QUANTIFYING THE CONFIDENCE IN MODELS OUTPUTTED BY SCAN-TO-BIM PROCESSES

Shirin Malihi, Frédéric Bosché & Martin Bueno Esposito
School of Engineering, University of Edinburgh

ABSTRACT: *3D spatial data is increasingly employed to generate Building Information Models (BIMs) by extension digital twins for various applications in the architecture, engineering, and construction (AEC) sector such as project monitoring, engineering analyses, retrofit planning, etc. The outputted models of Scan-to-BIM processes should satisfy pre-defined levels of quality. In the case of emerging automated Scan-to-BIM solutions, users however currently need to check all generated geometry manually, which is time-consuming. What would help users is if the automated systems could also provide a level of confidence in the detection and modelling of each element. In this paper three generic indicators are defined for analysing the reliability of the generated 3D models: $I_{coverage}$ estimates the portion of the surface of the modelled element that can be explained by the input point cloud. $I_{distance}$ defines the closeness of the generated element models to the input point cloud. The confidence of the generated 3D local models can be computed by combining the two aforementioned indices. The proposed indicators are assessed using actual examples and comparisons are conducted between automatically generated 3D BIM models and 3D models generated manually by a BIM modeler.*

Keywords: *BIM, point cloud, confidence, indoor modelling, wall, digital twin*

1 INTRODUCTION

Digital twinning of built environment assets is a modern data-driven process with benefits to improving performance and productivity within the Architecture, Engineering, and Construction (AEC) industry. It affords a multi-dimensional view of how an asset will perform by simulating, predicting, and making decisions based on real-world conditions (Boje et al., 2020). At the information (or data) level, it represents the information in a useful, structured form of description. At a higher level, it applies tools that make use of that information to provide diagnostics of why something might be happening, predict the possible future outcome, and decide the action based on the objectives. Optimising construction project execution (Akula et al., 2013; Bueno et al., 2018), building energy usage (Valero et al., 2021), and space utilization (Pan et al., 2022) are some of the numerous use cases of digital twins in the built environment. A built environment digital twin is commonly built from a Building Information Model (hereafter `BIM model'), which contains geometric and some semantics (such as element materials) that can be used to support the envisioned use cases (I. Giannakis et al., 2015).

The generation of BIM models of new buildings is done during the delivery process with an as-design BIM model created during the design phase that should then ideally be updated into an as-built BIM model that incorporates any change made during construction. But, BIM models also increasingly need to be created for existing buildings, for example to plan refurbishment or enhance operation.

In a 2020 survey conducted in 78 countries 2020, professionals asserted major benefits that BIM brings to the construction process from a geometric viewpoint (Rocha et al., 2021). However, the generation of BIM models is challenging due to the complexity and diversity of building geometry, and the possibly high levels of clutter existing in occupied buildings. New technologies, such as Terrestrial Laser Scanners (TLS) or photogrammetry (PG), now enable the acquisition of dense and accurate 3D geometric data, in the form of point clouds. Scan-to-BIM is the process to produce the as-built model of an asset from laser scanned or photogrammetric point clouds (Bassier & Vergauwen, 2020; Bosché et al., 2015a). It includes segmenting the data and generating a final semantically-rich 3D model (Rashdi et al., 2022). Despite the benefits afforded by those new sensing technologies, the generation of BIM models remains challenging due to the complexity and diversity of building geometry, and the possibly high levels of clutter existing in occupied buildings.

Although Scan-to-BIM is generally a manual process in current industrial practice, there is extensive research in academia and industry to develop automatic Scan-to-BIM algorithms (Nikoohemat et al., 2019; Thomson & Boehm, 2015; Valero et al., 2021). For example, a Scan-to-BIM solution based on deep learning is developed in (Perez-Perez et al., 2021) for semantic segmentation. It classifies beam, ceiling, column, floor, pipe, and wall elements using two convolutional neural network and one recurrent neural network.

Referee List (DOI: 10.36253/fup_referee_list)
FUP Best Practice in Scholarly Publishing (DOI 10.36253/fup_best_practice)
Shirin Malihi, Frederic Bosche, Martin Bueno Esposito, *Quantifying the Confidence in Models Outputted by Scan-To-BIM Processes,* pp. 1137-1146,
 DOI 10.36253/979-12-215-0289-3.113

Some researchers (Bassier & Vergauwen, 2020) have gone beyond the problem of object detection in scan-to-BIM, and presented results producing BIM models in IFC from semantic information extracted from point clouds (IFC is an open data schema described as in ISO 16739-1:2018 (BuildingSMART, n.d.)).

However, (Rocha et al., 2021) reported that Level of Accuracy was used in 9.2 % of the research reviewed from the literature, which shows a big room for development and using this concept. Guaranteeing the completeness and accuracy of BIM models generated through Scan-to-BIM processes (manual or automated) is an important issue. In previous research, authors have mainly done this manually. For example, in (Skrzypczak et al., 2022) the authors compare the lengths from total station measurements and the BIM model generated from Scan-to-BIM approach. But, comparisons like this are established to check quality manually for the purpose of academic assessment.

