser, Vilgertshofer and Borrmann (2022) for instance, developed a graph-based method for update tracking and federation of BIM models at building object level.

Similarly, it has been shown that encoding spatial and topologic relationships in point cloud processing brings the benefit of capturing long range relationships between element representations (Landrieu and Simonovsky, 2017). Typically, in cluttered scenes building elements are only partially observed. The result are occlusions obscuring the true whole geometry of building elements in the PCD. Deep Learning based enrichment methods for point clouds and images have shown to benefit from long-range edge connections (Landrieu and Simonovsky, 2017).

2.3 Graph matching

The recommended data format of different DT data streams converges to graphs because of their extendibility and ability to model relationships explicitly in their edge connectivity. The question of how to best integrate those streams of graphs into a consistent database remains. For instance, during the BIM process changes induced by one modelling party need to be integrated into the, possibly contradicting, main model. For this, Esser, Vilgertshofer and Borrmann, 2022 propose a property graph-based method that relies on knowing the match of a common parent node and then analysing the subgraphs for changes. A similar challenge arises when multiple geometric models from different building disciplines (e.g., architecture and structural graphs) are to be linked while showing different domain-specific geometry (Wang, Ouyang and Sacks, 2023).

Matching nodes between graphs has found multiple applications in Computer Vision to align images or point clouds to each other. Graph matching has been heavily investigated in literature with graph-edit-distance, maximum-common-subgraph (MCS) or by formulating it as a quadratic assignment problem (Fey *et al.*, 2020). Since all those approaches are NP-hard, using them in the context of large buildings seems unfeasible without reducing the problem space with prior knowledge. Assuming a matching parent node, geometric affinity or reducing the search space to one semantic class enriched during processing are means to apply such methods to the domain. Furthermore, such approaches generally consider the graph structure as the relevant matching criteria and do not leave aside node features.

Graph-based matching problems have most recently been used in large-scale problems involving LiDAR and BIM data. Pramatarov et al., (2022) for instance utilise a series of graph matching steps for pose estimation between LiDAR scans. More importantly Shaheer et al., (2023) employ their own graph matching steps to achieve robot localisation in buildings by matching a point cloud graph to a BIM-based architectural graph. The problem space is reduced by considering a semantic subset of nodes or by matching on a higher level such as rooms.

3 Methodology

In this section we outline our method for integrating PCD with existing DT models (here a BIM model) with geometric-topologic graphs. We revisit the graph formulation module previously

suggested (Collins et al., 2022) and describe the extensions made to it. Thereafter we outline the graph matching method used for establishing links (or matches) between the graphs. In brief, the methodology used in this paper consists of 2 steps: 1. Graph formulation for the two data streams, 2. Graph matching with geometric features and topology (Fig. 1).

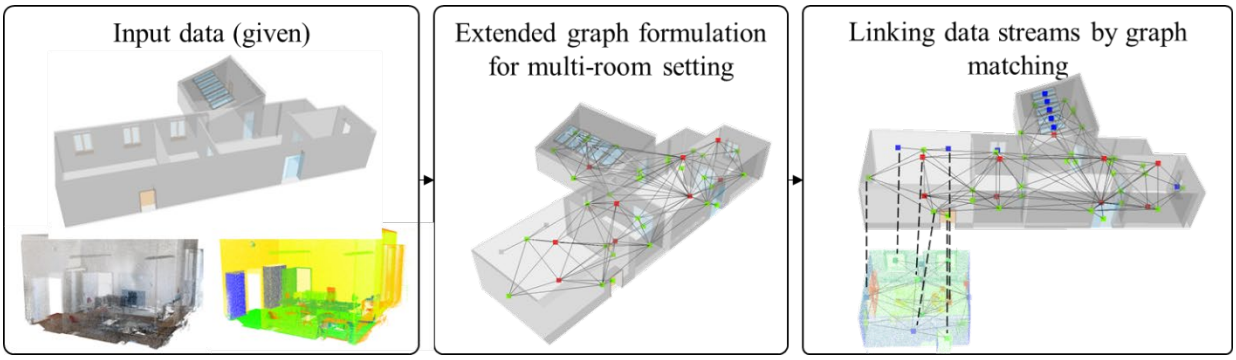


Figure 1: Methodologic overview.

