

Integrating Surface-related Indicators of Coverage, Distance and Distribution for Quantifying Scan-to-BIM Confidence Level

Shirin Malihi ¹, Frédéric Bosché ²

¹ Civil Engineering Department, University of Cambridge, United Kingdom - sm2852@cam.ac.uk

² School of Engineering, University of Edinburgh, United Kingdom - f.bosche@ed.ac.uk

Keywords: Scan-to-BIM, Modelling, Confidence, Coverage, Distribution, Distance, Voxelization

Abstract

Scan-to-BIM is a widely-used approach to generate Building Information Modelling and by extension Digital Twin models in the architecture, engineering, and construction sector. The resulting models need to be as accurate as possible to ensure subsequent activities that make use of them can do so effectively. Quality assessment of point clouds and occlusion assessment of BIM outputted from Scan-to-BIM has been investigated. However, Scan-to-BIM systems currently do not provide proper metrics as to the confidence the user can have in the quality, in particular geometric quality of the outputted model. This paper addresses this gap by introducing a confidence index, for analysing the reliability of the generated 3D models and thereby quantifying the confidence the user can have in them. Index of confidence is itself derived from three more specific indices: index of coverage estimates the portion of the surface of the modelled element that is explained by the input point cloud. Index of distribution estimates how well the points explaining the modelled surfaces are distributed around the overall object's surface. Index of distance defines the closeness of the generated element models to the input point cloud. The proposed indices are assessed using three real examples, demonstrating their adequacy.

1 Introduction

Digital twinning of built environment assets is a modern data-driven process with benefits to improve 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). Use cases of digital twins (DTs) in the built environment and construction sector include the monitoring and optimization of construction project execution (Deng et al., 2021), building energy usage (Zhao et al., 2021) and space utilization (Wang et al., 2022). A built environment DT is commonly built from a Building Information Model (hereafter 'BIM model'), that contains geometric and some semantics (such as element materials) that can be used to support the envisioned use cases (I. Giannakis et al., 2015).

In the context of developing BIM models from existing buildings (e.g. for refurbishment or renovation), best practice in industry follows a process commonly called Scan-to-BIM. This process uses as input the dense point clouds that can nowadays be obtained using 3D Laser Scanning (LS) or structure-from-motion (Pantoja et al., 2022).

The analysis of the point cloud data and modelling of the building asset are then commonly done manually, although significant efforts are currently put in automating these as much as possible. Scan-to-BIM point cloud analysis and modelling typically entails the steps of point cloud cleaning and denoising, possibly segmentation of the points into subsets that correspond to individual elements, and finally the modelling of those elements from those subsets (Rashdi et al., 2022). Despite

the benefits afforded by laserscanning and SFM technologies and the extensive research in academia and industry to develop automatic Scan-to-BIM algorithms (Thomson & Boehm, 2015), the generation of BIM models remains challenging due to the complexity and diversity of building geometry, and the common high levels of clutter existing in occupied buildings.

The resulting models need to be as accurate as possible to ensure subsequent activities that subsequently make use of them can do so effectively. Guaranteeing the completeness and accuracy of BIM models generated through Scan-to-BIM processes (manual or automated) is thus important. In previous research and current practice, people have mainly been doing this manually. For example, Skrzypczak et al. (2022) compare the lengths from total station measurements and the BIM model generated from Scan-to-BIM approach. But, comparisons like this one are established to check quality manually for the purpose of academic assessment and cannot be generalised for use in practice. In industry, users would usually visually check the adequacy of the model by overlapping it with the point cloud, sometimes by also colour-coding the point according to their distance to the model in order to get a sense of where any potential error may exist. But, this method is highly subjective, time-consuming (and therefore expensive) and prone to human error due to the quantity and variety of elements to check and the complexity of the scenes (especially when there is significant clutter).