In practice, however, the user would need to know to what extent it can be confident that a scan-to-BIM algorithm has produced a correct model from the input point cloud data. Without any such information, the user will need to check every reconstructed element against the input data and gauge correctness manually, which is a time-consuming, and error-prone process that partially undoes the benefits afforded by automated scan-to-BIM algorithms.

Reducing this manual work could be achieved if the scan-to-BIM algorithm could also report some level of confidence for the modelling of each element in the outputted model. In this paper, we explore two such generic metrics (and a third one combining them) to automatically assess the quality of the model generated by a Scan-to-BIM algorithm, focusing on geometrical fitness.

The proposed indicators of the confidence are introduced in section 2. Section 3 then reports experimental results on their evaluation using some real case studies. Finally, the results are discussed and avenues for future work suggested in Section 3.

2 METHOD

This section presents the method proposed to calculate the level of confidence in BIM models outputted by Scan-to-BIM processes. Two different indices are defined $I_{coverage}$, $I_{distance}$, and *their combination* that can be computed for any element in the outputted model. They are detailed in the following sub-sections.

2.1 $I_{coverage}$

$I_{coverage}$ is the principal index and aims to capture how much of the modelled 3D surface of a given element in the outputted model is explained by the input point cloud data. One quantitative measure of this consists in homogeneously discretizing the element's surface and check if some points from the input point cloud lay in the neighbourhood and describe that discrete surface.

A practical way to implement this is to use space voxelization. First, for each modelled element, a voxelization is performed in its bounding box, with a resolution δ (e.g. $\delta = 2.5$cm). The set of voxels intersecting the element mesh is then found (we use the method described in (Open3D)); we call this set γ_m . Then, we identify the subset of voxels in γ_m that also contain points from the point cloud. Points are searched inside the voxels. The centre of each voxel is the base of this search. KD tree structure is used to partition this space and efficiently search the set of points falling within each voxel. We call this second set γ_c. We then define $I_{coverage}$ as:

$$I_{coverage} = \frac{|\gamma_c|}{|\gamma_m|} \tag{1}$$

where $|.|$ is the cardinality operator. $I_{coverage}$ takes values between 0 and 1, with 1 indicating that the entire surface of the element's mesh has matched scanned point in its vicinity, i.e. within δ distance.

2.2 $I_{distance}$

$I_{coverage}$ captures how much of the modelled surface is explained by a point cloud in a somewhat coarse way, and considers neither how closely the modelled surface matches the point cloud nor the local orientation of the points and the modelled surface. The metric $I_{distance}$ aims to complement $I_{coverage}$. For this, we take the set of point cloud that are in the voxels in γ_c (called n), calculate their closest (orthonormal) distance to the element mesh, and then compute $I_{distance}$ as follows:

$$I_{distance} = \frac{1.0\ .n_1 + 0.5\ .n_2 + 0.25\ .n_3 + 0.125\ .n_4 + 0.0625\ .n_5}{n} \quad (2)$$

where n_1 is the number of the n points that are within $(1/5)\delta$ distance to the mesh, n_2 is the number of the n points that are between $(1/5)\delta$ 1 and $(2/5)\delta$ distance to the mesh, n_3 is the number of the n points that are between $(2/5)\delta$ and $(3/5)\delta$ distance to the mesh, n_4 is the number of the n points that are between$(3/5)\delta$ and $(4/5)\delta$ distance to the mesh, and finally n_5 is the number of the n points that are between $(4/5)\delta$ and δ distance to the mesh.

Calculation of the distance between a point of the point cloud and a triangle of the mesh satisfying two conditions that are the projection of the point on the plane, formed by the triangle, should be located inside this triangle, and additionally the distance between the point and the mesh triangle should be less than the buffer size. Afterwards points inside this buffer are used to compute the $I_{distance}$ using the weighted average formula in Equation 2. $I_{distance}$ also takes values between 0 and 1, with 1 indicating that all the points are very close to the mesh (within $(1/5)\delta$ distance).

3 EXPERIMENTAL VALIDATION

The validation of the proposed method is conducted using an example Scan-to-BIM algorithm, but the method is applicable to the use of any other algorithm. The employed Scan-to-BIM solution was developed as part of the EU-funded Horizon2020 BIMERR project (Valero et al., 2021). The whole solution is semi-automatic and aimed at producing as-is models that contain as much information as possible to support the efficient (automated) development of an energy model that can be used to conduct simulations for refurbishment planning. This solution is divided into three components (Valero et al., 2021):

1. The *Structural Scan-to-BIM* component that automatically generates an IFC model containing the main architectural elements (floors, walls, openings, spaces) as well as second level space boundaries;
2. The *Mechanical, Electrical and Plumbing (MEP) Scan-to-BIM* component that automatically enriches the IFC model with elements such as radiators, HAVC units and sockets (Bosché et al., 2015b); and
3. *Scan-to-BIM Editor* to manually enrich the model with wall layers, materials, material properties and MEP properties.

The modelling confidence metrics proposed herein could be employed in each component, but are assessed here in the context of the first component of the solution, the Structural Scan-to-BIM component.