3.1 Revisiting graph formulation

The graphs for the two data streams (PCD and DT, here BIM) are formulated such that they include both geometric properties about the elements and their topological connectivity. In general, we formalize the graphs for the two streams as $G = (\mathbf{V}, \mathbf{W})$, where **V** is the set of nodes and **W** the distance-weighted adjacency of the nodes. Each node is defined to represent one building element or sub parts of it. A set of features **F** that include high-level geometric characteristics about the geometric shape are attached to the nodes. The methods of assembling **V, W, F** differ between the data streams, yet the following holds for both subsequent paragraphs: The set of geometric features includes 1. the semantic class of the element, 2. the two first principal components (PC) obtained with a Principal Components Analysis (PCA) on 1000 randomly sampled points and, 3. the shapes extents in the direction of the principal components: $\boldsymbol{F} = \{class\ label, \boldsymbol{PC_1}, \boldsymbol{PC_2}, extent_1, extent_2\}$. For easy visualisation the building element's centroids are computed and used to place the nodes in space. It is noted, however, that the building element's centroid is not used elsewhere and therefore allows our method to circumvent the spatial registration step other methods use for linking PCD with DT data.

3.2 From BIM model to Graph G_{DT}

Nodes and node features. With help of the model parsing library IfcOpenShell, the building elements included in the model are extracted and their features computed by sampling a set of 1000 points on the element surface.

Weighted adjacency. The spatial-topological query language QL4BIM (Daum and Borrmann, 2014) is used to extract the directly adjacent building elements with the help of the "touching" operator. These element connectivities are weighted by the spatial distance between the node centroids and together form the matrix **W**.

Since we extend our method to cover multi-room settings the creation of G_{DT} requires an extra step. Commonly in BIM models slab and wall geometry is modelled to span multiple rooms, which might or might not reflect the as-built span of the concrete slab. In our case these elements result in single nodes with very high edge density ranging across a wide building area.

This is unsuitable for the graph matching process both in terms of geometric features as well as graph connectivity because the graphs look to dissimilar.

We therefore propose to sub-divide the node representations of slabs and walls where multiple rooms adjoin. For that we revisit the information extraction step from IFC with the aim of dividing large wall and slab nodes into a set of child nodes. First the directional operators *Above, Bellow, WestOf, EastOf, SouthOf* and *NorthOf* from QL4BIM are used to extract the relative positions between slabs/walls and their touching spaces. The space's footprint along the respective directional axis is used to create a sub view node of the respective large multi-room element. In the following we will refer to those elements as the target elements. The process is illustrated in Fig. 2.

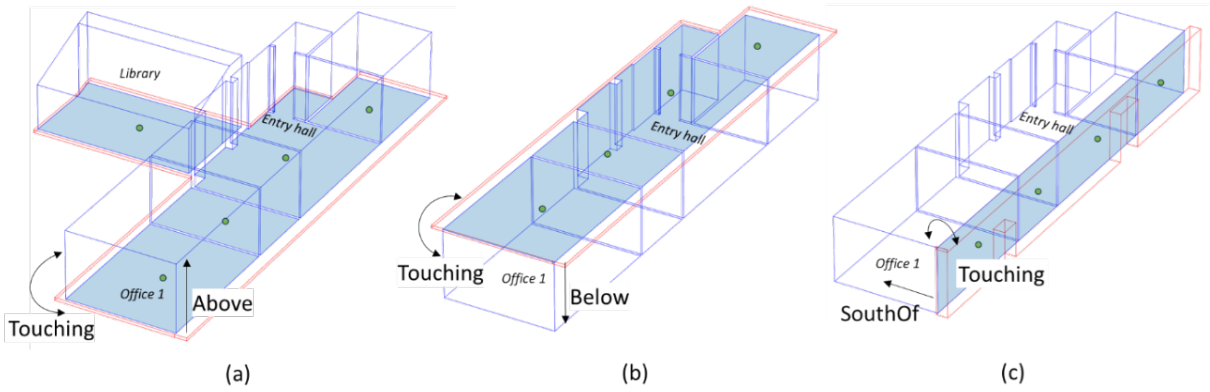


Figure 2: Visualisation of the method for slab and wall node division with topologic and directional operators. The red-outline shows target element, the blue outlines show touching spaces, and the blue-shaded polygons are the calculated footprints used to create sub-element nodes (in green) in G_{DT}. (a) Division of lower slab. (b) covering slab. (c) one exemplary wall element.

