

# COMPARISON OF 2D AND 3D WALL RECONSTRUCTION ALGORITHMS FROM POINT CLOUD DATA FOR AS-BUILT BIM

SUBMITTED: August 2019

REVISED: January 2020

PUBLISHED: March 2020

EDITOR: Turk Ž.

DOI: [10.36680/j.itcon.2020.011](https://doi.org/10.36680/j.itcon.2020.011)

**Maarten Bassier, Dr.Ing.,**

Dept. of Civil Engineering, TC Construction – Geomatics, KU Leuven, Belgium  
[maarten.bassier@kuleuven.be](mailto:maarten.bassier@kuleuven.be) and <http://iiv.kuleuven.be/onderzoek/geomatics>

**Meisam Yousefzadeh, Dr.Ir.,**

Computer Vision Field Team, PTC, Boston, USA  
[myousefzadeh@ptc.com](mailto:myousefzadeh@ptc.com) and <https://www.ptc.com/>

**Maarten Vergauwen, Prof.Dr.Ir.,**

Dept. of Civil Engineering, TC Construction – Geomatics, KU Leuven, Belgium  
[maarten.vergauwen@kuleuven.be](mailto:maarten.vergauwen@kuleuven.be) and <http://iiv.kuleuven.be/onderzoek/geomatics>

**SUMMARY:** *As-built Building Information Models (BIMs) are becoming increasingly popular in the Architectural, Engineering, Construction, Owner and Operator (AECOO) industry. These models reflect the state of the building up to as-built conditions. The production of these models for existing buildings with no prior BIM includes the segmentation and classification of point cloud data and the reconstruction of the BIM objects. The automation of this process is a must since the manual Scan-to-BIM procedure is both time-consuming and error prone. However, the automated reconstruction from point cloud data is still ongoing research with both 2D and 3D approaches being proposed. There currently is a gap in the literature concerning the quality assessment of the created entities.*

*In this research, we present the empirical comparison of both strategies with respect to existing specifications. A 3D and a 2D reconstruction method are implemented and tested on a real life test case. The experiments focus on the reconstruction of the wall geometry from unstructured point clouds as it forms the basis of the model. Both presented approaches are unsupervised methods that segment, classify and create generic wall elements. The first method operates on the 3D point cloud itself and consists of a general approach for the segmentation and classification and a class-specific reconstruction algorithm for the wall geometry. The point cloud is first segmented into planar clusters, after which a Random Forests classifier is used with geometric and contextual features for the semantic labelling. The final wall geometry is created based on the 3D point clusters representing the walls. The second method is an efficient Manhattan-world scene reconstruction algorithm that simultaneously segments and classifies the point cloud based on point feature histograms. The wall reconstruction is considered an instance of image segmentation by representing the data as 2D raster images. Both methods have promising results towards the reconstruction of wall geometry of multi-story buildings. The experiments report that over 80% of the walls were correctly segmented by both methods. Furthermore, the reconstructed geometry is conform Level-of-Accuracy 20 for 88% of the data by the first method and for 55% by the second method despite the Manhattan-world scene assumption. The empirical comparison showcases the fundamental differences in both strategies and will support the further development of these methods.*

**KEYWORDS:** *Scan-to-BIM, Wall Reconstruction, As-Built, Modelling, Point cloud, Building.*

**REFERENCE:** *Maarten Bassier, Meisam Yousefzadeh, Maarten Vergauwen (2020). Comparison of 2D and 3D wall reconstruction algorithms from point cloud data for as-built BIM. Journal of Information Technology in Construction (ITcon), Vol. 25, pg. 173-192, DOI: [10.36680/j.itcon.2020.011](https://doi.org/10.36680/j.itcon.2020.011)*

**COPYRIGHT:** © 2020 The author(s). This is an open access article distributed under the terms of the Creative Commons Attribution 4.0 International (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



## 1. INTRODUCTION

The Architectural, Engineering, Construction, Owner and Operator (AECOO) industry is increasingly adopting Building Information Modelling technology into their workflows. Central to this innovation is a semantic three dimensional data model known as a Building Information Model (BIM). These intelligent databases are capable of storing an immense amount of information about the structure at the varying stages of the asset's life cycle. They provide professionals in the industry with insights and tools to plan, design, construct, and manage buildings and infrastructure more efficiently (Ilter et al., 2015). The models also facilitate the exchange of information between the varying stakeholders in a project and are used in numerous applications. For instance, BIM models are commonly employed in project design, facility management and energy performance (U.S. General Services Administration, 2011)

While BIM is typically engaged for new structures, the industry also looks to implement the technology for existing buildings for the purpose of heritage documentation, maintenance, quality control, and so on (Agarwal et al., 2018; Volk et al., 2014). These applications require the BIM model to reflect the state of the facility according to as-built conditions. The creation of as-built and as-design BIM models face different challenges. For instance, the information about built structures is often sparse or non-existing. Moreover, the existing documentation often does not match the as-design model of the building due to construction changes or renovations (Brilakis et al., 2010; Patraucean et al., 2015; Tang et al., 2010). The production of as-built BIM models involves the acquisition of the geometry of the structure and the reconstruction of the BIM model based on these geospatial measurements.

Currently, this Scan-to-BIM process is a manual process performed by expert modelers that interpret the point cloud and manually design all the relevant objects in the scene. This is a time-consuming procedure and thus automated algorithms are presented to speed up the process. The presented methods are divided into 2D and 3D methods. 2D methods evaluate planar subsets of the observations and trade severe abstractions for computational efficiency. In contrast, 3D methods evaluate the scene in its entirety which can be computationally demanding. Overall, there is gap in the literature that compares both strategies in terms of accuracy and reliability.

The aim of this research is therefore to provide a comparison between 2D and 3D automated approaches for the creation of as-built BIM models. More specifically, the reconstruction of BIM wall elements of existing structures is evaluated as it forms the basis for further reconstruction of the BIM. Typically, these objects are heavily cluttered and have problems with accessibility and occlusions (Fig.1). Both methods process unstructured point cloud data and output generic BIM elements. This research is focused on the processing of metric information as it forms the basis for further applications. Both the advantages and disadvantages of the approaches are discussed. Additionally, the outputs are compared to ground truth to identify the metric accuracy of the reconstruction. We use the Level of Development (LOD) (BIMforum, 2015) and the Level of Accuracy (LOA) (USIBD, 2016) specification as the framework for the comparison.

The remainder of this work is structured as follows. Section 2 presents the context of the work. In section 3 the related work is discussed. In Section 4 the methodology is presented. The test design and experimental results are proposed in Section 5. Both approaches are compared in Section 6. Finally, the conclusions are presented in Section 7.

## 2. BACKGROUND

Data acquisition technology has dramatically improved over the last decade. The innovations in laser scanning and photogrammetry give access to highly accurate dense point cloud data. A common technique used in building surveying is Terrestrial Laser Scanning (TLS) (Tang et al., 2010; Bassier et al., 2016). This technique is capable of capturing millions of 3D points in the scene in a very short period of time. By setting up the system on multiple locations in a structure, a number of scans is acquired. After aligning the data, a uniform point cloud is created. The result is a set of Cartesian points with high accuracy and resolution in a local or global coordinate system. This information can also be acquired with other systems such as indoor and outdoor Mobile Mapping systems (MMS). These techniques continuously capture information while moving through the scene. While this reduces the data acquisition time significantly, the data quality suffers from the continuous movement of the system (Thomson et al., 2013; Bosché F., 2012; Zlot et al., 2013). Photogrammetric approaches are being explored as well but generally require additional geometric information (Furukawa et al., 2009).



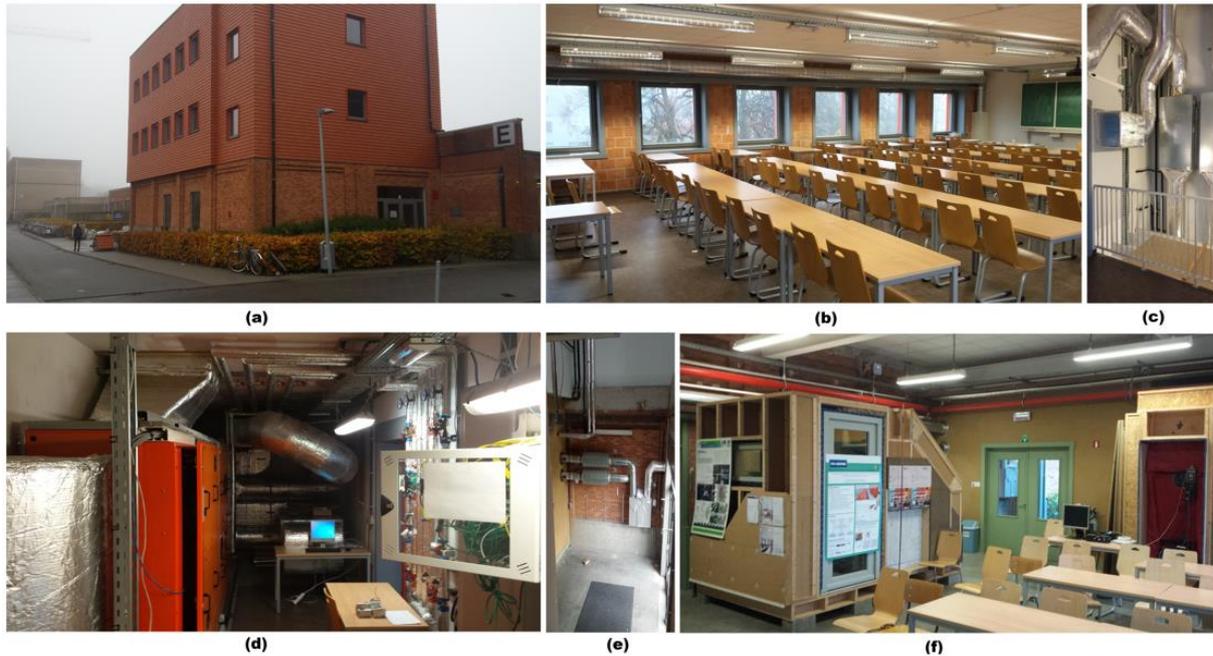


FIG. 1: Overview to the type of data that is processed by this research: (a) Overview building, (b) classrooms on 1<sup>st</sup> and 2<sup>nd</sup> floor, (c) hallway top, (d) maintenance room, (e) hallway bottom and (f) laboratory.