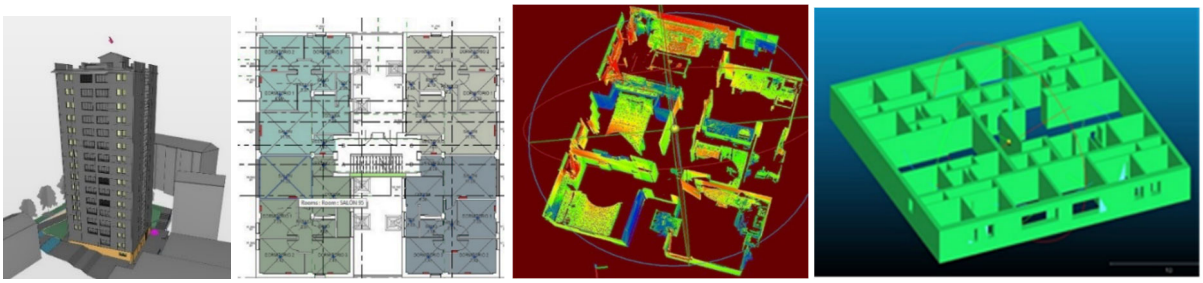
3.1 Experimental Data

One of the pilot sites of the BIMERR project is the two-story Kripis House located in Thessaloniki, Greece. A coloured point cloud of the whole house (exterior and interior) was captured using a terrestrial laser scanner Faro Focus 150s, and subsequently subsampled to a density of 1 pt/cm^2.

This house is furnished, however the interior is not much cluttered. The *Structural Scan-to-BIM* component automatically delivered a 3D model of the house in IFC format from that point cloud alone. A second pilot site is located in Bilbao, Spain. This is a multi-story apartment building, with essentially the same layout on all floors. Each floor contains four flats. A storey of the building was captured fully by terrestrial laser scanning. In contrast to the Kripis House, the Bilbao environment is cluttered because the building was scanned when fully inhabited. It contains wardrobes cupboards and other pieces of furniture and personal belongings inside rooms, resulting in significant levels of occlusion, which challenge Scan-to-BIM processes.

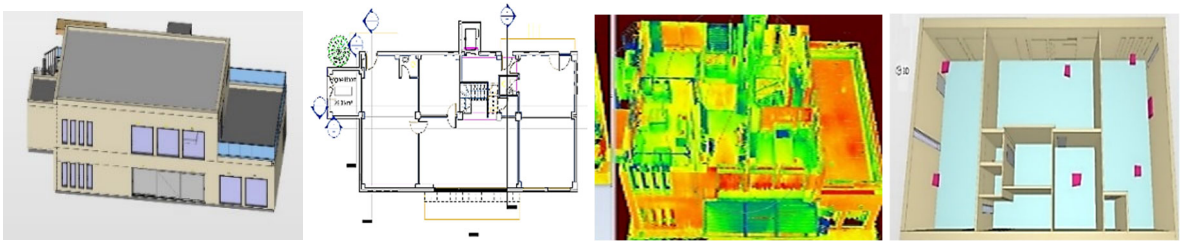
To assess the value of the proposed confidence metrics to report Scan-to-BIM confidence levels, manual scan-to-BIM was conducted by architects using standard commercial software and the resulting models exported in IFC. While those models may contain errors, they are generally good and can serve as ground truth. The validation then focuses on the walls, as walls are the most frequent elements in the models. The walls modelled manually and with the automated process are compared.

Figure 1 shows the Spanish dataset and the corresponding output IFC model generated by the *Scan-to-BIM* tool. Figure 2 shows the same information for the Greek dataset.



(a) (b) (c) (d)

Figure 1 : Spanish dataset and Scan-to-BIM output. The 3D model generated manually by the BIM modeller (a), plan view of one floor (b), the point cloud of one apartment (c), and 3D IFC model outputted by the Scan-to-BIM component for one floor (d).



(a) (b) (c) (d)

Figure 2: The 3D model generated manually by the BIM modeller (a), plan view of one floor (b), the point cloud (c), and 3D model outputted by the Scan-to-BIM (d). Stuff of indoor spaces are displayed in the point clouds.

3.2 Results and Discussion

First of all, we report on the overall wall detection performance of the Scan-to-BIM component. In an IFC model, correctly detected walls are counted as true positive (TP), non-existing detected walls are counted as false positive (FP), and missing walls are counted as false negative (FN).

Table 1 summarizes the performance obtained by the automated Scan-to-BIM algorithm. The results bring to light the challenges faced in the case of the Spanish dataset. While the automated Scan-to-BIM tool detected most walls (despite the clutter, noise and furniture of the rooms) but many walls modelled by the tool that actually do not exist, and four walls missed. In the case of the Greek project, 100% recall is achieved and 96% precision. Two FPs are reported, but these include a protruding beam confused as a wall and two columns modelled as a wall. These can in fact be considered acceptable because the automated Scan-to-BIM algorithm assumes that the structure of residential buildings is composed of walls and slabs only, and thus does not explicitly look for and model columns and beams.

Table 1: Wall detection performance of the Scan-to-BIM tool.

	TP	FP	FN	Recall	Precision
Spain dataset	60	14	4	94%	81%
Greece dataset	45	2	0	100%	96%

Figure 3 and Figure 4 show some of the modelling errors made by the automated algorithm with both datasets.