For each sub-element a new node is introduced. The footprint is used to recalculate the geometric feature matrix **F** as well as the geometric centroid. The adjacencies of the sub view nodes are inherited to be the shared "touching" elements of the initial node (e.g., multi-room spanning wall node) and the adjacent space node. The initial element node loses all its adjacencies to surrounding element nodes and instead gets a direct edge to its sub-element nodes. This process of managing the adjacencies is illustrated in Fig 3.

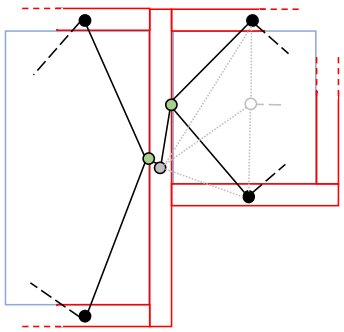


Figure 3: Division of multi-room spanning nodes (grey) into sub-element nodes (green). Deleted edges are shown in grey and newly added edges in solid black. The dotted lines show the "touching" edges towards shared adjacent elements of the initial wall node and the space node. Those define the establishment of new edges (solid black).

The current version of our graph dividing algorithm does not work well when a wall immerses into the room and all 3 faces of the wall are visible from one single space. Furthermore, the current state of our implementation requires manual input for the side-wards directional operators of QL4BIM. We plan to tackle those limitations in future works.

3.3 From PCD to Graph G_{PCD}

The PCD in this work is assumed to be acquired room-wise, without registration between individual room scans. Practically, this scenario might occur when, for a particular use case, only certain rooms in the DT require an accurate LiDAR-based assessment or the rooms are acquired at different moments in time or with differing equipment.

In line with the DT data procurement pipeline this work focuses on the data integration step and assumes the PCD to have undergone some processing steps such as point cloud semantic segmentation. Thereby the points are assigned to the semantic class encoding their affiliation to a building element class such as wall, floor, ceiling, window, door etc. Additionally, we assume a RANdom SAmple Consensus (RANSAC)-based instance segmentation of the walls and a density-based instance segmentation for other elements.

Nodes and node features. As suggested in Collins et al. (2022), G_{PCD} is formulated using the superpoint graph approach. One superpoint/node is defined for one semantic instance in the PCD. However, since PCD processing might result in several clusters for one element in case of occlusions, one building instance might be represented by one or several nodes. Like G_{DT}, the feature matrix **F** is calculated by randomly sampling 1000 points from each superpoint and calculating the relevant features.

Weighted adjacency. The edges are generated by a 3D Delaunay triangulation approach on the PCD (Landrieu and Simonovsky, 2017). To keep the graph as close as possible to the direct adjacency represented in G_{DT}, edges are omitted for all elements having Delaunay edges longer than 0.5m distance. The distance of the remaining edges is included in the adjacency matrix **W**.

3.4 Graph matching

The next step is to match the PCD represented as G_{PCD} and the DT data (G_{DT}) such that the matched node pairs reflect the true correspondences between the data streams. In the process, each node in G_{PCD} is compared and potentially matched to a corresponding node in G_{DT} by a series of criteria. Those criteria firstly include a similarity computation of the geometric node features between the graphs. However, in buildings where indoor layouts often are of repetitive nature, a purely geometry-based match could lead to confusion. A room dividing wall node isolated from its context, could yield ambiguous geometric correspondences by showing very high similarity scores for several nodes in G_{DT}. Therefore, in a second and third step, our method integrates an iterative node neighbourhood consensus. If one node of G_{PCD} is matched to a node in G_{DT} its neighbours in G_{PCD} should match to the corresponding neighbours in G_{DT} too. Both the geometric similarity as well as the iterative neighbourhood consensus lead to the final match. Finally, we form a bipartite graph connecting nodes of G_{PCD} with nodes of G_{DT}. Fig.4 shows an overview of the method. We note that only the main 5 structural classes wall, door, window, floor, and ceiling are used for the process. The detection of those classes in PCD is generally assumed to be robust and therefore assumed to be a good input to our method.

Initial and ranked candidate matches. In step one we compute the geometric similarities between all nodes of the same semantic class of G_{PCD} and G_{DT} with the cosine similarity. The result is a per-class ranked list of candidates matching pairs with a matching score M. At the top of these candidate lists the pair with the highest geometric matching score is located. In the following, an iterative update process is deployed to steadily ensure the neighbourhood consensus.