Reducing this tedious, manual and fairly unreliable work could be achieved if some algorithms could jointly analyse the input data (point cloud) and output data (BIM model) and report some level of confidence for the modelling of each element in the outputted model such as the indicators which Malihi et al. presented (Malihi et al., 2023). Geometric quality assessment

of BIM outputted from Scan-to-BIM is investigated in (Bonduel et al. 2017) in two macro and micro scales. Occluded zones and non-modelled objects for complete building or its parts in macro scale and indication of the geometric accuracy of BIM objects in micro scale were analysed. Rebolj et al. 2017 addressed the relation between levels of point cloud quality and element identification for Scan-to-BIM.

In this paper, we explore one such algorithm that computes an index $I_{\text{confidence}}$ assessing the reliability of the modelling of each element generated by a Scan-to-BIM algorithm, focusing on geometrical fitness. $I_{\text{confidence}}$ is itself the result of combining three sub-indices: I_{coverage} , $I_{\text{distribution}}$ and I_{distance} that measure three distinct characteristics of how well the outputted model is explained by the input data.

The proposed indices 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. Three underpinning indices are defined I_{coverage} , $I_{\text{distribution}}$ and I_{distance} , alongside the overarching index $I_{\text{confidence}}$ that integrated them. These can be computed for any element in the outputted model.

2.1 Index of coverage

I_{coverage} 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 checking 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, which achieves good results for modelling and replicating real-world for digital twin. First, for each modelled element, a voxelization is performed in its bounding box, with a resolution of δ (e.g. $\delta = 2.5\text{cm}$) considering the resolution of the point cloud and the limit of surface boundary accuracy. The scale of modelling affects the level of details of the modelled surfaces. The size of bounding boxes and the resolution of voxelization are defined to evaluate the coverage of major elements of the BIM model.

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:

$$I_{\text{coverage}} = \frac{|\gamma_c|}{|\gamma_m|} \quad (1)$$

where $|\cdot|$ 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 Index of distribution

While I_{coverage} much how much of the surface is explained by the data, it fails to capture which parts of the element's surface are explained, for example, whether the surface parts explained are all on one side of the element or not. $I_{\text{distribution}}$ addresses this gap by examining the distribution of point cloud data around the element. This can be indirectly done by looking at the distribution of the points explaining the element's surface, according to the normal directions of the surface locally.

Number of points in 8 directions is computed by clustering normal vectors of voxelized point cloud and mesh. The voxels' normal vectors are intersected with the $X - Y$ plane and then grouped in 8 orientation ranges (each covering 45°). $I_{\text{distribution}}$ is then calculated according to the difference of distribution of the as-built and as-designed normals:

$$I_{\text{distribution}} = 1 - \alpha \sum_{i=1}^8 \left| \frac{D_i}{\sum_{i=1}^8 D_i} - \frac{B_i}{\sum_{i=1}^8 B_i} \right|, \quad \alpha = 1/2 \quad (2)$$

where D_i [resp. B_i] is the number of as-designed [resp. as-built] voxel normal vectors that fall in the orientation range i .

2.3 Index of distance

I_{coverage} and $I_{\text{distribution}}$ capture how much of and with what distribution around it the modelled surface is explained by the point cloud, but they do that in a somewhat coarse way, using voxelization. They thus do not capture how closely the modelled surface matches the point cloud. To fill this gap, I_{distance} takes the set of points that are in the voxels in γ_c , calculate their closest (orthonormal) distance to the element, and then compute I_{distance} as follows:

$$I_{\text{distance}} = \frac{1.0 \cdot n_1 + 0.5 \cdot n_2 + 0.25 \cdot n_3 + 0.125 \cdot n_4 + 0.0625 \cdot n_5}{n} \quad (3)$$

where n_1 is the number of the points in γ_c that are within $\left(\frac{1}{5}\right)\sigma$ distance to the mesh; n_2 is the number of points in γ_c that are between $\left(\frac{1}{5}\right)\sigma$ and $\left(\frac{2}{5}\right)\sigma$ distance to the mesh; n_3 is the number of points in γ_c that are between $\left(\frac{2}{5}\right)\sigma$ and $\left(\frac{3}{5}\right)\sigma$ distance to the mesh, n_4 is the number of points in γ_c that are between $\left(\frac{3}{5}\right)\sigma$ and $\left(\frac{4}{5}\right)\sigma$ distance to the mesh, and finally n_5 is the number of points in γ_c that are between $\left(\frac{4}{5}\right)\sigma$ and σ distance to the mesh. All points whose closest distance to the element is larger than σ are discarded.