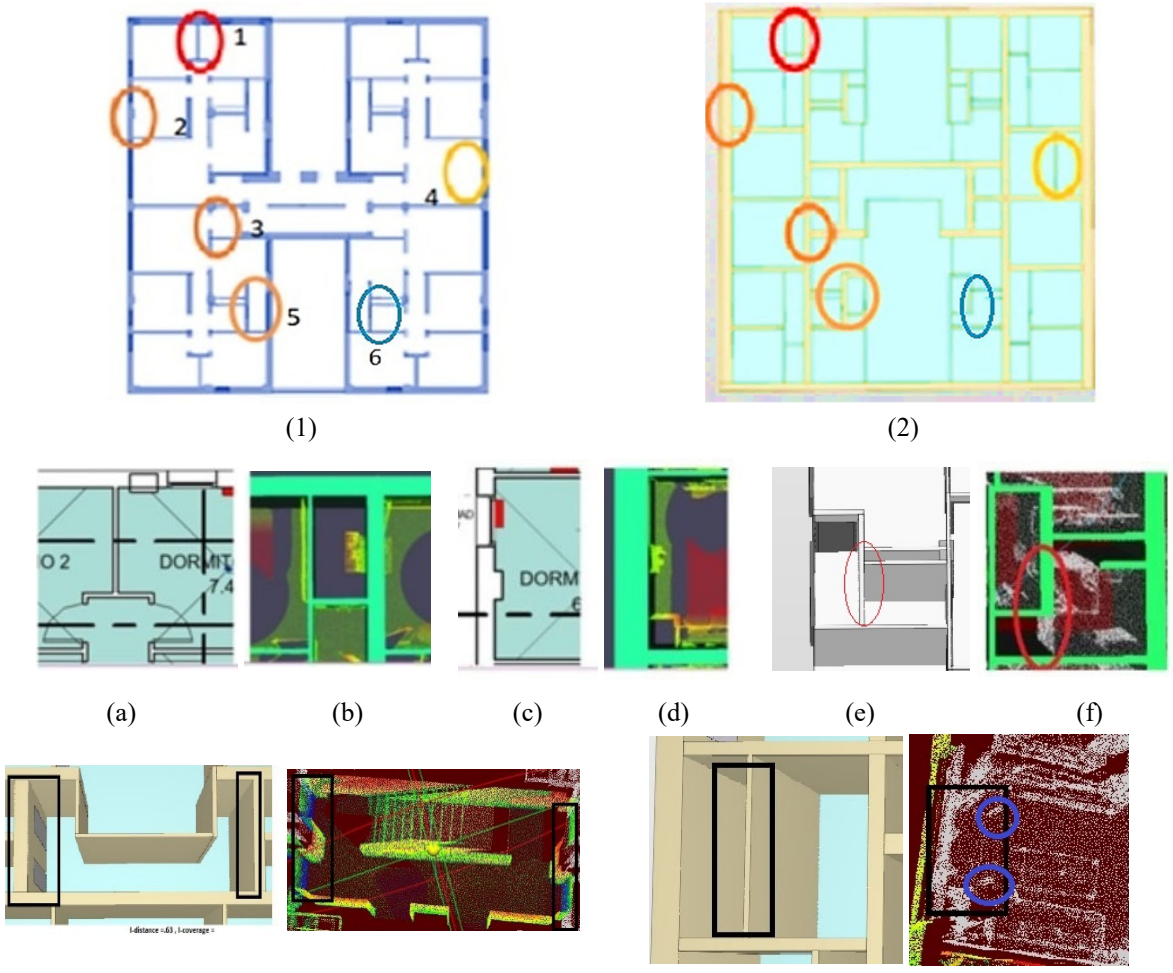
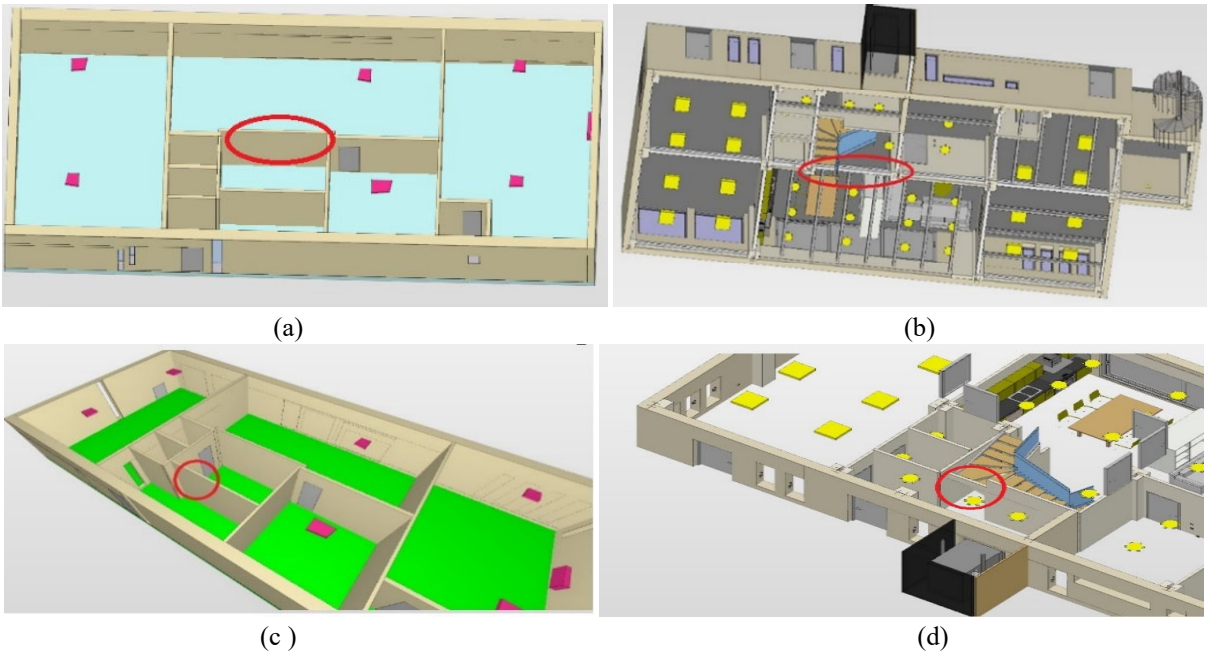