Refining the candidate matches with MCS. To that end the highest-ranked candidate pairs are picked and the MCS is computed in the 2-hop induced subgraphs. Since our graphs are not very dense, this order neighbourhood is manageable for the MCS computation. The node pairs contained in the MCS are rewarded by adding to the matching score M computed in the previous step. The depth and the amount of other top candidate matches in the MCS is used to weight the reward. The ranked list of candidates matching pairs is re-sorted and a next iteration starts. As soon as no resorting happens, we assume a true graph match between G_{PCD} and G_{DT}.

When computing the MCS it happens that more than one structurally equivalent graph is found (graph isomorphism). In this case no rewards are committed, and the next node pair is evaluated.

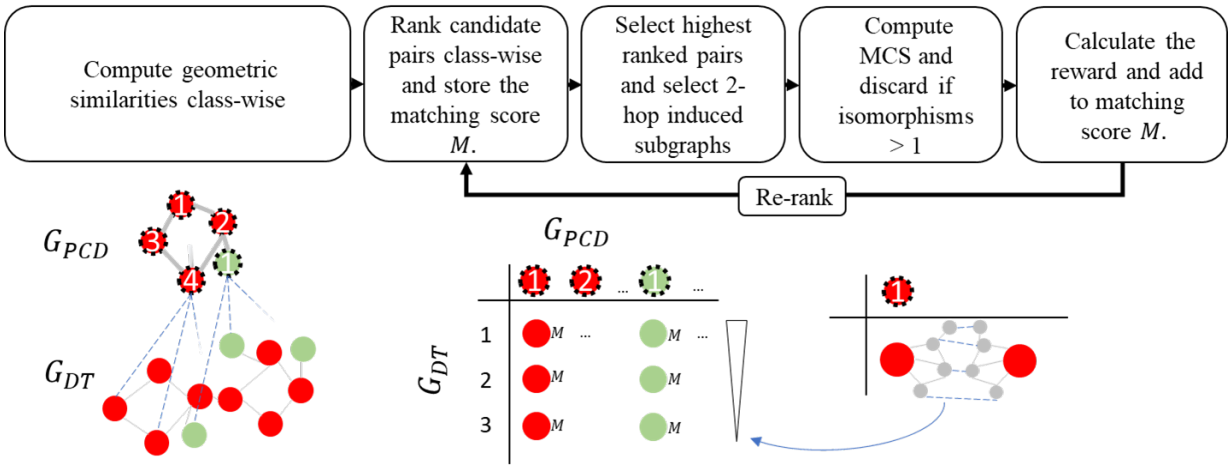


Figure 4: Visualisation of the method for graph matching.

4 Experiments and discussion

4.1 Case study

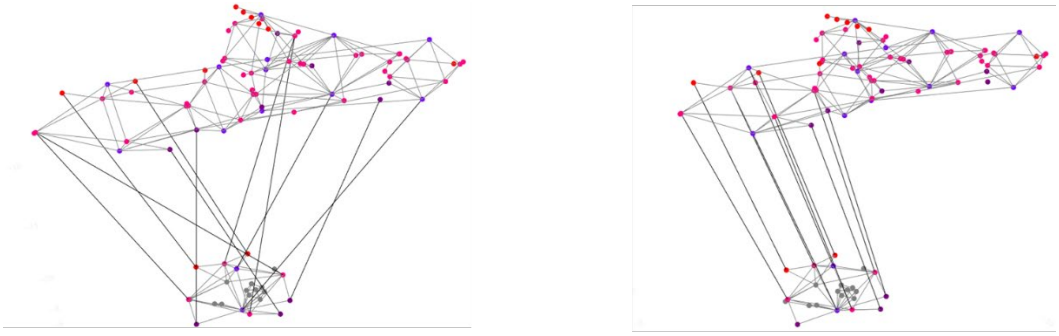
We demonstrate our method on a part-area of the main building of TU Munich, consisting of three rooms. The geometric-semantic model was modelled in a BIM authoring tool and lays the basis for generating G_{DT}. The area was scanned with a LiDAR scanner, yet the scans of the individual rooms have not been registered relative to each other, nor to the geometric-semantic model. Since the focus of this work is data integration, we assume the PCD to have undergone processing and enrichment steps beforehand: firstly, semantic segmentation to assign each point to a semantically coherent object and secondly, an instance segmentation.

The goal is to match as many nodes as possible from G_{PCD} with nodes in G_{DT} based on their geometric similarity and their neighbourhood connectivity in the graph. To show the boundaries of our method we perform slight modifications in selected element locations in the geometric-semantic model.