The acquired point cloud serves as a basis for the as-built BIM reconstruction, which is referred to as Scan-to-BIM. Currently, this process is mainly a labor intensive manual procedure. Therefore, the industry would greatly benefit from automation in Scan-to-BIM. Both 2D and 3D approaches are presented. The former reduces the initial point cloud to a set of 2D raster images by slicing the data or by projecting the points on a plane. This allows the use of efficient computer vision techniques for the segmentation and classification of the data. The latter processes the entire point cloud which is more time consuming but does not suffer from the abstractions of the 2D approaches.

The automated workflow can be divided into several phases (Fig.2). The order of the steps vary between approaches and some steps can be performed simultaneously. In general, the following steps are proposed: First, the point cloud is segmented for data reduction and interpretation. This is considered a well-understood problem in both 2D and 3D data structures and can be solved with a wide array of techniques including region growing, model-based methods, edge detection, graph based models, and so on (Hichri et al., 2013; Nguyen et al., 2013; Grilli et al., 2017). Typically, planar clusters in 3D and linear segments in 2D are considered as they best represent most structural elements (Barnea et al., 2013). Next, the data is classified. Each extracted cluster or segment is processed by a reasoning framework that computes class labels such as walls, floors and ceilings for each observation. The input data is described by a set of numeric values that encode its characteristics. Typically, these features represent both distinct geometric and contextual information (Anand et al., 2012; Weinmann et al., 2015). The classification algorithms that process the feature vectors use either heuristics or more complex classification methods such as Random Forests (RF), Neural Networks (NN), Probabilistic Graphical Models (PGM), Support Vector Machines (SVM), etc. (Xiong et al., 2013; Wolf et al., 2015). Finally, the labelled data set is used as a basis for class-specific reconstruction algorithms that create the correct BIM geometry for each class. This includes the associative clustering of the labelled points or segments to encompass a single object. Also, the individual components of the object are identified e.g. which clusters compose a single wall face of a wall. Once the input data is interpreted, the necessary metric information is extracted to compute the object's position, orientation and dimensions. Additionally, non-metric information can be extracted to texturize the object. The reconstructed walls can be further processed to form a watertight structure, merge similar wall types, etc. In this research, only the initial reconstruction is presented as the goal of our research is to compare the metric accuracy and reliability of 2D and 3D methods to the initial point cloud.

### 3. RELATED WORK

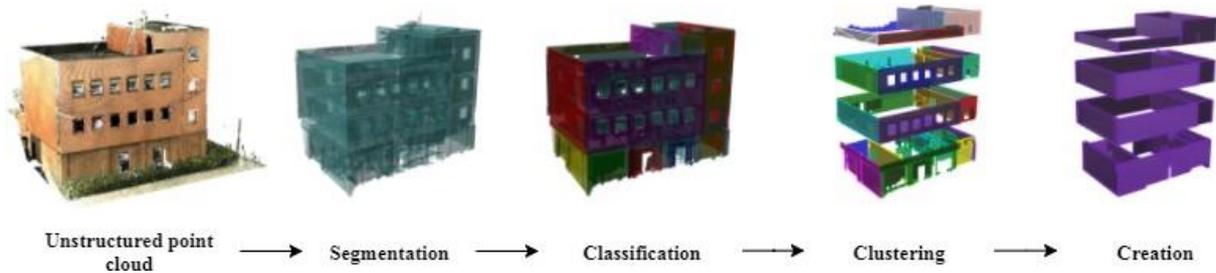


FIG. 2: Illustration of the general workflow of the automated Scan-to-BIM creation from unstructured point cloud data.

The segmentation, classification and reconstruction of point cloud data has been widely researched in the last decade. Several researchers have proposed complete 2D and 3D scan-to-BIM algorithms. A key aspect in 2D workflows is the segmentation of the stories of a building in order to create 2D representations of the data. A common approach is the use of histograms to vertically segment the data (Budroni et al., 2009; Budroni et al., 2010; Adan et al., 2011; Valero et al., 2012; Oesau et al., 2014; Macher et al., 2015). We propose a similar method as it correctly applies to most structures. The raster images derived from the segmentation can encode different types of information. For instance, Hong et al. (Hong et al., 2015) and Valero et al. (Valero et al., 2012) represent the floor and ceiling points in a 2D raster image in order to segment the wall points. Also floor plans themselves without point clouds have been represented as raster images (Gimenez et al., 2015). However, as this information is often unreliable, our approach does not include plan information. In this research, the use of all the points exempt for the ceilings and floors is proposed for the segmentation since this approach does not rely on the presence of floor or ceiling geometry but solely on the presence of the walls themselves.

Several methods have been proposed to detect wall points in 2D images including model-based methods, Hough transforms, edge detection algorithms, etc. (Schnabel et al., 2007; Lari et al., 2014; Previtali et al., 2014; Jung et al., 2014; Hong et al., 2015). Most approaches provide promising results in the case of clean data. However, they often only use a single slice of the story for the segmentation which makes the data prone to abstractions. Our model-based line segmentation implementation extends these approaches by considering all points in a story. Additionally, we present a reasoning framework to cluster the extracted segments before segmenting the wall points. Several 3D methods are proposed for the segmentation of wall geometry. Most methods are a 3D extension of their 2D counterparts. For instance, RANSAC and Hough transforms can be extended to 3D point cloud segmentation (Jung et al., 2014; Hulik et al., 2014). Other popular approaches include region growing, graph-based methods, histogram segmentation techniques, and so on (Turner et al., 2013; Richter et al., 2013; Dimitrov et al., 2015; Su et al., 2016; Son et al., 2015; Armeni et al., 2016). In this research a model-based method is employed for the 3D plane extraction.

3D workflows typically incorporate a classification step to identify the class of each observation. Several researchers have proposed methods employing heuristics to label building environments (Savaresse et al., 2016; Bassier et al., 2016; Rusu et al., 2009; Pu et al., 2011). Other approaches employ machine learning techniques such as neural networks (Wang et al., 2015), graphical models (Mnih et al., 2012; Niemeyer et al., 2014; Xiong et al., 2013), non-linear classifiers (Weinmann et al., 2015; Anand et al., 2012) or a combination of classification methods (Armeni et al., 2016). Heuristics have a computational advantage over more advanced techniques as they do not require intensive learning or large data sets in order to compute the class of an observation. However, heuristics are inherently biased and are often case specific. In this research we employ Support Vector Machines presented in previous work (Bassier et al., 2016) for the classification of the extracted planes similar to Anand et al. (Anand et al., 2012) and Yang et al. (Yang et al., 2013).

Several researchers consider the wall reconstruction as a room boundary optimization problem (Kabolizade et al., 2012; Mura et al., 2014). Turner and Zakhor presented a 2D triangulation method to segment rooms (Turner et al., 2014). They create the wall geometry based on the computed room boundary. Ochmann et al. proposed a planar graph cut method for room boundary extraction (Ochmann et al., 2016). The focus of these methods differs from our approach as they look to create consistent watertight rooms opposed to reconstructing accurate wall geometry.

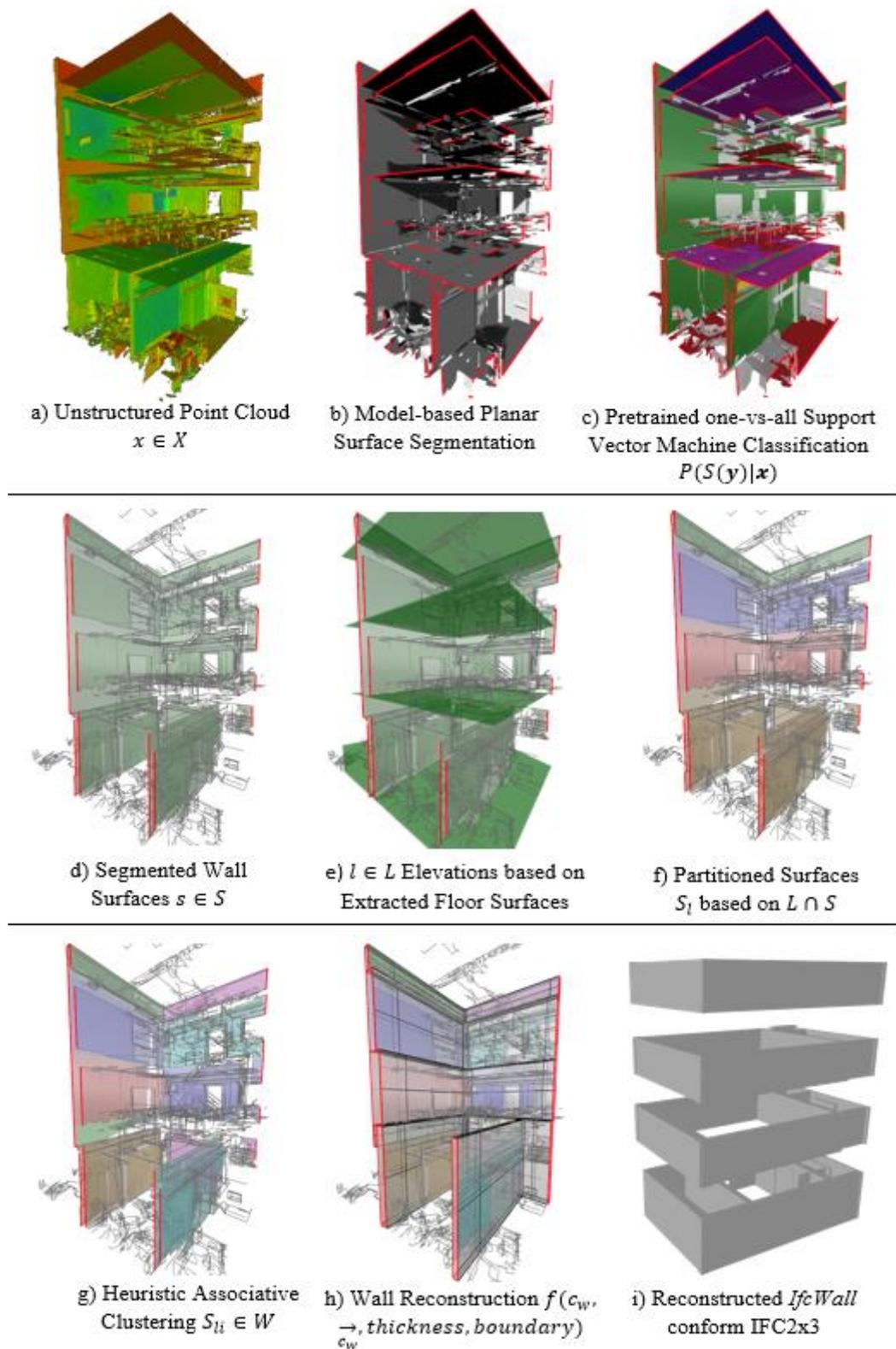


FIG. 3: Overview workflow of automated 3D method from point cloud to *IfcWall* geometry on multi-story school facility.