Calculation of the distance of a cloud point to the element can easily be done by converting the element's surface geometry into a mesh. The calculation of the distance then requires the

calculation of the distance between it and every triangle of the mesh, which is done by projecting the point on the plane defined by the triangle from which the orthogonal distance is obtained. But, this distance is valid only if the projected point is located inside this triangle. Finally, the closest orthogonal distance of the point to the element is set as the shortest of all its distances to all the element mesh triangles. I_{distance} also takes values between 0 and 1, with 1 indicating that all the points are very close to the element (within $\left(\frac{1}{5}\right)\sigma$ distance).

3 Experimental Validation

3.1 Experimental Data and Setup

Two case studies are used for validating the proposed indices:

- *Kripis House*: A two-storey house located in Thessaloniki, Greece. The house is furnished, but the interior is not very cluttered. For this case study, 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/cm³.
- *Bilbao Building*: One of the floors of a multi-storey flat building in Bilbao, Spain. Each floor contains four flats. A story 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.

Two Scan-to-BIM processes were then applied to these datasets:

- *Manual Scan-to-BIM*: Modelling was done manually by a professional BIM modeller using standard commercial software. This method was applied to both the Kripis House data only.
- *Automated Scan-to-BIM*: Modelling was done automatically by the algorithm proposed by Valero et al., (2021) that generates an IFC model containing the main architectural elements (floors, walls, openings, spaces). This method was applied to both the Kripis House and Bilbao Building data.

Figure 1 gives an overview of the Kripis House dataset, while Figure 2 gives an overview of the Bilbao Building dataset.

It is important to highlight that the proposed indices are agnostic of the Scan-to-BIM method employed. The Scan-to-BIM methods employed in this paper are thus used only for illustrative purposes. However, they enable the comparison of the current industry best practice (i.e. manually by a professional BIM modeller) and some automated approaches (which are more likely to be defeated in more complex situations).

The validation reported in this paper focuses on the walls, as walls are the most frequent elements in the models. In order to assess the adequacy of the proposed indices to capture and quantify the confidence in modelling, a measure of wall modelling quality needs to be used and evaluated independently. With our focus on walls, we select wall thickness error as proxy to measure modelling quality where we have manually measured the thickness of each wall in the 3D point cloud data and computed the modelling error as the difference between the carefully manually measured thickness and that of each modelled wall (whether modelled using the Manual Scan-to-BIM or Automated Scan-to-BIM methods).

3.2 Results and Discussion

Figure 3, Figure 4 and Figure 5 report the results obtained for the three evaluation cases: Kripis House + Manual Modelling, Kripis House + Automated Modelling, and Bilbao Building + Automated Modelling. For each case, the figures contrast I_{coverage} , $I_{\text{distribution}}$ and I_{distance} obtained for all wall elements with the thickness modelling error for those walls. Since wall thickness error is used here as a proxy metric for wall modelling quality, this enables us to assess whether those indices are good predictors of modelling error or more exactly of the confidence in the modelling.

Note that in this experiment we use $\delta = 2.5$ cm, and as a result red vertical lines are inserted in the I_{coverage} graphs at the thickness error 2δ (i.e. 5 cm). These lines are important for the analysis for the following reason. Assuming the wall is modelled (approximately) at the right location but with a thickness error larger than 2δ , then the number of points within δ of each wall side, and as a result the value of I_{coverage} , should be much lower. In other words, we should see a drop in the I_{coverage} for values of thickness errors larger than 2δ . This appears to be the case in all three cases, although maybe not as sharply as expected. Notwithstanding, it can be noted that even for walls with small thickness modelling error, I_{coverage} have values that span a large range. This is because a wall may be modelled perfectly both if its surface is fully visible or if only a small part of its surface is visible (for example due to clutter). There is also the possible case that a wall is modelled with the right thickness but at the wrong location, in which case less point will fall with δ of the surface and thus I_{coverage} is low. Note that this behaviour of I_{coverage} is good, since it means it captures modelling error, and would be a good predictor of low confidence in modelling in that case.

