



Automatic BIM component extraction from point clouds of existing buildings for sustainability applications

Chao Wang ^{a,*}, Yong K. Cho ^a, Changwan Kim ^b

^a School of Civil and Environmental Engineering, Georgia Institute of Technology, 790 Atlantic Drive, Atlanta, GA 30332-0355, USA

^b Department of Architectural Engineering, School of Engineering, Chung-Ang University, 221 Heukseok-dong, Dongjak-gu, Seoul 156-756, Republic of Korea



ARTICLE INFO

Article history:

Received 5 August 2014

Received in revised form 15 January 2015

Accepted 4 April 2015

Available online xxxx

Keywords:

As-is model

BIM

Laser scanner

Point cloud

Object recognition

Energy simulation

ABSTRACT

Building information models (BIMs) are increasingly being applied throughout a building's lifecycle for various applications, such as progressive construction monitoring and defect detection, building renovation, energy simulation, and building system analysis in the Architectural, Engineering, Construction, and Facility Management (AEC/FM) domains. In conventional approaches, as-is BIM is primarily manually created from point clouds, which is labor-intensive, costly, and time consuming. This paper proposes a method for automatically extracting building geometries from unorganized point clouds. The collected raw data undergo data downsizing, boundary detection, and building component categorization, resulting in the building components being recognized as individual objects and their visualization as polygons. The results of tests conducted on three collected as-is building data to validate the technical feasibility and evaluate the performance of the proposed method indicate that it can simplify and accelerate the as-is building model from the point cloud creation process.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Building energy performance and other measurements such as acoustic and fire propagation simulation results are becoming increasingly important for design decisions concerning building sustainability that have to be made in the early design stage. Simulation in this paper means using computer-based tools to simulate the performance analysis of a building throughout an entire year of operation. The simulation results can be also used to evaluate the payback of green building solutions and validate points required by the U.S. Green Building Council's Leadership in Energy and Environmental Design Rating System (LEED™). It can provide quantitative evaluation of the proposed design.

President Obama launched the Better Building Challenge which asks leading organizations to commit to reducing the energy use of their buildings by 20% by 2020. In the United States, the majority of the energy usage is consumed by building sector [4,7,8], and the largest part of energy consumption is contributed by the existing buildings. In addition, the average age of the existing buildings is more than five decades, and 85% of them were built before year 2000 [7]. Therefore, any effort to significantly improve building energy efficiency and reduce environmental impacts must consider the existing buildings [53].

Currently, many building performance simulation tools are available to the public. Most of the tools require building geometry data as the simulation inputs, and the geometry data of the individual building envelope component (exterior walls, openings, roofs, and etc.) can further enable a detailed simulation [48]. Each component can be represented as an object, capable of representing its relationship with the walls in which it is placed. Currently, the building geometry data are able to be extracted from its corresponding BIM if the model is available [53]. BIM can provide views from multiple angles, and each individual object element has its corresponding information attached. More importantly, abstract objects in BIMs such as space and zone, can be created by defining the relationships between the physical building elements [49].

For most engineering analysis tools, not all the semantic information that BIM contains about a building is essential. Further, attempts at exporting all the information that BIM contains about a building often fail because a building model is too complex or not built correctly for simulation. For example, energy simulation tools only require a thermal view of a building, which is a simplistic representation of a building containing all the information about each room such as volume, geometry, and adjacency. With the help of neutral file formats such as green building XML (gbXML) schema, Industrial Foundation Classes (IFC), and ifcXML, the communication between BIM modeling tools and third party simulation tools is becoming less complicated [53]. However, BIM is still not always available for existing buildings. Furthermore, because the buildings are constantly being renovated, even a BIM does exist, it may be out of date or inaccurate. The preparation for as-is BIM

* Corresponding author. Tel.: +1 402 590 7278.

E-mail addresses: cwang2@gatech.edu (C. Wang), yong.cho@ce.gatech.edu (Y.K. Cho), changwan@cau.ac.kr (C. Kim).

is usually a time-consuming, labor-intensive, and costly process. Moreover, different BIM modelers inevitably create different models even when they are modeling the same building using the same modeling tool [50].