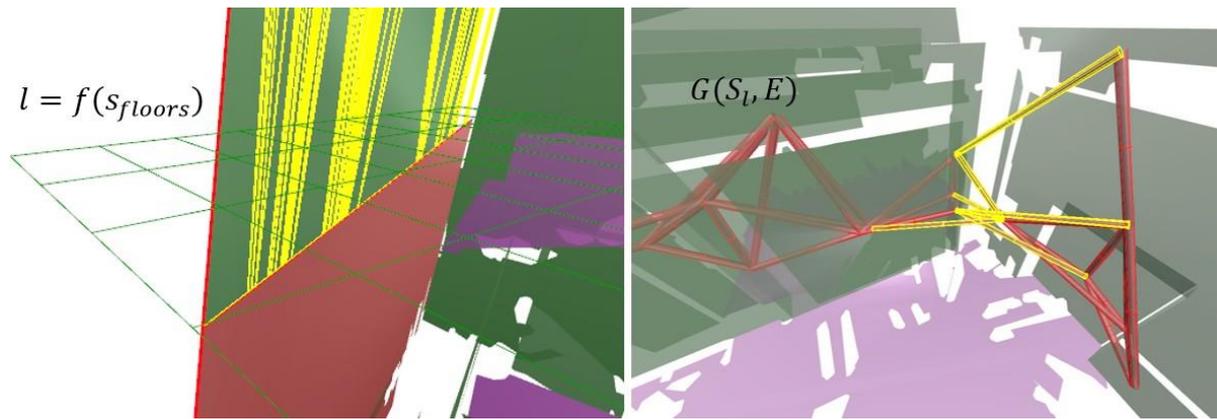


FIG. 4: Overview segmentation of meshes per story given reference plane  $l$  (left) and graph representation of adjacent surfaces in red with non-valid connections in yellow (right).

Only few researchers have proposed a metric quality assessment between the reconstructed geometry and the initial data. Thomson and Boehm performed a metric comparison between automatically reconstructed wall geometry by commercial software and a human-generated model (Thomson et al., 2015). Hulik et al. compared their detected planes to the initial data for single scans (Hulik et al., 2014). Previtali et al. compared their final model with the initial point cloud data using the extracted edges (Previtali et al., 2014). The comparison of both 2D and 3D reconstruction methods to ground truth is a frequently overlooked part in the literature. Researchers that do provide comparison data, typically present this information regardless of existing standards or specifications. In our method, we overcome this issue by comparing the reconstructed objects conform the Level of Development (LOD) (BIMForum, 2015) to the point cloud itself with reference to the U.S. Level of Accuracy specification (LOA) (USIBD, 2016).

## 4. METHODOLOGY

Both 2D and 3D scan-to-BIM algorithms are capable of reconstruction wall geometry from point cloud data. In this section, we present our implementation of both these types of algorithms with their advantages and disadvantages. The methods are designed to process generic multi-story industry buildings. Sections 4.1 and 4.2 respectively elaborate on the 3D and 2D approach in detail.

### 4.1 3D approach

The 3D method operates on the point cloud itself and processes the entire data set at once. While this can be computationally challenging, it also allows for better point cloud interpretation. Also, this method can deal with more complex environments as it does not rely on Manhattan-world assumptions such as wall verticality. A detailed overview of the steps and intermediate results is shown in Fig. 3. The following paragraphs discuss the four consecutive algorithms for the segmentation, classification, clustering and reconstruction.

#### 4.1.1 Segmentation

3D segmentation of point clouds can be performed with variety of techniques. In this work, we employ the planar model-based segmentation software Pointfuse developed by Arithmetica (Arithmetica Inc., 2015). This software incrementally fits planar surfaces through the point cloud and isolates those with sufficient support. After the initial segmentation, the 3D points are replaced by planar surfaces for data reduction. The boundary of the resulting planar surfaces is computed by an intersection algorithm that replaces the initial boundaries with the detected intersection lines of nearby other surfaces. Boundaries that do not intersect with other surfaces are abstracted to compose of the fewest linear segments as possible. The result of the segmentation is a set of planar surfaces  $s \in S$  that form a near-watertight mesh representation of the 3D point cloud as shown in Fig. 3(b).

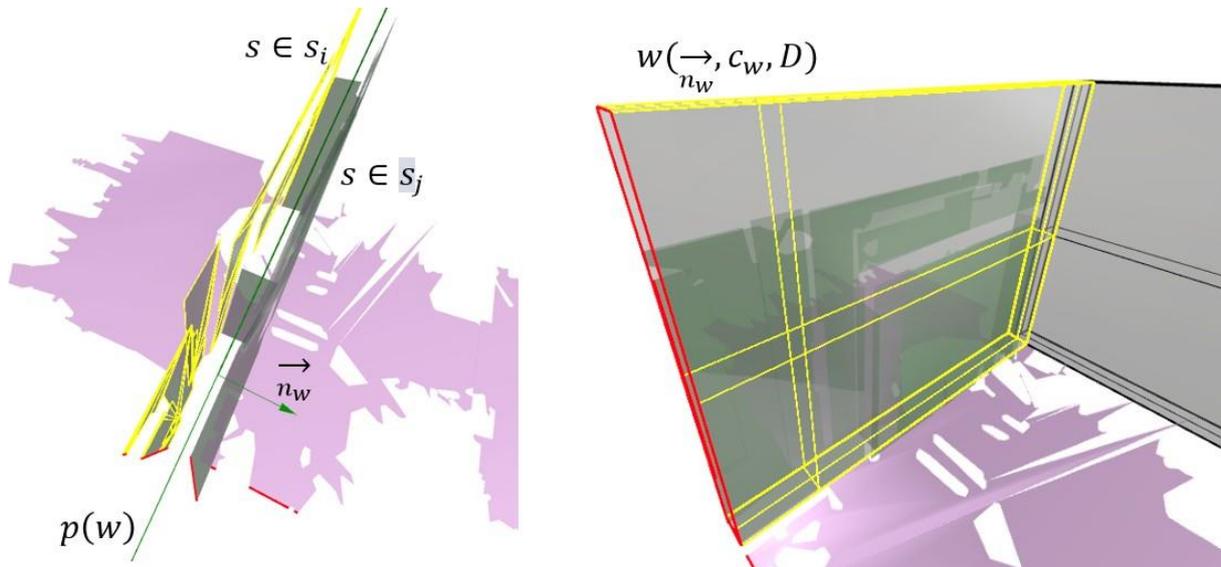


FIG. 5: Conceptual illustration of the wall faces given  $p(w)$  (left) and the final positioning of the wall with respect to the centre plane (right).

#### 4.1.2 Classification

Once the data is segmented, class labels are computed for each observed surface. The available classes in this research include walls, floors, ceilings, roofs, beams and clutter. The classification is performed using machine learning techniques. More specifically, we propose the use of Support Vector Machines (SVM) as discussed in previous work (Bassier et al., 2017). First, distinct geometric and contextual features  $x = \{x_1, x_2, \dots, x_n\}$  are extracted from each observed surface. These feature vectors are then processed by a set of pre-trained SVM's to compute the probability  $P(S(y)|x)$  of the class labels given the feature values. The fundamentally binary SVMs are combined in a one-vs all configuration to form a discriminative model. The final class labels are then computed based on the majority vote over the combined SVMs. The result of the classification is a set of labelled surfaces  $S(y)$  as shown in Fig. 3(c).

#### 4.1.3 Clustering

In this step, the individual wall segments (Fig.3(d)) are associatively clustered to form the basis for the final objects. In this research we propose the clustering of the geometry per wall object and per story. The partitioning per story is based on common practice in the manual modelling procedure. Additionally, walls can have deviating thicknesses on different stories and thus should be split during reconstruction. The floor surfaces are used to partition the wall segments. First, the floors are clustered based on their height. Next, a set of horizontal  $l \in L$  reference planes is computed based on the floor data (Fig 3(e)). These elevations serve as the reference levels for the project (Fig.4(right)). The wall surfaces are partitioned by the intersection of  $L \cap S$  in a set of surfaces per story  $S_l$  (Fig 3(f)).

The resulting wall segments are subjected to the wall clustering algorithm. First, a graph  $G(S_l, E)$  is defined for each story with each node  $s \in S_l$  a surface and each edge  $e \in E$  a connection between neighboring surfaces. The Euclidean distance between the boundaries is used as the criterion for the adjacency matrix of the graph. Pairwise relations are computed for each edge including the parallelity, coplanarity, orthogonality, dimensions, intersection of the edges with other surfaces and distance between boundaries. A majority vote based on heuristic thresholds is used to remove weak edges from the graph. the most probable clusters of surfaces  $s$  in  $S_l$  are computed by grouping the linked nodes (Fig 4(right)). As a result, the surfaces per story  $S_l$  are clustered in groups of wall candidates  $\forall w \in W, w = \{s_1, \dots, s_i\}$  (Fig.3(g)).

#### 4.1.4 Reconstruction

The reconstruction step computes the necessary parameters for the creation of the BIM geometry (Fig.3(h)). In this research, generic IfcWall entities are constructed based on the parameters extracted from the wall candidates  $W$ . The extracted parameters include the orientation, thickness, location and boundary of the walls. As is common in manual modelling procedures, standard wall cases are created. This implies that the reconstructed walls are

vertical, have a uniform thickness and have a height equal to the story height. Even though these abstractions introduce major deviations, it is a reliable basis to compare both reconstruction techniques.

**Orientation** The orientation of the IfcWalls is derived from the surfaces in  $W$ . First, the centre plane  $p(w)$  of the walls is defined by computing the normal  $\vec{n}_w$  and centre  $c_w$  based on each  $s \in w$  (Eq. 1). The calculations are weighted ( $\omega$ ) based on the surface areas of  $s$  for optimal fitting. Also, only the surfaces approximately parallel to the centre plane are taken into consideration.

$$p(w) = \begin{cases} c_w = \sum_{s \in w} \omega_s c_s \\ \vec{n}_w = \sum_{s \in w} \omega_s \vec{n}_s \end{cases} \quad (1)$$

**Wall thickness** As discussed above, each wall is computed with a uniform thickness. First, the surfaces on both faces of each wall  $s_i, s_j$  are identified using  $p(w)$  (Fig. 5). Next, the best fit thickness  $d$  of  $w$  is given by the weighted average distance from  $s_i$  and  $s_j$  to  $p(w)$  along  $\vec{n}_w$  (Eq. 2).

$$d(p(w), s_i, s_j, \omega) = \sum_{s \in s_i, s_j} \omega_s \frac{|ax_s + by_s + cz_s + d|}{\sqrt{a^2 + b^2 + c^2}} \quad (2)$$

The thickness of a wall in the model is considered to be dependent on the type of application. For true as-built models every wall has a unique thickness. However, most applications want to minimize the number of wall types. Therefore, the user can influence the number of wall thicknesses created in the project. By default, a clustering threshold of 3cm is set for the wall thickness. The result is a clustered thickness  $D$  for each wall  $w \in W$ . Some walls in the data set only consist of a single wall face due to occlusions or misclassification. As the goal is to create a volumetric entity for each wall, a default thickness is used for the reconstruction of these walls.