Looking at $I_{\text{distribution}}$, it similarly appears that it generally increases with modelling quality. In other words, cloud points covering parts of the model with more various orientations tend to lead to better modelling outcomes (as expected), and $I_{\text{distribution}}$ seems to capture that well. This is particularly clear in the case of the Spanish Building + Automated Scan-to-BIM and Kripis House + Automated Scan-to-BIM experiments. For the Kripis House + Manual Scan-to-BIM experiment, the results are less obvious because the modelling error is in most cases lower than 4cm. Nonetheless some outliers appear in this

experiment (1 outlier) and in the Kripis House + Automated Scan-to-BIM experiment (3 outliers). These are discussed further below.

Table 1 shows, for seven example walls: the modelled wall, the point cloud used for the modelling, the wall thickness modelling error, and the computed indices. The example walls were selected for representing various combinations of index values and modelling quality, and are ordered according to their thickness modelling error. Walls (a) (b) and (c) may be considered to have a reasonable error ($\leq 2\text{cm}$), while the other ones have more problematic levels of errors, in particular (f), (g), (h) and (i) having significant errors.

Examples (a) and (b) show two walls with good modelling quality (at least in terms of thickness error) and reasonably high I_{coverage} values. The main difference however is that it can be seen that wall (b) is modelled with fewer points coming from one side of the wall, which should be highlighted to the user (for them to check) because it should lead to a reduced confidence in the modelling outcome. This difference is captured well by $I_{\text{distribution}}$ that is reasonably high for wall (a) but lower for wall (b). Wall (c) presents a similar case but with much lower $I_{\text{distribution}}$ which is due to very few points matched on the one side of the wall.

Wall (d) has indices with similar values as Wall (b) and the situation at first appears indeed similar. However, the modelling error of Wall (d) is much higher. The difference here is that the Scan-to-BIM method, which is Automated Scan-to-BIM in this case, mistakenly detected the wall boundary at the location of the drawn curtains. However, $I_{\text{distribution}}$ remains low which would rightly draw the attention of the user to this wall.

Finally, Walls (e), (f) and (g) show three examples where modelling is wrong due to one of the wall faces being wrong identities (cf. Walls (f) and (g)) alongside various levels of occlusion due to clutter.

Overall, lower I_{coverage} and $I_{\text{distribution}}$ and to a lesser extent I_{distance} values, show some level of correlation with low confidence in modelling outcome (i.e., the modelling could be either good or bad), while higher levels show some level of correlation with high confidence in modelling outcome. We have also seen that some walls with non-negligible modelling error may still score highly in one of the indices. Therefore, a logical conclusion is to develop an overarching index, $I_{\text{confidence}}$, that integrates all three indices in a way that it can only achieve a high value if all three indices have high values. As a first suggestion, we propose:

$$I_{\text{confidence}} = I_{\text{coverage}} * I_{\text{distance}} * I_{\text{distribution}} \quad (4)$$

The $I_{\text{confidence}}$ for all three experiments are shown in Figure 6 against the wall thickness modelling errors. They show that $I_{\text{confidence}}$ works nicely, with a clear trend where low $I_{\text{confidence}}$ values really correlate with high uncertainty (i.e. low

confidence) in modelling quality and in contrast, high $I_{\text{confidence}}$ values correlate with low uncertainty (i.e. high confidence) in modelling quality.

Nonetheless, further improvements may still be needed to enhance the metric's discriminatory performance. For example, it is observed that many of the modelling thickness errors, even for elements that still achieve reasonably good indices, 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. To reduce the influence of these points, future work could explore the use of some point cloud semantic segmentation algorithm, which could be used to refine the calculations of the indices by ensuring that wall elements are indeed modelled with points that are mostly labelled as being in the "wall" category.

Moreover, modelling errors of wall thickness was the only metric which is investigated in this research. Other error sources can interfere in the results for example occlusion and visibility of walls on each side of the walls, wrong location of walls, non-modelled elements. Hence, uncertainty of the proposed metric for the proposed indicators is affected by errors which can not be evaluated by this metric, and it needs further research which opens new areas in computing confidence of Scan-to-BIM.