Although much progress has been made in automating visual data workflows, construction progress assessment, defect management, and building performance simulation are not fully attuned to the benefits of automatic as-is geometric creation during the design, construction, and facility management stages. Point clouds are increasingly being used as 3D as-is geometric representations of buildings. An as-is point cloud, obtained from a laser scanner or from the processing of massive photographs, comprises millions of individual points, each having its own 3D relative coordinate information. Although much work has been done on the processing of point cloud data for progress in construction and safety monitoring [11], performance visualization [11], and bridge management [6,14], not much work has been done to facilitate simulation of building performance. Further, as regards practicability, the current point clouds processing technologies are still in the very early stages.

The primary objective of our study is to provide a preliminary solution that automatically and rapidly creates 3D geometric model of existing buildings from point cloud data that can be further utilized for building performance simulation applications. We collected point cloud data from a laser scanner system and processed it to recognize and distinguish different building envelope components (such as windows, doors, walls, and roof) as individual objects for utilization in the applications mentioned above. (The applications of our proposed method are out of the scope of this paper and so will be published in subsequent papers in the near future.) In the ensuing sections, we first review the state-of-the-art in this area, Section 2, and then give an overview of our proposed method, Section 3. In Section 4, we outline the implementation of our proposed method, discuss the preliminary

experiments conducted to evaluate its efficacy, and analyze the results obtained. Section 5 validates the feasibility of applying the auto-generated gbXML based model in a building energy simulation tool. Finally, we outline prospective future work and conclude this paper in Section 6.

2. Background

In this section, we discuss the current point cloud collection methods and recent research efforts towards object recognition and modeling from point clouds.

2.1. State-of-the-art point cloud collection methods

There are two major methods of achieving point cloud data, processing massive photographs or using a laser scanner. Golparvar-Fard et al. [11] introduced an image-based as-built modeling technique based on computing from the images themselves, the photographer's locations and orientations, and a sparse 3D geometric representation of the as-built scene using daily progress photographs. The major advantages of photogrammetric systems include fast data collection rates (tens to hundreds of 1024×1024 pixel frames per second), and acquisition of rich color and textural information about workspace objects for appearance based object recognition. However, this method has a number of limitations: Different lighting and weather conditions make it difficult to use time-lapse photography for performing consistent image analysis at occluded and dynamic site conditions [1,12,13]. Further, the geometry of the area will be overlooked if common features from multiple images cannot be found.

Compared to photography, laser scanners facilitate wide-range measurements at higher resolutions and accuracies, and are generally not limited by ambient conditions during operation [14]. Laser scanning

Category	Advantages	Limitations
Commercial software programs	<ul style="list-style-type: none"> Providing a point cloud processing plug-in capability for many of the most popular CAD systems Semi-automatic pipe center line creation Designed for various applications 	<ul style="list-style-type: none"> Need manual point cloud segmentation and modeling object category selection
	<ul style="list-style-type: none"> Enabling user for comparison of as-designed model vs. as-is point cloud 	<ul style="list-style-type: none"> Mainly for industrial application Need manually creating a pipe center line for pipe modeling
	<ul style="list-style-type: none"> More easily performing whole-project review. Customized standard equipment library Externally referencing or parametrically creating structural elements Combine 3D scanning, imaging and position data 	<ul style="list-style-type: none"> Model spec-driven pipelines and components semi-automatically or manually Mainly for modeling pipe, equipment, and structure
	<ul style="list-style-type: none"> Semi-automatically generating 2D plans from 3D laser scanner data Designed for various applications Combine 3D scanning, imaging and position data 	<ul style="list-style-type: none"> Two manually selected reference points needed for pipeline modeling Openings (windows, doors) need to be manually created in the 2D plans.
	<ul style="list-style-type: none"> Accurately positioning and clash checking new 3D design in situ within an existing installation 	<ul style="list-style-type: none"> Need manual point cloud segmentation and modeling object category selection Limited for industrial application
	<ul style="list-style-type: none"> Combine 3D scanning, imaging and position data Semi-automatically modeling pipes 	<ul style="list-style-type: none"> Need manual point cloud segmentation and modeling object category selection
	<ul style="list-style-type: none"> Semi-automated feature extraction Deep integration with Revit, AutoCAD, PDMS and other platforms Designed for various applications 	<ul style="list-style-type: none"> Need manually segmenting point cloud and selecting object category for object modeling
Academic research efforts	<ul style="list-style-type: none"> Automatically modeling for indoor environment Modeling regular shape 	<ul style="list-style-type: none"> Difficult to accurately model curved shapes Incomplete building envelope components recognition

Fig. 1. Literature review of the current as-is BIM recognition techniques.