**Positioning** Once the clustered wall thickness  $D$  for each wall is determined,  $p(w)$  is optimized to represent the centreline of the wall as is common in BIM. The position of  $c_w$  is derived from equation 3:

$$p(w) = \begin{cases} D = (d_{s_i} + \omega_i \Delta d) + (d_{s_j} + \omega_j \Delta d) \\ c_p = \frac{D}{2} \end{cases} \quad (3)$$

where  $d_{s_i}$  and  $d_{s_j}$  are the distances from  $p(w)$  to  $s_i$  and  $s_j$ ,  $\Delta d$  the difference between the true thickness  $d$  and the clustered thickness  $D$ , and  $\omega$  the weights based on the surface areas.

**Wall Boundary** The length of the walls is derived from the boundaries of the surfaces  $s \in w$ . The bounding box of  $w$  is employed to determine the length of each wall. The height of the wall is equal to the story height as discussed in Section 4.1. The final BIM entities are created based on the extracted parameters (Fig.5(right)). The result is a set of volumetric IfcWall objects as shown in Fig.3(i).

## 4.2 2D approach

A popular approach for the reconstruction of BIM geometry from point cloud data is the use of raster images as discussed in Section 2. These 2D methods reduce the 3D point cloud to a set of planar raster images. This allows the use of efficient computer vision techniques for the segmentation and classification of the information. While these methods are typically restricted to Manhattan-world scenes, many AEC industry buildings consist of clean vertical walls and horizontal floors. A detailed overview of the steps and intermediate results is shown in Fig. 6. The following paragraphs discuss the consecutive steps for the segmentation and reconstruction of the wall geometry.

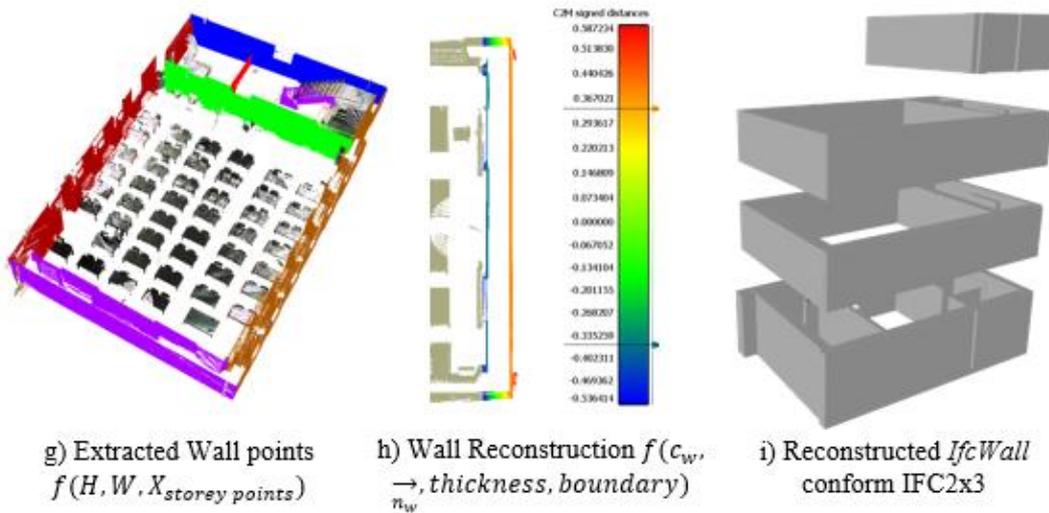
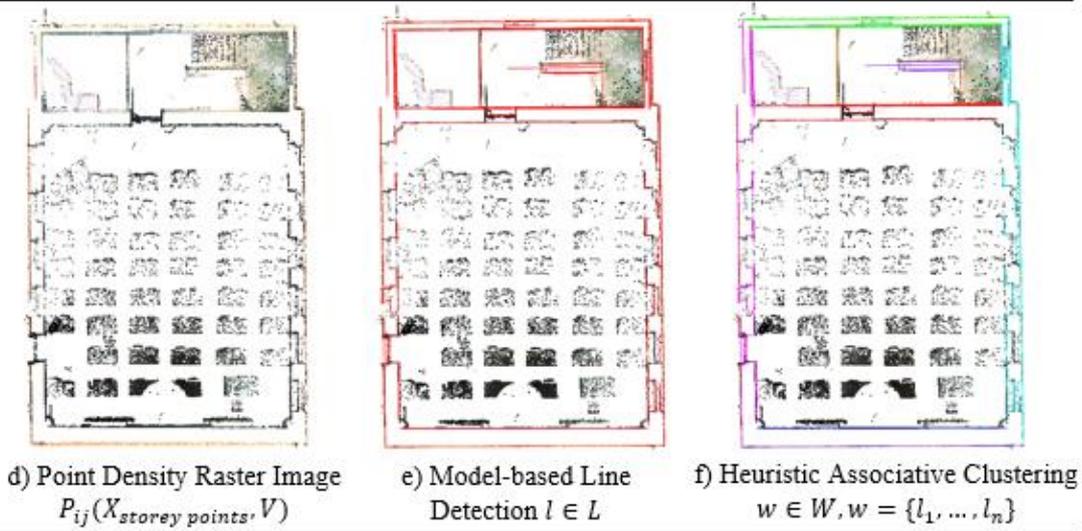
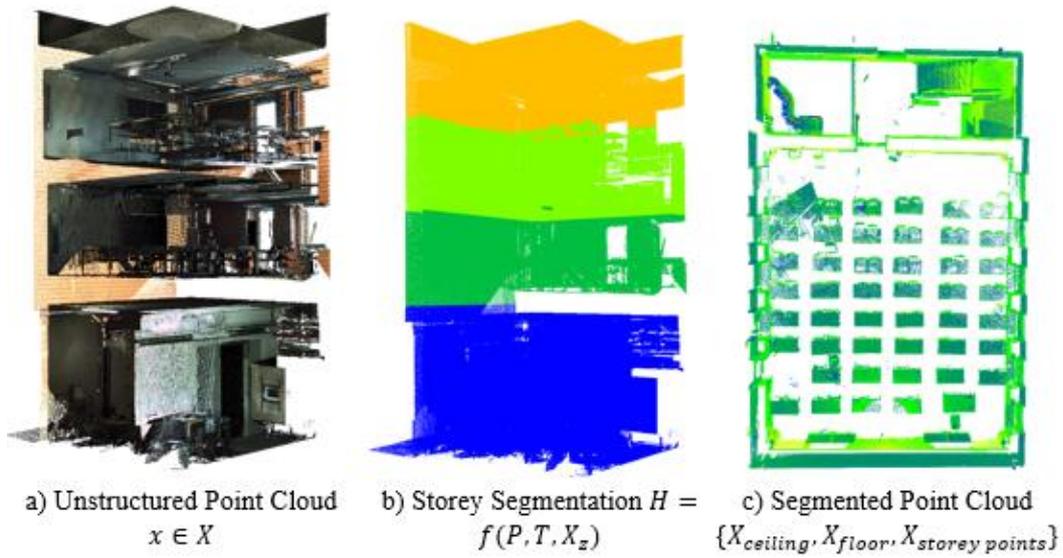


FIG.6: Overview workflow of automated 2D method from point cloud to *IfcWall* geometry on multi-story school facility.

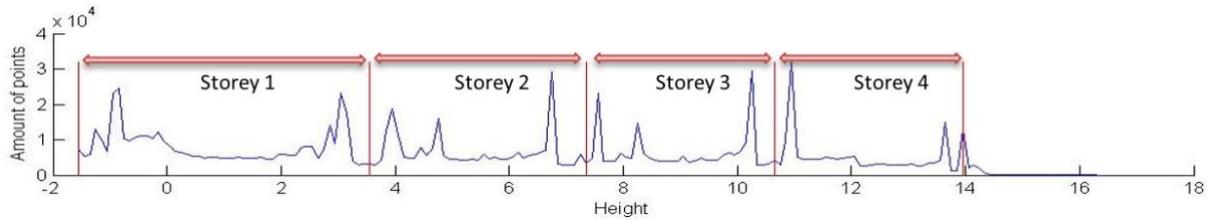


FIG. 7: Histogram  $H(X_z)$  of points along  $\vec{n}_z$  with the detected peaks and troughs. The extracted Segmentation Heights  $H = f(P, T, X_z)$  are depicted in red.

#### 4.2.1 Story Segmentation

A key aspect in 2D approaches for building element detection is the segmentation of the different stories. Opposed to the floor detection through SVM classification used in the 3D approach, the proposed 2D method directly exploits the point density of the initial point cloud.

Algorithm 1: Pseudocode for Story detection using point density histogram.

---

```

1: procedure Input: Point cloud data(X)
2:   Hist = Histogram(Xz)
3:   P = FitPeak(Hist)           d P=peak values
4:   T = FitTrough(Hist)        d T=trough values
5:   H(0) = T(1)                d H=Segmentation Height
6:   for i = 1 : T do
7:     If (Ti - Pbefore < Ht1 & Pafter - Ti < Ht1 & Ti - Hi-1 > Ht2)
8:       Hi = Ti
9:   end for
10: end procedure

```

---

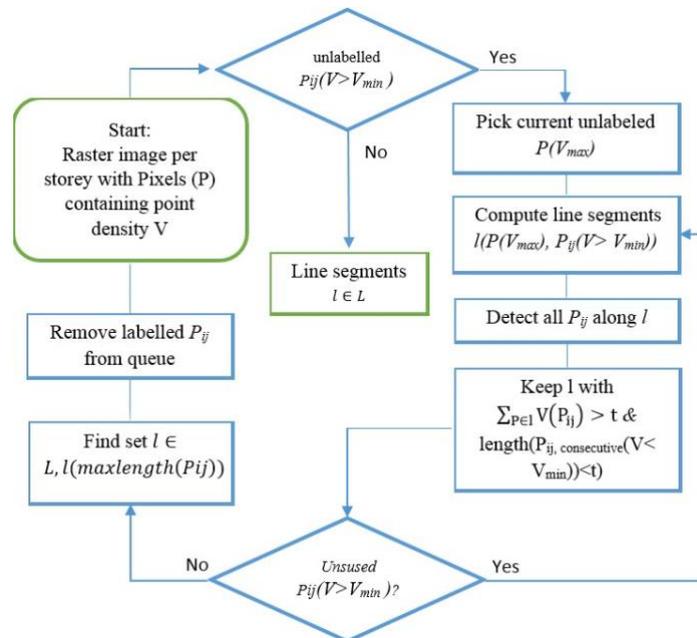


FIG. 8: Overview line segment  $l \in L$  detection method given  $P_{ij}(V > V_{min})$ .