Figure 3: Modelling errors in the Spanish dataset. Errors in the manually-generated BIM model and the automated Scan-to-BIM algorithm are shown in (1) and (2) respectively. (a) and (b) show a wall which is modelled as a room with walls due to the presence of wardrobes inserted in the spaces on both sides. (c) and (d) depict a wall which is modelled as thick the column that is embedded in it. (e) and (f) demonstrate FN wall examples. (g) and (h) display an error in the modelling of thickness of walls. (i) and (j) show FP walls due to clutter in the room near the wall.

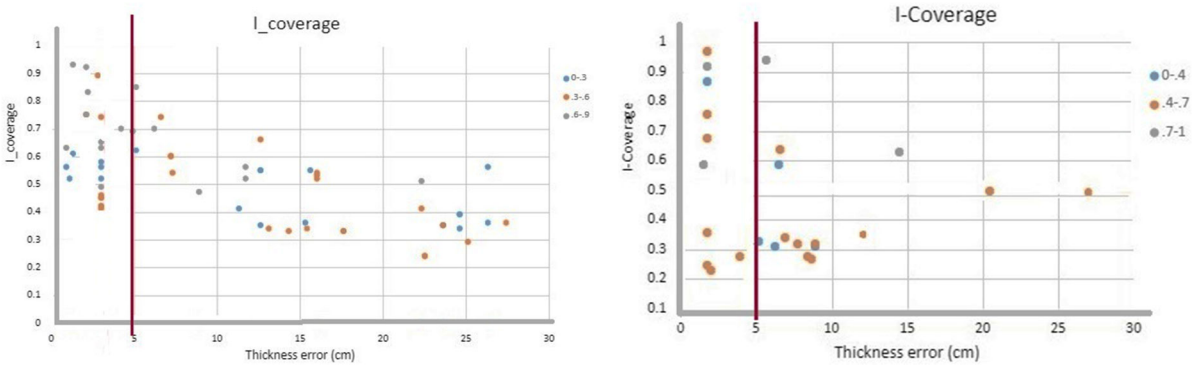


(a) (b)

(c) (d)

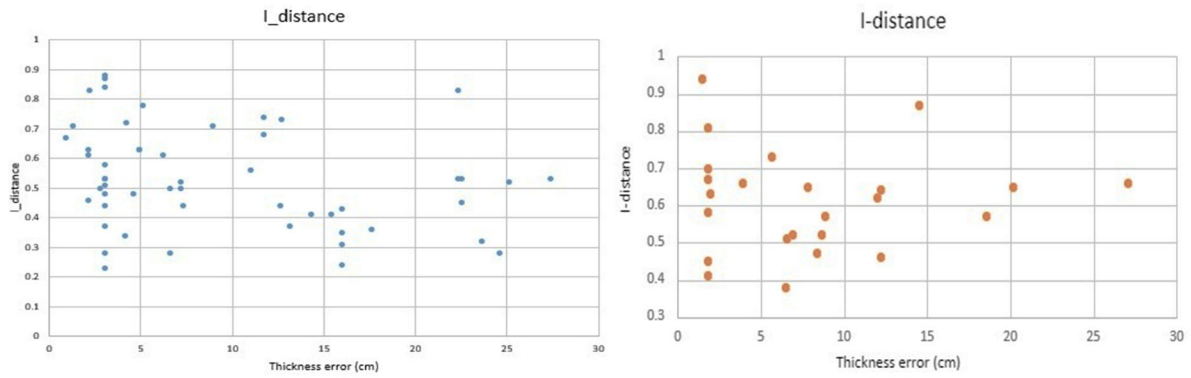
Figure 4: modelling errors in the Greek dataset. (a) and (c) show the model generated by the automated algorithm, while (b) and (d) show the model generated manually. In (a) and (b) a wall is modelled instead of two columns and a beam. (c) and (d) show a wall modelled instead of a beam.

Figure 5 reports the $I_{coverage}$ values obtained for each wall in the Spanish and Greek datasets. This figure plots the $I_{coverage}$ against the difference between the thicknesses of a given wall modelled manually (ground truth) and the same wall modelled automatically by the Scan-to-BIM algorithm. Note that in this experiment we use δ = 2.5 cm. The red vertical lines in Figure 5 are inserted on the distance of 2δ. This line is important because, if the wall is modelled at the right location but with a thickness error larger than 2δ, than the number of points within δ of each wall side, and a result the value of $I_{coverage}$ should be much lower. Figure 6 shows the $I_{distance}$ against the wall thickness error for the two datasets.