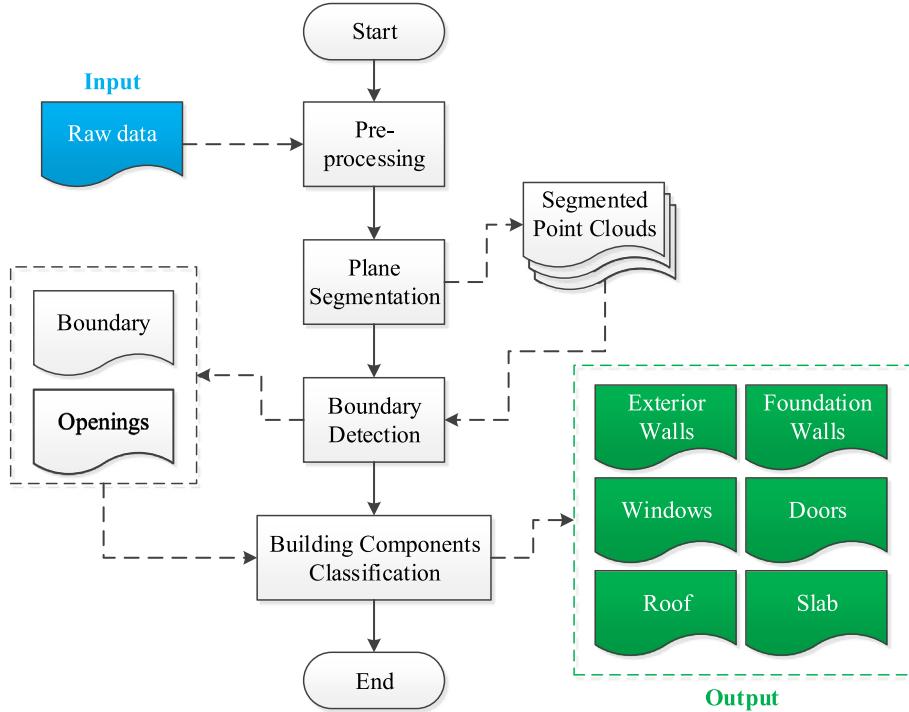


Fig. 2. Flowchart for our proposed method.

can also better holistically address all of the listed inefficiencies associated with the current practice of progress monitoring through rapid and detailed geometric data collections than other 3D remote sensing technologies [1]. In the domains of construction and facility management, researchers have conducted various studies investigating the issues related to utilizing laser scanners for a wide range of purposes, including fast workspace modeling [2,15], real-time safety management on site [16], construction progress monitoring [5,17–19,29], defect detection [3,20], as-built modeling [21–25], deflection assessments of bridges [26–32], and pavement thickness assessments [28].

The point clouds collected from various devices can be categorized as either organized or unorganized. An organized point cloud has a data structure that is similar to an image or a matrix, and each point of the point cloud has its index in rows and columns. Such point clouds include data collected from stereo cameras or time-of-flight cameras. The advantage of the organized point cloud over the unorganized point cloud

is that data processing is more efficient because the relationship between adjacent points or nearest neighbors is known. In unorganized point clouds, no data structure or point reference exists between points because of varied sizes, resolutions, densities, and point sequences. As a result, more time is usually consumed processing unorganized point cloud data.