4 Conclusion

Users of Scan-to-BIM algorithms should be provided with reliable metrics of confidence for geometric modelling, so that they do not need to check everything manually and corrective works can be eased. For this purpose, we introduce indices related to coverage, distribution and distance that can be used collectively to estimate confidence of the modelling outcome. The coverage and distribution indices appear to be the most critical, and the overall index $I_{\text{confidence}}$ shows value for use in practice.

However, due to obstruction, and the presence of furniture and decorative items in close proximity of or onto walls, which contribute to reducing the index values even in cases where the modelling is of good quality. To further enhance the discriminatory performance of the proposed index, 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 the confidence index.

Acknowledgments

This work was conducted as a part of the Bimerr and Cogito projects that received funding from the European Commission's Horizon 2020 research and innovation programme.

References

Boje, C., Guerriero, A., Kubicki, S., & Rezgui, Y., 2020: towards a semantic construction digital twin: directions for future research. *Automation in Construction*. <https://doi.org/10.1016/j.autcon.2020.103179>.

Bonduel, M., Bassier, M., Vergauwen, M., Pauwels, P., & Klein, R. 2017. Scan-to-BIM output validation: towards a standardized geometric quality assessment of building information models based on point clouds. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*.

Deng, M., Menassa, C., Kamat, V., 2021: from BIM to digital twins: a systematic review of the evolution of intelligent building representations in the AEC-FM industry, *Journal of Information Technology in Construction*.

I. Giannakis, G., N. Lilis, G., Angel Garcia, M., D. Kontes, G., Valmaseda, C., & V. Rovas, D., 2015. a methodology to automatically generate geometry and material inputs for energy performance simulation from IFC BIM models. *14th Conference of International Building Performance Simulation Association*. <https://doi.org/10.26868/25222708.2015.2363>.

Malihi, S., Bosche, F., Esposito, M., 2023. quantifying the confidence in models outputted by Scan-To-BIM processes, *CONVR*, pp. 1137-1146, DOI 10.36253/979-12-215-0289-3.113.

Pantoja-Rosero, B. G., Achanta, R., Kozinski, M., Fua, P., Perez-Cruz, F., & Beyer, K., 2022: generating LOD3 building models from structure-from-motion and semantic segmentation. *Automation in Construction*, 141, 104430.

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>.

Rebolj, D., Pučko, Z., Čuš Babič, N., Bizjak, M., Mongus, D., 2017: point cloud quality requirements for Scan-vs-BIM based automated construction progress monitoring, *Automation in Construction*.

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>.

Wang, T., Gan, V. J., Hu, D., & Liu, H., 2022: digital twin-enabled built environment sensing and monitoring through semantic enrichment of BIM with SensorML. *Automation in Construction*, 144, 104625.

Zhao, L., Zhang, H., Wang, Q., Qang, H., 2021: digital-twin-based evaluation of nearly zero-energy building for existing buildings based on Scan-to-BIM. *Advances in Civil Engineering*, 6638897.

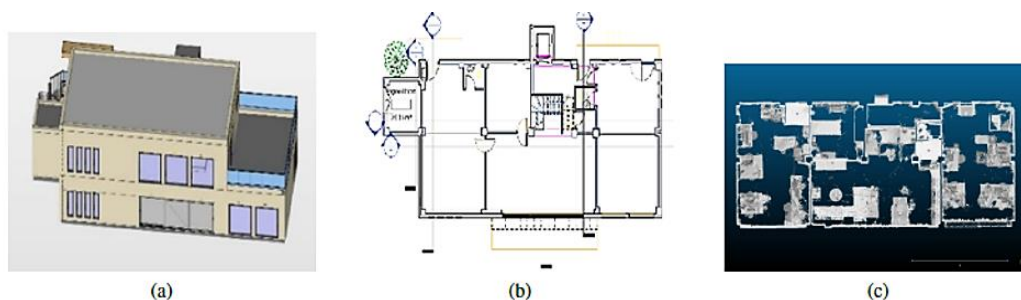


Figure 1. Kripis House dataset: (a) 3D model generated manually by the BIM modeller; (b) plan view of one floor; and (c) the point cloud of that floor.