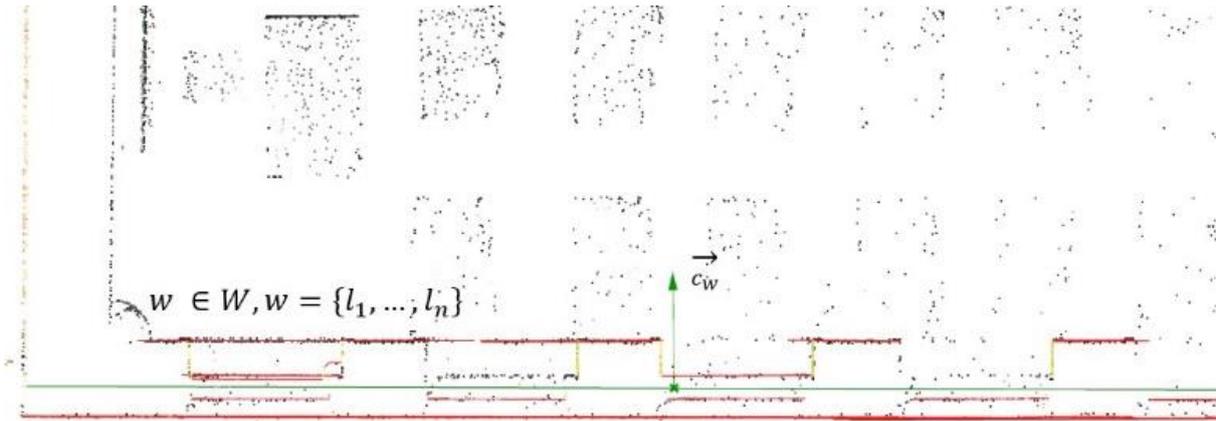


FIG. 9: Computation of  $\vec{n}_w$  (green) given  $w = \{l_1, \dots, l_n\}$  (red) exclusive the non-parallel  $l \in w$

First, the point cloud is restructured as a voxel octree with a sampling density of one point per cubic centimeter. Next, a vertical point histogram is generated similar to Oesau et al. (Oesau et al., 2014). As depicted in Fig. 7, the point density peaks at the height of the floor and ceiling data. Additionally, the point density will reach a minimum in the space between a ceiling and the floor above it. A peak and trough detection algorithm is presented that exploits this information in order to segment the story points (Algorithm 1). The detected peaks  $P$  and troughs  $T$  are iteratively grouped per two peaks (ceiling and floor) and a trough (gap between them) to compute the segmentation height  $H$ . To avoid false positives,  $H_i - H_{i-1}$  is conditioned to be greater than a threshold  $H_{t1}$ . Also, the height difference of  $P$  and  $T$  should lie within a threshold  $H_{t2}$ . The derived segmentation heights  $H$  are used to partition the point cloud (Fig. 7). The result is a segmented data set consisting of floors points  $X_{f \text{ floor}}$  (the bottom peak inliers), ceiling points  $X_{ceiling}$  (the above peak inliers) and the story points  $X_{storypoints}$  (Fig. 6(c)).

#### 4.2.2 Wall detection

In this step, the points representing the wall geometry are extracted from the story points. A raster image approach is proposed that represents the density of the story points in a plane (Fig. 6(d)). The advantage of this representation opposed to a section is that the entire data set is analyzed. A section might miss important details as it only observes a slice of data. Also, approaches that only evaluate the floor and ceiling data such as Valero et al. (Valero et al., 2012) or Macher et al. (Macher et al., 2015) heavily rely on the presence of that information and do not extract exterior walls.

**Line detection** The raster image  $P$  is created by projecting the points per story along the  $Z$ -axis onto the  $XY$  plane (Fig. 6(d)). Each pixel  $P_{ij} \in P$  contains the point density  $V$  in vertical direction as an attribute. Next, a 2D model-based line detection algorithm is implemented to extract line segments  $l \in L$ . An overview of the method is shown in Fig. 8. Iteratively, lines segments  $l$  are computed between the pixel with the highest density  $P(V_{max})$  and other unlabeled  $P_{ij} (V > V_{min})$ .

All inlier pixels  $P_{ij} \in l$  are detected. The sum of the point densities  $\sum_{P \in l} V(P_{ij})$  is considered as the validation criterion. Additionally the candidate segments are tested for gaps and pixels with insufficient point density  $P_{ij} (V < V_{min})$ . A set of longest lines is defined by removing the overlap between the extracted segments. The extracted  $P_{ij} (V > V_{min}) \in l$  are removed from the candidate pool and the process is iterated until all pixels with sufficient density are processed. The result is a set of line segments  $l \in L$  representing all significantly dense pixels (Fig. 6(e)).

**Clustering** The line segments  $L$  are associatively clustered to create consistent wall line geometry  $w \in W$ . An iterative heuristic line-growing algorithm is implemented to merge the individual line segments. Starting with the longest line segment  $l(P_{max})$ , nearby candidate segments  $l$  are tested for parallelity and collinearity. The Euclidean distance between segments is used as a criterion for the neighborhood search. Once a cluster of line segments is created, small segments that are enclosed by the 2D convex hull of the cluster are also considered inliers. The result of the line-growing algorithm is a set of wall lines  $w \in W, w = \{l_1, \dots, l_n\}$  as shown in Fig. 6(f). Finally, the wall points are extracted based on the story height  $H$  and the wall lines  $W$ . A 3D Bounding box is defined for each wall after which the  $X_{storypoints}$  are segmented. The results are point cloud clusters containing the wall geometry per story (Fig. 6(g)).

### 4.2.3 Reconstruction

Similar to the first method, the reconstruction step computes the necessary parameters for the creation of the generic IfcWall entities. The extracted parameters include the orientation, thickness, location and boundary of the walls. For comparison purposes, the walls are reconstructed with the same conditions as the 3D method. Standard wall cases are created which are vertical, have a uniform thickness and have a height equal to the story height.

**Orientation** The orientation of the IfcWall entities is derived from the wall lines  $W$ . The wall normal  $\vec{n}_w$  is computed orthogonally to the direction of the wall lines  $w \in W$  (Fig. 9). For optimal fitting, weights  $\omega$  are introduced and only the lines approximately parallel to the direction of the averaged line are taken into account. In contrast to the weights of the 3D method that were based on the surface areas, the weights in the 2D method are based on the length of the wall lines. However, the difference in weights is expected to have a negligible impact on the computation of  $\vec{n}_w$  as the wall lines are near-parallel.

**Wall thickness** As discussed above, each wall is computed with a uniform thickness. The best fit wall thickness  $d$  is computed using a histogram method similar to the story segmentation method in Section 4.2.1. A point density histogram is defined by plotting the wall points of each  $w$  along  $\vec{n}_w \cdot d$  is defined as the distance between the detected peaks in the histogram (Fig. 10). As previously discussed in the 3D method, the number of reconstructed wall types is clustered to limit the number of wall types created. The walls with a single wall face or peak are again given a default thickness. The result is a clustered thickness  $D$  for each wall  $w \in W$ .

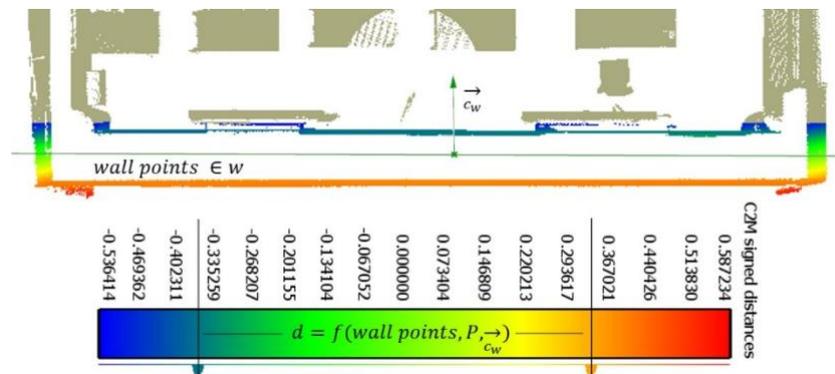


FIG. 10: Overview wall thickness  $d$  computation given wall points  $\in w$ . Peak detection  $P$  given

**Positioning & wall boundary** In the 2D method, the position and wall boundary is extracted similar to the 3D method. As previously presented, the final positioning of the wall geometry is derived from equation 3 given  $d$ ,  $D$  and the position of both peaks. The final location is weighted based on the number of points in both peaks. The wall boundary is derived from the longest line along the weighted averaged direction of  $l \in w$  given the dimensions of the bounding box of the extracted wall points. The height of the wall is equal to the story height  $H$ . The final BIM entities are created based on the extracted parameters. The result is a set of volumetric IfcWall objects as shown in Fig.6(i).

## 5. EXPERIMENTS

Both approaches were compared to the same test data. A four-story building on a university campus was selected for the experiments (Fig. 1). This recently renovated school facility consists of a laboratory, two classrooms, a staircase and a maintenance room. A large amount of clutter is present in the environment. A total of 55 scans were acquired resulting in over half a billion 3D points. No imagery was captured during the scanning. The resulting point cloud was processed independently by both reconstruction methods (Fig. 11).

The results were validated with the U.S. Institute of Building Documentation Level of Accuracy (LOA) Specification (USIBD, 2016) using the CloudCompare Software (Girardeau-Montaut D., 2016). For the comparison, 36 walls were identified as ground truth. By default, the walls types were clustered within 3cm and the single faced walls were given a thickness of 1cm. In the following paragraphs, the results for both methods are discussed.



FIG. 11: Overview reconstruction results. The initial point cloud (left), the reconstructed geometry by the 3D method (middle) and the reconstructed geometry by the 2D method (right).