4.2 Optimum assignment

The first step of our method produces candidate matches based on geometry. Since the geometric descriptor is not very sophisticated it is sensitive to noise in the point cloud semantic segmentation and occlusions. Indeed, on average only 27% of all highest-ranked geometric matches are correct (see Fig. 5a). However, when looking at the top-3-ranked geometric matches the correct pair is listed in 92% of the cases. For geometrically prominent elements

and low point cloud occlusions (e.g., the slanted ceiling in the *Library*) the geometric match is correct. This lets us validate the method as a first guess for the next step.



(a) (b)

Figure 5: Illustration of the graph matches for *Office 1* where G_{PCD} is the lower graph, and G_{DT} the one at the top. (a) match with geometric features and (b) after the neighbourhood consensus.

Table 1: graph matching results for the three rooms.

	Office 1	**Library**	**Entry hall**
# of iterations	9	13	Not converging
# of correct node matches before and after neighbourhood consensus	Before: 1/5 After: 7/8	Before: 4/15 After: 13/15	Before: 4/18 After: 5/18 (manual stop)

The library has one wall entirely covered by bookshelves. However, there is still a narrow piece of wall between the roof and the bookshelves. The graph formulation for G_{PCD} succeeds in capturing the relationship of this narrow wall piece to its surrounding walls however not to the floor. Despite this complex, yet very typical setting in PCD our method succeeds in matching both the wall and floor node. This is possible thanks to the iterative approach that assesses the matching for each node from multiple sides of its neighbourhood. Since the geometric matching score of the narrow wall element is proportionally low, a higher set of iterations is needed for matching the *Library* compared to *Office 1*.

On the other hand, the *Entry Hall* is a large open space with a complex space bounding wall (see Fig. 2). As pointed out earlier, our graph formulation module G_{DT} fails to properly capture the walls in this scenario. And thus, both the geometric match and the neighbourhood consensus method do not perform well (see Tab. 1). Additionally, the doors in the *Entry Hall* are large glass structures with inside window frames, which, well-knowingly are not well captured by LiDAR technology. Since the graph formulation for G_{PCD}, does not include any node merging strategy, the door is represented by several instead of one single node. Furthermore, since those doors are large, the connectivities of those nodes differ.

4.3 Robustness to positional and geometric deviations

Commonly the linking of two element representations between data streams is used to report deviations between the historic DT model and the as-is PCD or to perform updates. For *Office 1* the door was artificially deviated by 2m to the left (see Fig. 6). Structurally the graph remains identical to the one without deviations. Our method therefore produces the same matching result.

Geometric deviations might occur because of varying LoG between the two data sources or because of inherent traits of the scan process (e.g., open doors). Indeed, the geometric feature description of the door in G_{PCD}, illustrated in Fig. 6, will cause a wrong match. However, the surrounding elements are robust enough to update the door's matching score to become the highest and therefore yield the correct match. A more meaningful geometric shape descriptor would be beneficial to render our first candidate pairs more robust and reduce the number of iterations in our method.

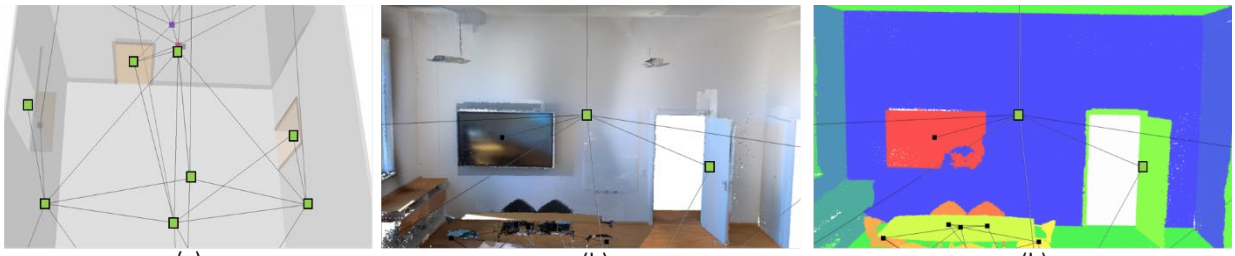


Figure 6: Illustrations of apparent deviations in PCD vs DT model in Office 1. The matched graph nodes are marked in green. (a) DT model with G_{DT} (b) PCD with G_{PCD}, (c) Processed PCD with G_{PCD}.