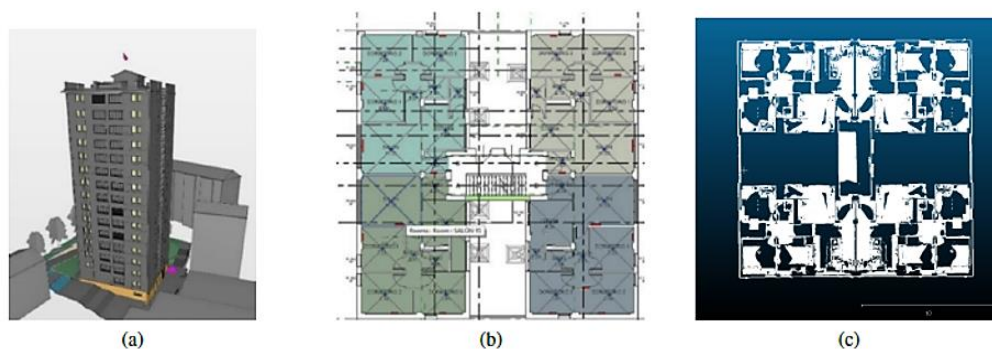


Figure 2. Bilbao Building dataset: (a) 3D model generated manually by the BIM modeler; (b) plan view of one floor; and (c) the point cloud of one apartment

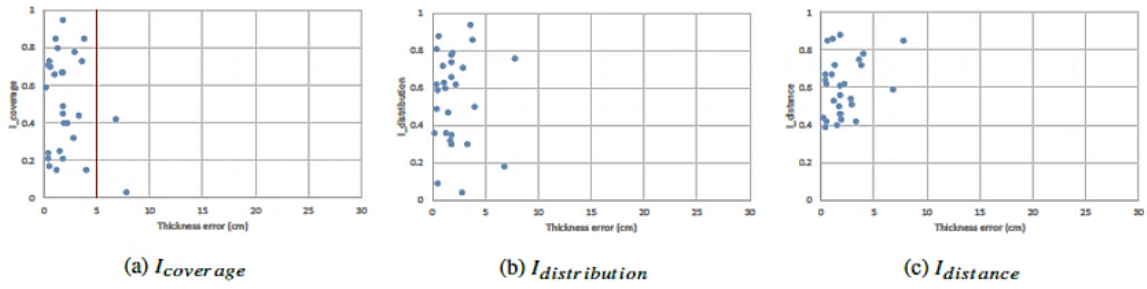


Figure 3. Confidence sub-indices for the Kripis House + Manual Scan-to-BIM experiment.

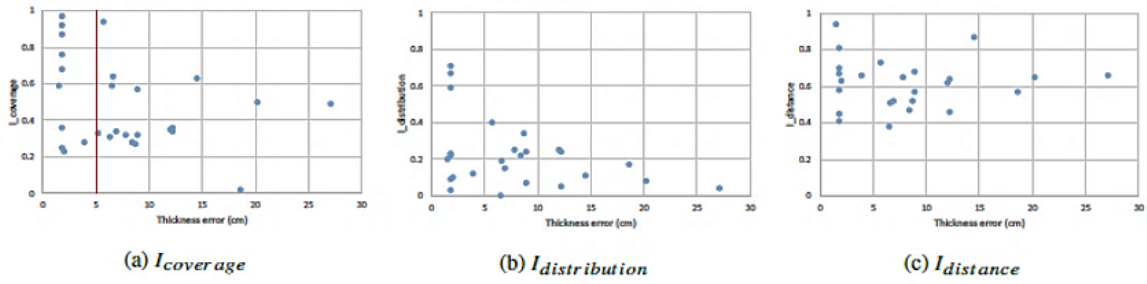


Figure 4. Confidence sub-indices for the Kripis House + Automated Scan-to-BIM experiment.