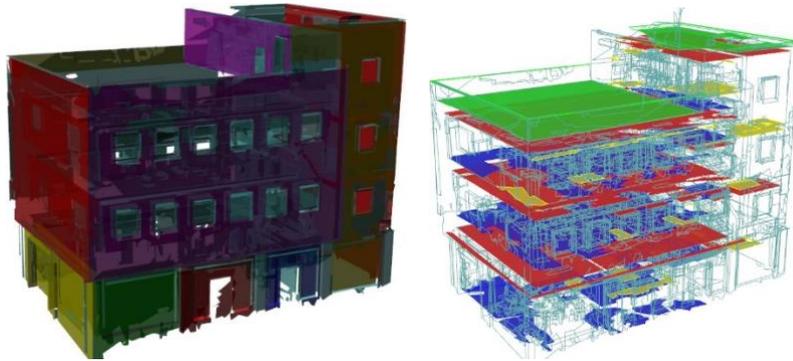


FIG. 12: Overview Classification Results, The detected wall surfaces (left) and the floors, ceilings and roof surfaces (right).

## 5.1 3D Approach

The presegmentation and primitive fitting was performed using Arithmetica's Pointfuse (Arithmetica Inc., 2015). A point sampling resolution of 1cm was used for the entire data set. Overall, 5670 surfaces were created with a standard deviation error of 0.001m. The probability  $P(S(y)|x)$  of the class labels given the feature values was computed using a Support Vector Machines (SVM) presented in previous work (Bassier et al., 2017). The available classes were floor, ceiling, roof, walls, beams and clutter. The model was trained on a wide variety of buildings including houses, castles, offices and factories. The geometric features included the surface area, dimensions  $s(u, v)$  and the orientation  $\vec{n}_s$ . The contextual features encode the coplanarity, parallelity, various topological relations and the proximity to characteristic reference surfaces. In total, 17 predictors were used. The results of the classification are shown in table 1 and Fig. 12. Overall, 87% of the wall surfaces were found with 85% precision, which is high considering the clutter and occlusion in the building. No less than 90% of the scene consists of clutter surfaces representing furniture and non-identified classes. False positives include built in closets, black boards and other wall like structures. Also, wall detailing such as the niches on the ground floor are prone to misclassification. However, the majority of wall detailing only has a minor impact on the reconstruction compared to the large wall surfaces.

Table 1: SVM classification Results

	Average [%]	Beams	Ceilings	Clutter	Floors	Roofs	Walls
Recall	76	47	85	77	86	74	87
Precision	80	69	77	77	88	83	85



FIG. 13: Overview 3D method: SVM classification with false positives in yellow (left) and the clustering results with overclustered geometry in yellow and correct clustering in green (right).

The wall surface clustering was performed with the heuristic graph-based algorithm using coplanarity, proximity and parallelity. Compared to the 36 walls that were considered as ground truth, the algorithm in the 3D method detected 43 walls. The overclustering of the walls is mainly due to gaps in between wall surfaces, causing the creation of multiple walls (Fig. 13(right)). Additionally, an increased number of walls is detected due to false positives from the classification step (Fig. 13 (left)).

The BIM object reconstruction was performed with the parameter extraction of the orientation, wall thickness, position and boundary. The reconstructed geometry is tested by observing the Euclidean distance between the manually segmented point cloud and the closest reconstructed geometry. Fig. 14 depicts the deviations between both data sets. As the distances between both models are not normally distributed, the entire dataset is compared to the 95% confidence intervals of the LOA specification. Overall, the geometry (minus the false positives of the classification) were reconstructed with 88% of the data within 5cm error (LOA20) and 54% within 1.5cm error (LOA30). While is very accurate considering the abstractions caused by the reconstruction constraints. One of the factors contributing to the accuracy is the weighted positioning of the walls that favors large wall surfaces. However, small surfaces are prone to increased errors. Other errors in the data set are caused by non-uniform wall thicknesses, clustered wall thickness, wall detailing and non-vertical walls.

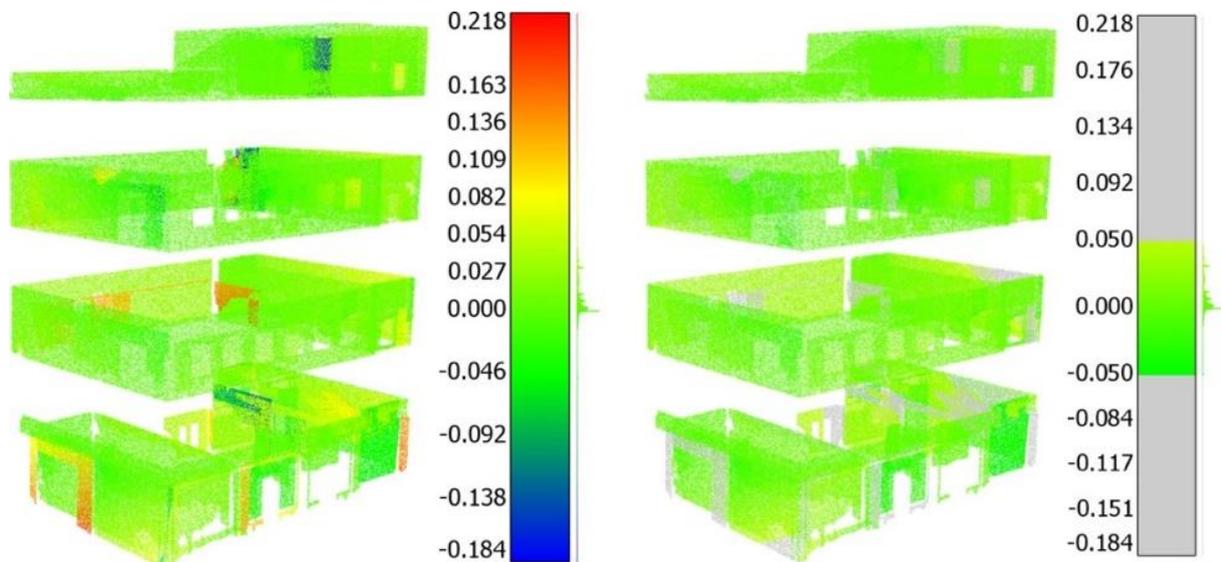


FIG. 14: Overview deviations [m] 3D method between reconstructed geometry and reference point cloud: The full error scale depicting grave error (left) and the error conform LOA20 (5cm 95% confidence interval) (right).

## 5.2 2D Approach

The second approach exploits raster images to segment the wall geometry. Prior to the segmentation, the data is restructured as a voxel octree. In the experiments, an octree of the test building was created with a spatial resolution of  $1\text{cm}^3$ . Following, the story segmentation was performed by representing the point cloud as a point density histogram along Z-direction. A default story height  $H_{t1}$  was defined of 3m and the inlier distance  $H_{t2}$  between peaks and troughs was set to 0.5m. 4 stories were detected using the peak and trough detection algorithm, resulting in 4 raster images with a resolution of  $1\text{cm}^2$  based on the projected voxels per story.

The line segments in the raster images were computed using a model-based line detection algorithm. The minimum point density  $P_{ij}(V_{\min})$  was set equal to a height of 2m and the maximum gap to 0.5m. Overall, 723 line segments were detected resulting in 81% recall and 83% precision for the wall detection. This is very promising given the fact that this method is constrained by Manhattan-world scene assumptions. However, several false positives were detected due the height of objects including tall doors and wall like geometry such as built-in closets (Fig. 15(left)). Also, low walls are overlooked due to their reduced point density in vertical direction. For instance, the roof walls on the fourth floor are not detected by the 2D approach (Fig. 11(right)). Similar to the 3D approach, the wall detailing is prone to errors but only have a minor impact on the final wall reconstruction compared to the larger wall segments.

The clustering of the line segments was performed by an heuristic line-growing algorithm based on proximity, parallelity and collinearity. Overall, 38 walls were detected by the 2D method which is closer to the 36 walls defined as ground truth. Similar to the 3D method, several erroneous walls are created due to false positives from the detection step. Fewer walls were created in comparison to the 3D method. This is caused by the iterative nature of the growing method as well as less strict clustering parameters. As a result, the 2D method connects line segments over larger spans (Fig. 15(right)).

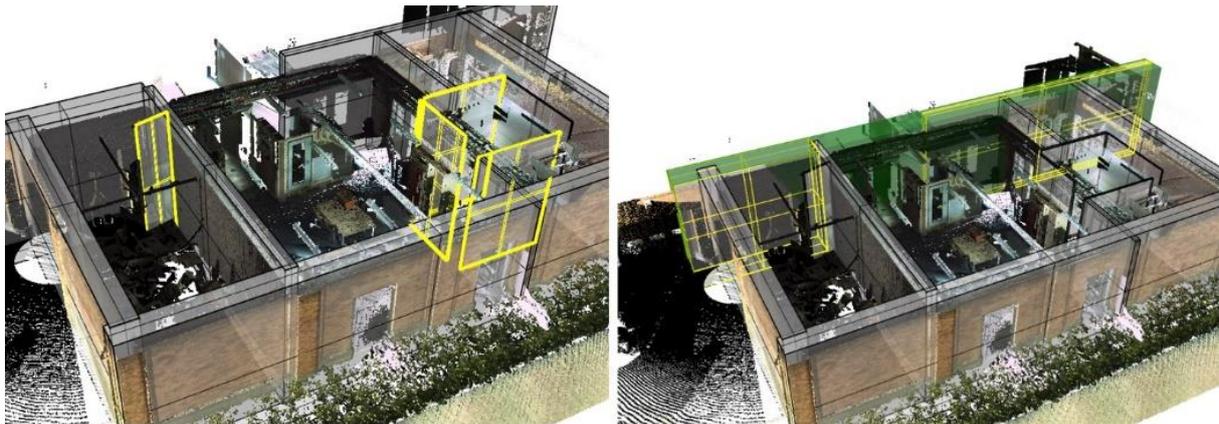


FIG. 15: Overview 2D method: model based line detection with false positives in yellow (left) and the clustering results with overclustered geometry in yellow and correct clustering in green (right).

The final reconstructed BIM objects are compared to the same ground truth as the first method. Fig. 16 depicts the deviations between the point cloud and the wall geometry. Overall, the walls were reconstructed with 55% of the data within 5cm error (LOA20) and 25% within 1.5cm error (LOA30). Again, a portion of the errors are caused by the reconstruction constraints concerning verticality, uniform wall thickness. A clear tradeoff is observed between the clustering and the reconstruction errors. The 3D method creates more clusters from the candidate wall surfaces and thus has increased wall reconstruction accuracy. The 2D method creates less clusters from the line segments but has increased reconstruction errors. These clustering errors have a large impact on the overall reconstruction error. There is a large difference between the impact of false positives and false negatives. Clustering with too strict parameters reduces the overall reconstruction error as single faced walls give near zero errors. Clustering with too loose parameters increases the reconstruction error dramatically as walls that are falsely grouped together will have large deviations.