(a) (b)

Figure 5: $I_{coverage}$ and thickness errors for two datasets of Spanish (a) and Greek (b). They are coloured based on their corresponding $I_{distance}$ and split into three groups. The red vertical lines are inserted on the distance of 2δ.

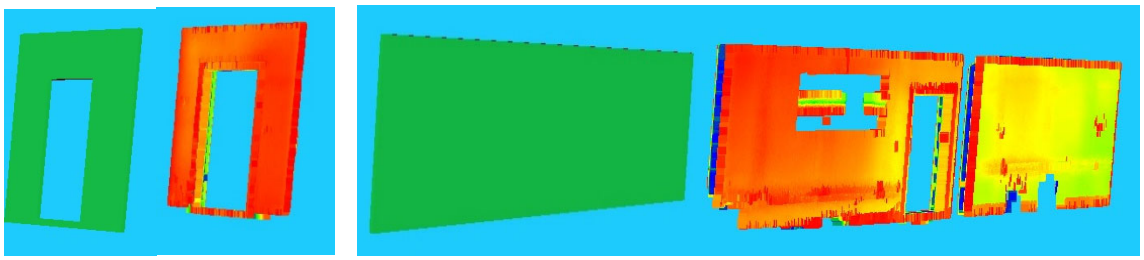


(a) (b)

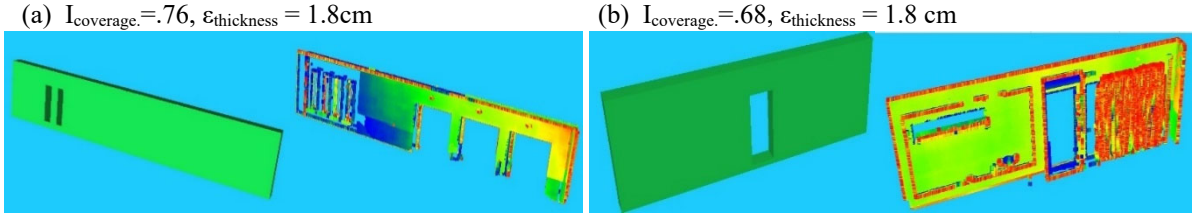
Figure 6: $I_{distance}$ and thickness errors for two datasets of Spanish (a) and Greek (b).

Looking at Figure 5, one can see a trend where $I_{coverage}$ is lower for walls with greater thickness error (in particular >2δ). This implies some correlation between $I_{coverage}$ and the confidence in the modelling quality. Nonetheless, some outliers do exist and good confidence seems to be only achieved for very high values of $I_{coverage}$ For $I_{distance}$ (Figure 6) a similar, but less positive trend can also be observed, as more obvious outliers can be observed.

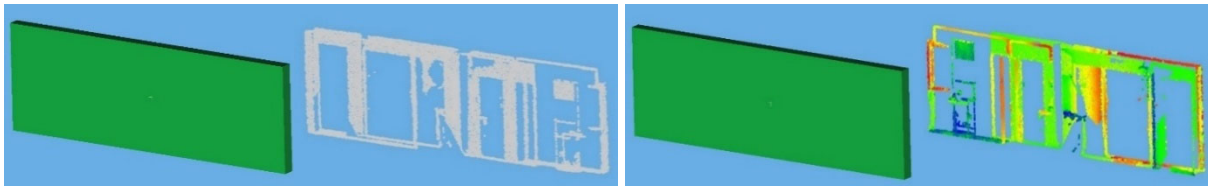
Figure 7 shows coverages and thickness errors of seven sample walls which were selected from different parts of Figure 5. (a, b) show examples where $I_{coverage}$ >0.5 and thickness error < 2δ. (c, d) show example for which $I_{coverage}$ is lower with values closer to 0.5, with one wall having thickness error >2δ and the other just below 2δ. The average value of $I_{coverage}$ in both cases are due to the fact that one of the faces of the walls is detected correctly, but the other one is wrongly detected. In the case of (c) this is due to a structural column (which the algorithm fails to detect) that leads to a gross over estimation of the width of the wall which subsequently leads to only very few points being matches to that face of the wall. The two large undetected windows also impact $I_{coverage}$. In (d), the second face of the wall is also wrongly modelled because the algorithm wrongly selected the curtains as the boundary for that wall. This results in lower (although not insignificant) thickness error which is still high enough to impact $I_{coverage}$. The conclusion is that, in both cases, $I_{coverage}$ rightly represents some level of error (for one face of the walls). (e, f) show walls for which $I_{coverage}$ <0.5 and thickness errors > 2 δ. These walls similarly have one face of the wall that is wrongly modelled, which is the source of the thickness error and implies that $I_{coverage}$ couldn't be higher than 0.5, as in the examples (c, d). But, (e, f) additionally contain undetected large windows and many occlusions due to desk, frame and mirror. Finally, (g) shows a wall that has a very low $I_{coverage}$ value (0.23) despite a small thickness modelling error. This may first appear to show a weakness of the proposed $I_{coverage}$ index. But, actually, in this case, while the wall is modelled with only 2cm thickness error, both faces of the wall are wrongly modelled and the wall end up looking like it was modelled at a location 4cm away from its true location. Therefore, $I_{coverage}$ rightly responds to this important modelling error.