2.2. Recent research efforts towards as-is BIM recognition from point clouds

Manually creating 3D model from point cloud is a labor-intensive and time-consuming process. Many commercial software programs or plug-ins have been developed to accelerate this manual process. For example, *Leica CloudWorx* [34] is able to automatically create a pipe center line based on manually selected pipe, and then the pipe can be manually created following the center line; *Intergraph Smart 3D for Plants* [35] can automatically model pipes after user identifying the

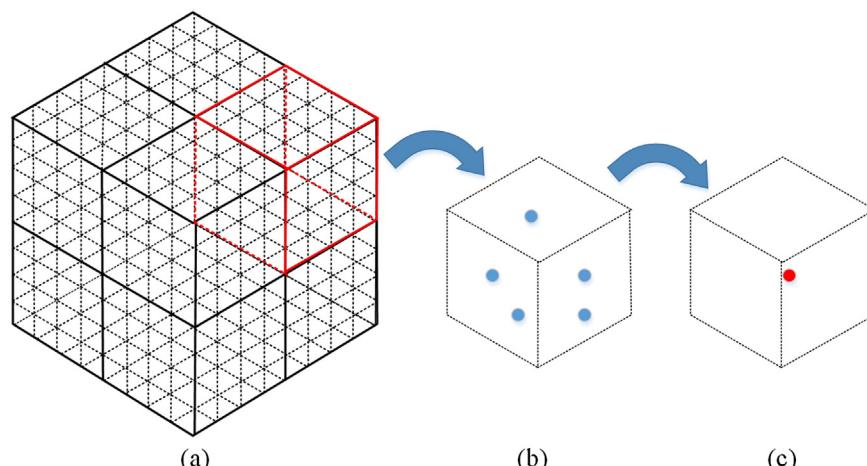


Fig. 3. (a) 3D uniform voxel grid structure; (b) a voxel and the points located in it; and (c) one estimated point left after data downsizing. Illustration adapted from [42].

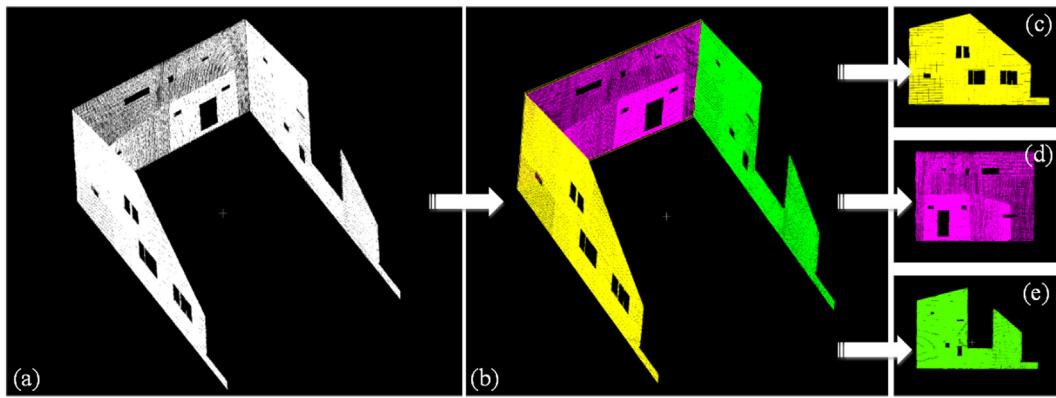


Fig. 4. Segmented point cloud clusters.

scanned piping axis of symmetry; Autodesk Plant 3D® [36] and Kubit PointSense Plant [37] enable user for manually choosing two points from an image of a pipe, and the corresponding 3D point cloud between the two points can be automatically located and modeled; Kubit PointSense Building [37] can automatically generate 2D building plan (wall, floor, ceiling) from 3D laser scanner data, but with manual openings (window, door) creation; and AVEVA Laser Model Interface™ [38], Trimble RealWorks [39] and ClearEdge3D [40] are designed to automatically create 3D model by manually segmenting the point cloud and choose the corresponding catalogs for each segment of point cloud. The abovementioned programs (see Fig. 1) are all semi-automated, and most of them are for industrial application only. Therefore, there is a need for a method of fully automated geometric model creation from point clouds, especially for building envelope modeling which is important to building energy simulation.