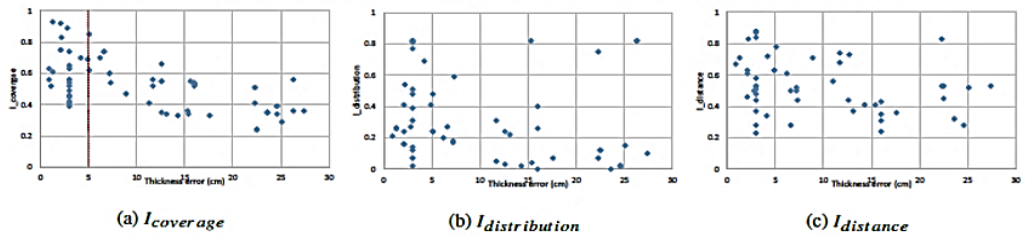


Figure 5. Confidence sub-indices for the Bilbao Building + Automated Scan-to-BIM experiment.

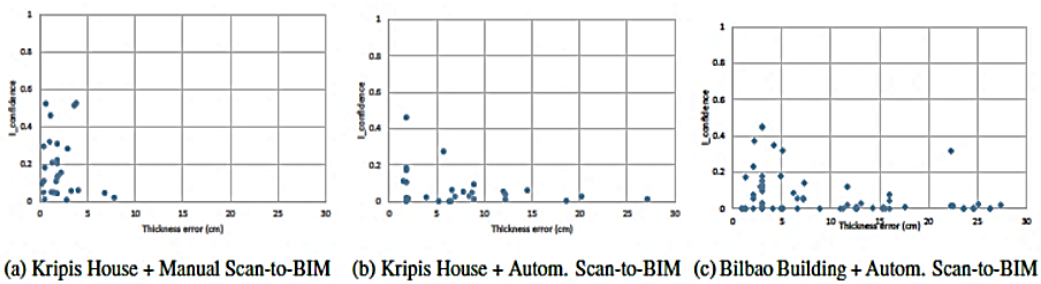


Figure 6. $I_{confidence}$ against wall modelling thickness errors for the three experiments.


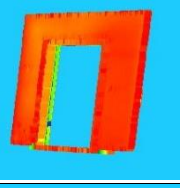

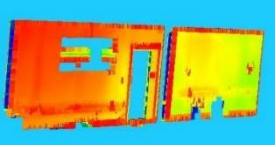
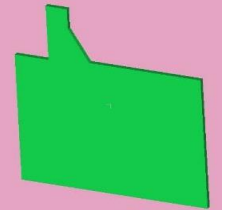


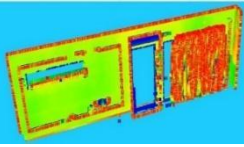


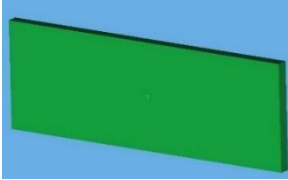
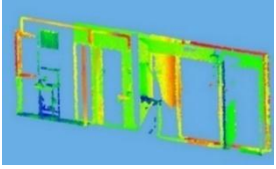
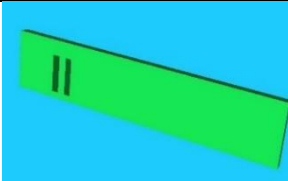
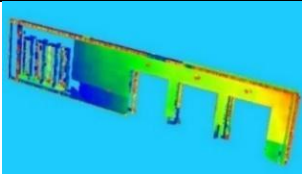
ID	Modelled wall	Point cloud	Thickness error (cm)	$I_{coverage}$; $I_{distribution}$; $I_{distance}$
a			1.8	0.76 0.80 0.41
b			1.8	0.68 0.61 0.70
c			3.8	0.78 0.22 0.72
d			4.5	0.63 0.55 0.87
e			6.8	0.42 0.87 0.59
f			12.6	0.35 0.50 0.32
g			27.1	0.51 0.52 0.66

Table 1: Illustrative examples of results showing, for each wall: modelled wall model, point cloud used for the modelling, the computed indices and thickness.