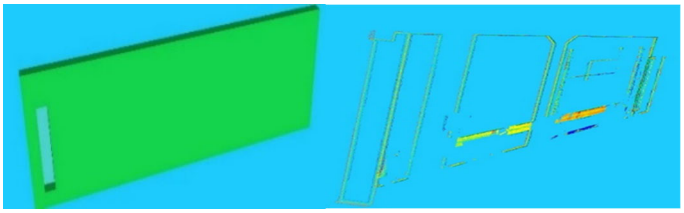
(a) $I_{coverage.}$=.76, $\varepsilon_{thickness}$ = 1.8cm (b) $I_{coverage.}$=.68, $\varepsilon_{thickness}$ = 1.8 cm



(c) $I_{coverage.}$=.51, $\varepsilon_{thickness}$ = 27.1cm (d) $I_{coverage.}$=.63, $\varepsilon_{thickness}$ = 4.5 cm



(e) $I_{coverage.}$=.41, $\varepsilon_{thickness}$ = 11.3 cm (f) $I_{coverage.}$=.35, $\varepsilon_{thickness}$ = 12.6 cm



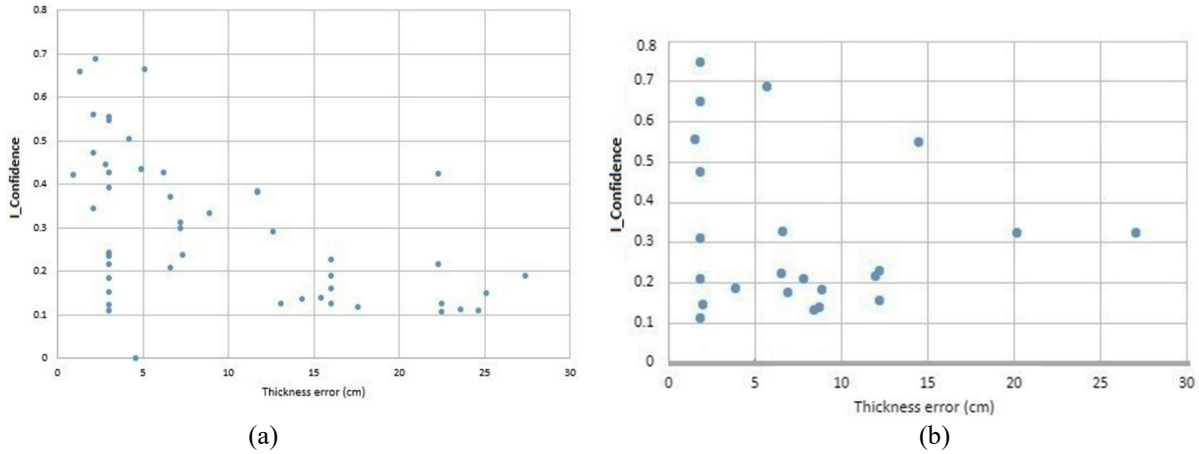
(g) $I_{coverage.}$=.23, $\varepsilon_{thickness}$ = 2.0 cm

Figure 7: $I_{coverage}$ and thickness error of sample walls

Overall, higher $I_{coverage}$, and to a lesser extent $I_{distance}$, show some level of correlation with smaller thickness modelling errors. However, in the presence of occlusion and obstruction from items such as cabinets, wall decorations, or curtains, $I_{coverage}$ is less reliable as a determinant factor for confidence of the automated Scan-to-BIM tool. To has achieve a higher correlation with the confidence of modelling, we can first look at combining these indices as:

$$I_{confidence} = I_{coverage} * I_{distance} \quad (3)$$

The results for $I_{confidence}$ are shown in Figure 8. This shows that $I_{confidence}$ has a slightly improved correlation with the confidence level of modelling.



(a) (b)

Figure 8: $I_{confidence}$ against the thickness errors for two datasets of Spanish (a) and Greek (b).

A second observation is that, since many of the modelling thickness errors arise from the presence of pieces of furniture or decoration in close proximity to the walls as well as the confusion of the algorithm between walls and columns, it is suggested, for future work to explore the use of some point cloud semantic segmentation algorithm (e.g. (Armeni et al., 2016)), which could provide further support during the modelling as well as to refine the confidence index by ensuring that wall elements are indeed modelled with points that are mostly labelled as being in the "wall" category.

4 CONCLUSION

Users of Scan-to-BIM algorithms and generally digital twins should be provided with reliable metrics of confidence for geometric modelling; hence they do not need to check everything manually to ease quality control and corrective works.

For this purpose, we introduce indices related to coverage and distance. These indices use information to analyse estimate confidence of the modelling tool. Coverage and distance information of the point cloud and IFC models are used to determine the consistency of the modelling. The major indicator is defined based on the coverage. Information of coverage provides local qualification of the modelling. The coverage index shows the best results, but, However due to obstruction, and the presence of furniture and decorative items in close proximity of or onto walls, only fairly high values of this index (>0.8) can be used to have high confidence in modelling. The distance index can be combined with to it and to improve its results, but still further work is necessary to improve the reliability of these indices. Semantic segmentation could be employed to detect different elements such as desk, mirror, frame, cupboard, as well as distinguish columns from walls, which would then be removed before modelling and/or accounted for in the calculation of a confidence index.