Limitations. Our method relies on the fact that most connections between the relevant elements (wall, floor, ceiling, door, and window) are present in both graphs. Even though the iterative method revisits each node pair from different neighbouring sides and therefore having some redundancy, potential issues in the graph formulation modules will result in less accurate results. The test on the *Entry Hall* exemplifies this. The graph formulation must therefore be robust and potentially include a node merging mechanism additionally to node splitting. In future we plan to enhance the graph formulation and test on more complex sites. With respect to the neighbourhood consensus, we currently bypass isomorphic MCS by skipping them and have no strategy for symmetrically equivalent subgraph matches. Our method needs to encompass the neighbourhood consensus with a more diverse set of element semantics to further narrow down the solution of structural equivalence.

5 Conclusion and future work

Integration of different DT data streams is key to the concept of DT for buildings. Frequently, decisions about the current state of a building are tied to the integration of spatial and visual as-is data in form of point clouds with geometric-semantic data from a previous DT model. Integrating those two data streams traditionally happens in a point-wise manner where proper registration is paramount for proximity-based element matching. Alternatively, our method uses geometry, semantics and topology encoded in graphs to represent and link the PCD and DT. This abstraction allows to leverage the topologic arrangements of elements and loosens the need for spatial proximity or exact geometric accordance in the models. However, our method is highly dependent on the quality of the DT and PCD graph formulation, especially with respect to large open spaces and multi-node element representations from point cloud processing. The presented method will be developed and evaluated further to assess its potential for generalising to adequately sized real-world scenarios.

6 Acknowledgements

We would like to express our gratitude to Prof. Georg Nemetschek for his support in funding this research with a scholarship.

References

Braun, A. et al. (2020) 'Improving progress monitoring by fusing point clouds, semantic data and computer vision', Automation in Construction, 116(March), p. 103210. Available at: https://doi.org/10.1016/j.autcon.2020.103210.

Brilakis, I. et al. (2020) Built Environment Digital Twinning.

Chuang, T.Y. and Yang, M.J. (2023) 'Change component identification of BIM models for facility management based on time-variant BIMs or point clouds', Automation in Construction, 147(June 2022), p. 104731. Available at: https://doi.org/10.1016/j.autcon.2022.104731.

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

Daum, S. and Borrmann, A. (2014) 'Processing of topological BIM queries using boundary representation based methods', Advanced Engineering Informatics, 28(4), pp. 272–286. Available at: https://doi.org/10.1016/j.aei.2014.06.001.

Esser, S., Vilgertshofer, S. and Borrmann, A. (2022) 'Graph-based version control for asynchronous BIM collaboration', Advanced Engineering Informatics, 53(June). Available at: https://doi.org/10.1016/j.aei.2022.101664.

Fey, M. et al. (2020) 'Dgmc', Iclr, pp. 1–23.

Kritzinger, W. et al. (2018) 'Digital Twin in manufacturing: A categorical literature review and classification', IFAC-PapersOnLine, 51(11), pp. 1016–1022. Available at: https://doi.org/10.1016/j.ifacol.2018.08.474.

Landrieu, L. and Simonovsky, M. (2017) 'Large-Scale Point Cloud Semantic Segmentation with Superpoint Graphs', CoRR, abs/1711.0. Available at: https://doi.org/10.1109/CVPR.2018.00479.

Meyer, T., Brunn, A. and Stilla, U. (2022) 'Change detection for indoor construction progress monitoring based on BIM, point clouds and uncertainties', Automation in Construction, 141(February), p. 104442. Available at: https://doi.org/10.1016/j.autcon.2022.104442.

Pramatarov, G. et al. (2022) 'BoxGraph: Semantic Place Recognition and Pose Estimation from 3D LiDAR'.

Sacks, R. et al. (2020) 'Construction with digital twin information systems', Data-Centric Engineering, 1. Available at: https://doi.org/10.1017/dce.2020.16.

Sacks, R., Girolami, M. and Brilakis, I. (2020) 'Building Information Modelling, Artificial Intelligence and Construction Tech', Developments in the Built Environment, 4(March), p. 100011. Available at: https://doi.org/10.1016/j.dibe.2020.100011.

Shaheer, M. et al. (2023) 'Graph-based Global Robot Localization Informing Situational Graphs with Architectural Graphs'.

Vincke, S. and Vergauwen, M. (2022) 'Vison based metric for quality control by comparing built reality to BIM', Automation in Construction, 144(November 2021), p. 104581. Available at: https://doi.org/10.1016/j.autcon.2022.104581.

Wang, Z., Ouyang, B. and Sacks, R. (2023) 'CBIM: object-level cloud collaboration platform for supporting across-domain asynchronous design'.