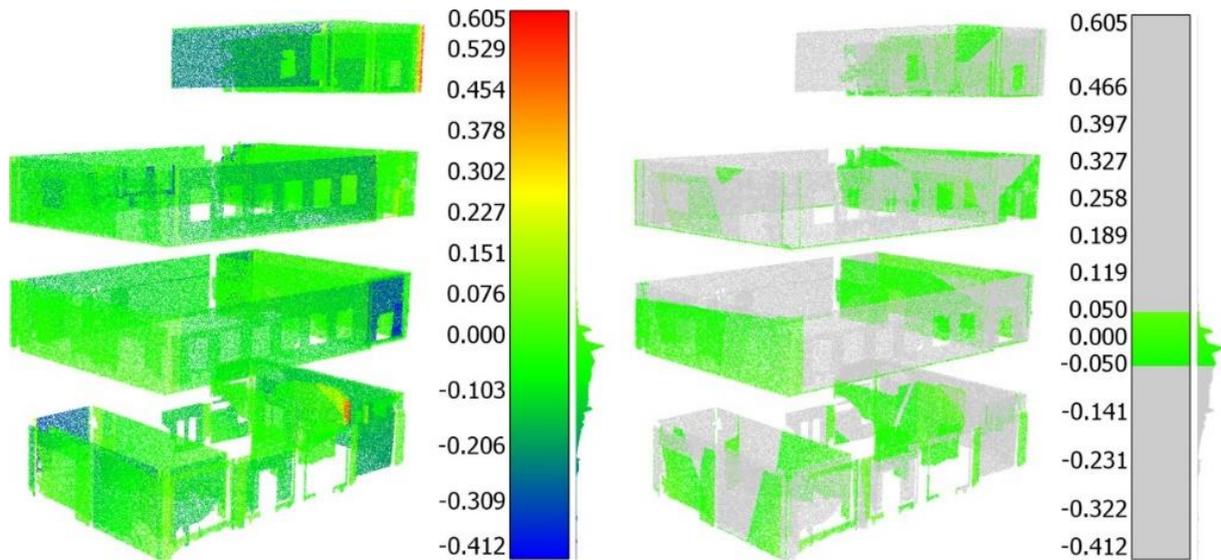


FIG. 16: Overview deviations [m] method 2 between reconstructed geometry and point cloud: The full error scale depicting grave error (left) and error according to LOA20 (right).

Table 2: Overview pros and cons 2D and 3D method

Category	2D method	3D method
Segmentation	Height based	Surface based
Classification	85% Recall 81% Precision	81% Recall 83% Precision
Clustering	43 walls	38 walls
Reconstruction	88% LOA20 54% LOA30	55% LOA20 25% LOA30

## 6. COMPARISON

In this section, the pros and cons of 3D and 2D algorithms are discussed based on the empirical data in Table 2. First, both the primitive labelling and the point density histogram method are capable of fast, reliable story detection in cluttered environments. This is expected since floor and ceiling surfaces typically have distinct point density signatures. However, in a project where multiple sets of elevations are required, e.g. buildings with intermediate levels, both algorithms have to be extended to properly segment the data. This can easily be dealt with by defining zones for the sets of elevations. In case the floor shows inconsistencies, e.g. only detached fragments of the floors are scanned, a point density evaluation might still detect the floor while the 3D classification is confused by the nearby surfaces. In contrast, the first method is able to deal with non-horizontal floors and ceilings.

The main difference between both methods is the wall detection. The 3D method uses SVM classification on a set of presegmented planar surfaces while the 2D method projects the data in a raster image and uses a model-based method for the detection. While both approaches manage to extract the required geometry, the 3D approach has access to more appropriate features resulting in increased recall and precision. For instance, the relative positioning in height is a major factor in the outlier detection of doors, blackboards, etc. In contrast, the computational effort is lower for the 2D approach. Overall, the 3D approach has better results in complex environments while the 2D method can deal more efficiently with Manhattan-world scenes. When comparing the 2D method to other research, it is observed that the use of point density in vertical direction is a reliable detector for wall geometry. While approaches exploiting ceiling and floor information such as Hong et al. (Hong et al., 2015), Valero et al. (Valero et al., 2012) or Macher et al. (Macher et al., 2015) are able to extract the boundaries of rooms, they struggle with zones that do not have distinct floor and ceilings such as staircases.

For the clustering, the 3D method optimizes for precision while the emphasis of the 2D method is on recall. The experiments show that a strict clustering leads to an excess of walls created as some surfaces fail to group. However, the walls that are created are very accurate and thus have lower reconstruction errors. Loose clustering parameters result in increased errors and fewer walls.

The reconstruction of the wall geometry is similar in both approaches. The use of weights during the parameter extraction has a significant impact on the reconstruction accuracy. While this results in increased local errors on small wall segments, the global error is reduced since the reconstructed geometry better aligns with the largest wall segments.

## 7. CONCLUSION & DISCUSSION

In this paper, a much-needed empirical comparison is presented between 2D and 3D BIM reconstruction methods. More specifically, the detection and reconstruction of wall objects in existing AEC industry buildings is investigated. Two state of the art methods were implemented to automatically reconstruct as-built BIM geometry. The first method consists of a modular 3D approach exploiting planar primitives. The second method consists of a 2D approach that exploits the point density in raster images. Both methods were evaluated using a real life test case and compared using the Level of Accuracy specification.

The point cloud is first segmented into planar clusters, after which a Random Forests classifier is used with geometric and contextual features for the semantic labelling. The final wall geometry is created based on the 3D point clusters representing the walls. The second method is an efficient Manhattan-world scene reconstruction algorithm that simultaneously segments and classifies the point cloud based on point feature histograms. The wall reconstruction is considered an instance of image segmentation by representing the data as 2D raster images.

Overall, both methods are able to reconstruct wall objects in highly cluttered environments. The test results proved that the two methods are capable of detecting wall geometry fully automatically with both high precision and recall. Additionally, the detected wall surfaces were clustered reliably. Both methods were able to reconstruct the geometry with high accuracy. However, there are clear differences in the detection and reconstruction errors which are linked to the respective 2D and 3D strategies. For instance, while both methods respectively detect 81% and 87% of the walls, the 2D method is inherently prone to the misclassification of tall vertical objects in the scene. Also, the parameter extraction is less accurate due to 2D abstractions of the raster imagery. In contrast, the LOD200 design strategy of the 3D method also has some inevitable errors since wall detailing influences the parameter extraction which it shouldn't. Overall, the test results show that the assumption of a uniform thickness and vertical walls introduce severe errors in the reconstruction and are one of the main obstacles for methods to achieve LOA30. This is caused by non-parallel, non-planar walls and details in the wall geometry. Despite these challenges, 88% of the reconstructed data is within 5cm (LOA20) of the initial geometry for the first method and 55% for the second method.

Both methods have their advantages and disadvantages. The first method exploits three dimensional features for the detection and clustering. This results in accurate detection of wall geometry even in complex scenes but is computationally demanding. The second approach exploits raster images which are very efficient but assume wall verticality. In the context of AEC buildings where wall features are straight forward, both approaches are fit for the detection and reconstruction of wall geometry.

From the presented methods and experiments, developers will be able to better judge the impact of their reconstruction algorithms. For instance, the degree of error introduced through abstractions such as Manhattan-scene assumptions significantly impact the detection rate of more complex structures which can be a deciding factor during the development stage. Also, stakeholders benefit from this work as they now have a more realistic view on the potential of both 2D and 3D methods. Overall, we state that the empirical testing of reconstruction algorithms on realistic test cases is a vital but frequently overlooked part in the existing literature. In future work, we will investigate the standardization of evaluation procedures to provide developers with the proper tools to evaluate the performance of their algorithms. Also, we will investigate how both strategies can overcome their current shortcomings during the consecutive steps of data interpretation and parameter estimation.

## ACKNOWLEDGEMENTS

This work was supported by the research group Geomatics of the Department of Civil Engineering, TC Construction of the KU Leuven University and the company 3DGeosolutions. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement 779962).

## REFERENCES

- Adan A., Huber D. (2011). 3D Reconstruction of Interior Wall Surfaces under Occlusion and Clutter, 2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, 275–281doi:10.1109/3DIMPVT.2011.42.
- Agarwal, R., Chandrasekaran, S., & Sridhar, M. (2018). Imagining construction's digital future. McKinsey Productivity Sciences Center.
- Anand A., Koppula H. S., Joachims T., Saxena A. (2012). Contextually guided semantic labeling and search for three-dimensional point clouds, *The International Journal of Robotics Research* 32 (1), 19–34. doi: 10.1177/0278364912461538.
- Arithmetica, L. (2015). Pointfuse. <http://pointfuse.com/>
- Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., & Savarese, S. (2016). 3D Semantic Parsing of Large-Scale Indoor Spaces. *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, <https://doi.org/10.1109/CVPR.2016.170>
- BIMForum (2015). Level of Development Specification.
- Barnea S., Filin S. (2013). Segmentation of terrestrial laser scanning data using geometry and image information, *ISPRS Journal of Photogrammetry and Remote Sensing* 76, 33–48. doi: 10.1016/j.isprsjprs.2012.05.001.
- Bassier M., Vergauwen M., Van Genechten B. (2016). Standalone Terrestrial Laser Scanning for Efficiently Capturing AEC Buildings for As-Built BIM, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences III-6 (July)* 49–55. doi:10.5194/isprsannals-III-6-49-2016.
- Bassier, M., Vergauwen, M., & Van Genechten, B. (2016). Automated Semantic Labelling of 3D Vector Models for Scan-to-BIM. 4th Annual International Conference on Architecture and Civil Engineering (ACE 2016), 93–100. <https://doi.org/10.5176/2301-394X>
- Bassier, M., Vergauwen, M., & Van Genechten, B. (2017). Automated Classification of Heritage Buildings for As-Built BIM using Machine Learning Techniques. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, IV-2/W2*, 25–30. <https://doi.org/10.5194/isprs-annals-IV-2-W2-25-2017>
- Bosche F. (2012). Plane-based registration of construction laser scans with 3D/4D building models, *Advanced Engineering Informatics* 26 (1), 90–102. doi: 10.1016/j.aei.2011.08.009.
- Brilakis M., Lourakis, R. Sacks, S. Savarese, S. Christodoulou, J. Teizer, Makhmalbaf (2010). Toward automated generation of parametric BIMs based on hybrid video and laser scanning data, *Advanced Engineering Informatics*, 24 (4), 456–465. doi: 10.1016/j.aei.2010.06.006.
- Budroni A., Bohm J. (2010). Automatic 3d Modelling Of Indoor Manhattan- world Scenes From Laser Data, *ISPRS Commission V Mid-Term Symposium: Close Range Image Measurement Techniques XXXVIII*, 115– 120.
- Budroni A., Bohm J. (2009). Toward automatic reconstruction of interiors from laser data, *Proceedings of Virtual Reconstruction and Visualization of Complex Architectures (3D-Arch)*.
- Dimitrov A., Golparvar-Fard M. (2015). Segmentation of building point cloud models including detailed architectural/structural features and MEP systems, *Automation in Construction*, 32–45. doi: 10.1016/j.autcon.2014.12.015.
- Furukawa Y., Curless B., Seitz S. M., Szeliski R. (2009). Reconstructing building interiors from images, *IEEE 12th International Conference on Computer Vision*. doi:10.1109/ICCV.2009.5459145.
- Gimenez L., Hippolyte J.-L., Robert S., Suard F., Zreik K. (2015). Review: reconstruction of 3D building information models from 2D scanned plans, *Journal of Building Engineering* 2, 24–35. doi: 10.1016/j.jobbe.2015.04.002.
- Girardeau-Montaut, D. (2016). CloudCompare. <http://www.danielgm.net/cc/>
- Grilli E., Menna F., Remondino F. (2017). A Review of Point Clouds Segmentation And Classification Algorithms, *Int. Arch. Photogramm. Remote Sens. Spatial Inf.*, 1–3. doi:10.5194/isprs-archives-XLII-2-W3-339-2017.