ACKNOWLEDGMENTS

This work was conducted as part of the BIMERR and COGITO projects that received funding from the European Commission's Horizon 2020 research and innovation programme under grant agreements No 820621 and 958310 respectively. For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising from this submission.

REFERENCES

Akula, M., Lipman, R. R., Franaszek, M., Saidi, K. S., Cheok, G. S., & Kamat, V. R. (2013). Real-time drill monitoring and control using building information models augmented with 3D imaging data. Automation in Construction, 36, 1–15. https://doi.org/https://doi.org/10.1016/j.autcon.2013.08.010

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., & Savarese, S. (n.d.). 3D Semantic Parsing of Large-Scale Indoor Spaces (a) Raw Point Cloud (b) Space Parsing and Alignment in Canonical 3D Space (c) Building Element Detection Enclosed Spaces. http://buildingparser.stanford.edu/

Bassier, M., & Vergauwen, M. (2020). Unsupervised reconstruction of Building Information Modeling wall objects from point cloud data. Automation in Construction, 120, 103338. https://doi.org/10.1016/J.AUTCON.2020.103338

Boje, C., Guerriero, A., Kubicki, S., & Rezgui, Y. (2020). Towards a semantic Construction Digital Twin: Directions for future research. Automation in Construction, 114(March), 103179. https://doi.org/10.1016/j.autcon.2020.103179

Bosché, F., Ahmed, M., Turkan, Y., Haas, C. T., & Haas, R. (2015a). 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. https://doi.org/10.1016/j.autcon.2014.05.014

Bosché, F., Ahmed, M., Turkan, Y., Haas, C. T., & Haas, R. (2015b). 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. https://doi.org/https://doi.org/10.1016/j.autcon.2014.05.014

Bueno, M., Bosché, F., González-Jorge, H., Martínez-Sánchez, J., & Arias, P. (2018). 4-Plane congruent sets for automatic registration of as-is 3D point clouds with 3D BIM models. Automation in Construction, 89, 120–134. https://doi.org/https://doi.org/10.1016/j.autcon.2018.01.014

BuildingSMART. (n.d.).

I. Giannakis, G., N. Lilis, G., Angel Garcia, M., D. Kontes, G., Valmaseda, C., & V. Rovas, D. (2015, December 7). A Methodology to Automatically Generate Geometry And Material Inputs for Energy Performance Simulation from Ifc Bim Models. https://doi.org/10.26868/25222708.2015.2363

Nikoohemat, S., Diakité, A., Zlatanova, S., & Vosselman, G. (2019). INDOOR 3D MODELING AND FLEXIBLE SPACE SUBDIVISION FROM POINT CLOUDS. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, IV-2/W5, 285–292. https://doi.org/10.5194/isprs-annals-IV-2-W5-285-2019

Open3D. (n.d.). No Title. http://www.open3d.org/docs/release/tutorial/geometry/voxelization.html

Pan, Y., Braun, A., Brilakis, I., & Borrmann, A. (2022). Enriching geometric digital twins of buildings with small objects by fusing laser scanning and AI-based image recognition. Automation in Construction, 140, 104375. https://doi.org/10.1016/J.AUTCON.2022.104375

Perez-Perez, Y., Golparvar-Fard, M., & El-Rayes, K. (2021). Scan2BIM-NET: Deep Learning Method for Segmentation of Point Clouds for Scan-to-BIM. Journal of Construction Engineering and Management, 147(9), 4021107. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002132

Rashdi, R., Martínez-Sánchez, J., Arias, P., & Qiu, Z. (2022). Scanning Technologies to Building Information Modelling: A Review. Infrastructures 2022, Vol. 7, Page 49, 7(4), 49. https://doi.org/10.3390/INFRASTRUCTURES7040049

Rocha, G., Mateus, L., Malinverni, S., & Pierdicca, R. (2021). A Survey of Scan-to-BIM Practices in the AEC Industry—A Quantitative Analysis. ISPRS International Journal of Geo-Information 2021, Vol. 10, Page 564, 10(8), 564. https://doi.org/10.3390/IJGI10080564

Skrzypczak, I., Oleniacz, G., Leśniak, A., Zima, K., Mrówczyńska, M., & Kazak, J. K. (2022). Scan-to-BIM method in construction: assessment of the 3D buildings model accuracy in terms inventory measurements. Https://Doi.Org/10.1080/09613218.2021.2011703, 50(8), 859–880. https://doi.org/10.1080/09613218.2021.2011703

Thomson, C., & Boehm, J. (2015). Automatic geometry generation from point clouds for BIM. Remote Sensing, 7(9), 11753–11775. https://doi.org/10.3390/rs70911753

Valero, E., Mohanty, D. D., Ceklarz, M., Tao, B., Bosche, F., Giannakis, G. I., Fenz, S., Katsigarakis, K., N. Lilis, G., Rovas, D., & Papanikolaou, A. (2021, November 2). An Integrated Scan-to-BIM Approach for Buildings Energy Performance Evaluation and Retrofitting. https://doi.org/10.22260/ISARC2021/0030