- Hichri N., Stefani C., Luca L. D., Veron P., Hamon G. (2013). From Point Cloud to BIM: A Survey of Existing Approaches, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XL- 5/W2*, 2–6.
- Hong S., Jung J., Kim S., Cho H., Lee J., Heo J. (2015). Semi-automated approach to indoor mapping for 3D as-built building information modeling, *Computers, Environment and Urban Systems* 34–46. doi: 10.1016/j.compenvurbsys.2015.01.005.
- Hulik R., Spanel M., Smrz P., Materna Z. (2014). Continuous plane detection in point-cloud data based on 3D Hough Transform, *Journal of Visual Communication and Image Representation* 25 (1), 86–97. doi: 10.1016/j.jvcir.2013.04.001.
- Ilter, D., & Ergen, E. (2015). BIM for building refurbishment and maintenance: current status and research directions. *Structural Survey*, 33(3). <https://doi.org/10.1108/SS-02-2015-0008>
- Jung J., Hong S., Jeong S., Kim S., Cho H., Hong S., Heo J. (2014). Productive modeling for development of as-built BIM of existing indoor structures, *Automation in Construction* 42, 68–77. doi: 10.1016/j.autcon.2014.02.021.
- Kabolizade, M., Ebadi, H., & Mohammadzadeh, A. (2012). Design and implementation of an algorithm for automatic 3D reconstruction of building models using genetic algorithm. *International Journal of Applied Earth Observation and Geoinformation*, 19, 104–114. <https://doi.org/10.1016/j.jag.2012.05.006>
- Lari Z., Habib A. (2014). An adaptive approach for the segmentation and extraction of planar and linear / cylindrical features from laser scanning data, *ISPRS Journal of Photogrammetry and Remote Sensing* 93 , 192–212.
- Macher H., Landes T., Grussenmeyer P. (2015). Point clouds segmentation as base for as-built BIM creation, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences II-5/W3* 191–197. doi:10.5194/isprsannals-II-5-W3-191-2015.
- Mnih, V., & Hinton, G. (2012). Learning to Label Aerial Images from Noisy Data. *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, 567–574.
- Mura, C., Mattausch, O., Jaspe Villanueva, A., Gobbetti, E., & Pajarola, R. (2014). Automatic room detection and reconstruction in cluttered indoor environments with complex room layouts. *Computers & Graphics*, 44, 20–32. <https://doi.org/10.1016/j.cag.2014.07.005>
- Nguyen, B. Le (2013). 3D Point Cloud Segmentation: A survey, *6th IEEE Conference on Robotics, Automation and Mechatronics*, 225–230.
- Niemeyer, J., Rottensteiner, F., & Soergel, U. (2014). Contextual classification of lidar data and building object detection in urban areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87, 152–165. <https://doi.org/10.1016/j.isprsjprs.2013.11.001>
- Ochmann, S., Vock, R., Wessel, R., & Klein, R. (2016). Automatic reconstruction of parametric building models from indoor point clouds. *Computers & Graphics*, 54, 94–103. <https://doi.org/10.1016/j.cag.2015.07.0088>
- Oesau S., Lafarge F., Alliez P. (2014). Indoor Scene Reconstruction using Primitive-driven Space Partitioning and Graph-cut, *Eurographics workshop on urban data modelling and visualization*.
- Patraucean V., Armeni I., Nahangi M., Yeung J., Brilakis I., Haas C. (2015). State of research in automatic as-built modelling, *Advanced Engineering Informatics* 29, 162–171. doi: 10.1016/j.aei.2015.01.001.
- Previtali M., Barazzetti L., Brumana R., Scaioni M. (2014). Towards automatic indoor reconstruction of cluttered building rooms from point clouds, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences II-5 (June)*, 281–288. doi:10.5194/isprsannals-II-5-281-2014.
- Previtali, M., Scaioni, M., Barazzetti, L., & Brumana, R. (2014). A flexible methodology for outdoor/indoor building reconstruction from occluded point clouds. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3(September), 119–126. <https://doi.org/10.5194/isprsannals-II-3-119-2014>
- Pu, S., Rutzinger, M., Vosselman, G., & Oude Elberink, S. (2011). Recognizing basic structures from mobile laser scanning data for road inventory studies. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6), S28–S39. <https://doi.org/10.1016/j.isprsjprs.2011.08.006>
- Richter R., Behrens M., Dollner J. (2013). Object class segmentation of massive 3D point clouds of urban areas using point cloud topology, *International Journal of Remote Sensing* 34 (23) 8408–8424. doi:10.1080/01431161.2013.838710.

- Rusu, R. B., Marton, Z. C., Blodow, N., Holzbach, A., & Beetz, M. (2009). Model-based and learned semantic object labeling in 3D point cloud maps of kitchen environments. *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, (November), 3601–3608. <https://doi.org/10.1109/IROS.2009.5354759>
- Savarese S. (2016). 3D Semantic Parsing of Large-Scale Indoor Spaces, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition 2016*. doi:10.1109/CVPR.2016.170.
- Schnabel R., Wahl R., Klein R. (2007). Efficient RANSAC for point-cloud shape detection, *Computer Graphics Forum* 26 (2) 214–226. arXiv:arXiv:1507.03351v1, doi:10.1111/j.1467-8659.2007.01016.x.
- Son H., Bosche F., Kim C. (2015). As-built data acquisition and its use in production monitoring and automated layout of civil infrastructure: A survey, *Advanced Engineering Informatics*.
- Su Y. T., Bethel J., Hu S. (2016). Octree-based segmentation for terrestrial LiDAR point cloud data in industrial applications, *ISPRS Journal of Photogrammetry and Remote Sensing*, 59–74. doi:10.1016/j.isprsjprs.2016.01.001.
- Tang P., Huber D., Akinci B., Lipman R., Lytle A. (2010). Automatic reconstruction of as-built building information models from laser-scanned point clouds: A review of related techniques, *Automation in Construction*, 19 (7), 829–843. doi: 10.1016/j.autcon.2010.06.007.
- Thomson C., Apostolopoulos G., Backes D., Boehm J. (2013). Mobile Laser Scanning for Indoor Modelling, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences II-5/W2* (November) 289–293. doi:10.5194/isprannals-II-5-W2-289-2013.
- Thomson, C., & Boehm, J. (2015). Automatic geometry generation from point clouds for BIM. *Remote Sensing*. <https://doi.org/10.3390/rs70911753>
- Turner E., Zakhor A. (2013). Watertight Planar Surface Meshing of Indoor Point-Clouds with Voxel Carving, *2013 International Conference on 3D Vision*, 41–48doi:10.1109/3DV.2013.14.
- Turner, E., & Zakhor, A. (2014). Floor Plan Generation and Room Labeling of Indoor Environments from Laser Range Data. *GRAPP, International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 1–12. <https://doi.org/10.5220/0004680300220033>
- U.S. General Services Administration (2011). *GSA BIM Guide 08 - Facility Management*, no. 1. arXiv:arXiv:1011.1669v3, doi:10.1007/s13398-014-0173-7.2.
- U.S. Institute of Building Documentation (2016). *USIBD Level of Accuracy (LOA) Specification Guide C120 ver. 0.95 TM Guide*.
- Valero E., Adan A., Cerrada C. (2012). Automatic method for building indoor boundary models from dense point clouds collected by laser scanners, *Sensors (Switzerland)* 16099–16115. doi:10.3390/s121216099.
- Volk R., Stengel J., Schultmann F. (2014). Building Information Modeling (BIM) for existing buildings - Literature review and future needs, *Automation in Construction* 38 109–127. doi: 10.1016/j.autcon.2013.10.023.
- Wang, C., Cho, Y. K., & Kim, C. (2015). Automatic BIM component extraction from point clouds of existing buildings for sustainability applications. *Automation in Construction*. <https://doi.org/10.1016/j.autcon.2015.04.001>
- Weinmann M., Jutzi B., Hinz S., Mallet C. (2015). Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers, *ISPRS Journal of Photogrammetry and Remote Sensing* 105, 286–304. doi: 10.1016/j.isprsjprs.2015.01.016.
- Wolf D., Prankl J., Vincze M. (2015). Fast Semantic Segmentation of 3D Point Clouds using a Dense CRF with Learned Parameters, *IEEE International Conference on Robotics and Automation (ICRA)*. Doi: 10.1109/ICRA.2015.7139875.
- Xiong X., Adan A., Akinci B., Huber D. (2013). Automatic creation of semantically rich 3D building models from laser scanner data, *Automation in Construction* 31, 325–337. doi: 10.1016/j.autcon.2012.10.006.
- Yang, B., & Dong, Z. (2013). A shape-based segmentation method for mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 81, 19–30. <https://doi.org/10.1016/j.isprsjprs.2013.04.002>
- Zlot R., Bosse M., Greenop K., Jarzab Z., Juckes E., Roberts J. (2013). Efficiently capturing large, complex cultural heritage sites with a handheld mobile 3D laser mapping system, *Journal of Cultural Heritage*, 1–9. doi: 10.1016/j.culher.2013.11.009.