Some research efforts have been made on automatic building modeling from point cloud to assist building facility management and performance analysis. Pu and Vosselman [41] proposed a knowledge based method for reconstructing building models from laser scanner data, in which they extract the features and the outline of the building and make the geometric model of the building based on several assumptions because only facades on the street side are scanned. Xiong et al. [9] proposed a context-based modeling algorithm for creating semantic 3D as-is building models of the interior of buildings. Their context-based modeling algorithm was able to identify and model the main visible structural components of an indoor environment, but could not recognize components with irregular shapes that are frequently seen from the exterior of the building envelope. Hinks [33] introduced a point-based voxelization method to automatically transform point cloud data into solid models for computational modeling. This method can directly convert point data into solid models based on volumetric subdivision, however, limits on 2D building façade modeling. It is also difficult to accurately model curved shapes, such as arched windows. Díaz-Vilariño et al. [10] presented a semantic as-built 3D modeling method for shading analysis on buildings. In this method, only wall, ceiling, and floor components can be recognized and segmented. Complete building envelope component recognition is essential for building performance analysis. As a result, automated and efficient extraction of building envelope geometric information is a challenging emerging topic to be solved. In the following sections, we introduce our proposed automatic geometric modeling method and examine its validity by testing three existing buildings.

3. Research methodology

The data presented in this paper are unorganized point clouds collected from a Time of Flight (TOF) laser scanner. These unorganized point cloud data contain only the x, y, z coordinates of each point. Our proposed method comprises four main steps: first, the collected raw

data is pre-processed by removing noise data and downsizing the data. At the completion of data preprocessing, the region growing plane segmentation algorithm is applied to divide the raw data into segments of point cloud which are located on the same plane. Then, a boundary detection algorithm is introduced to recognize boundary points in each segment of point cloud. Further, all the detected boundary points are categorized into their own building component category

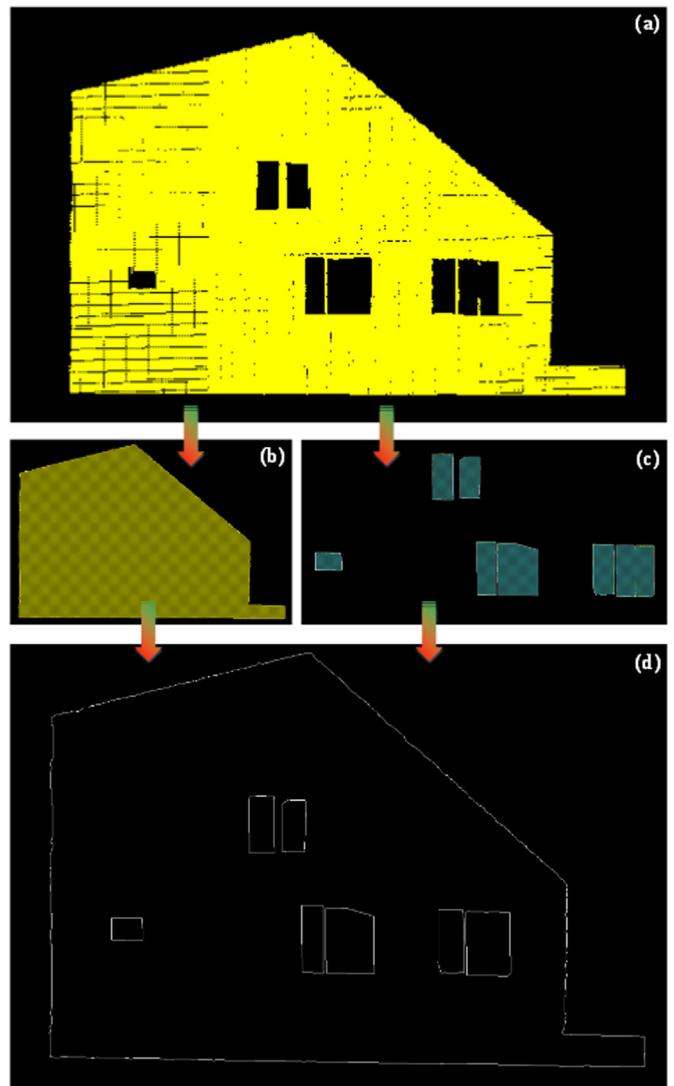


Fig. 5. Outer boundary and inner boundary recognition.

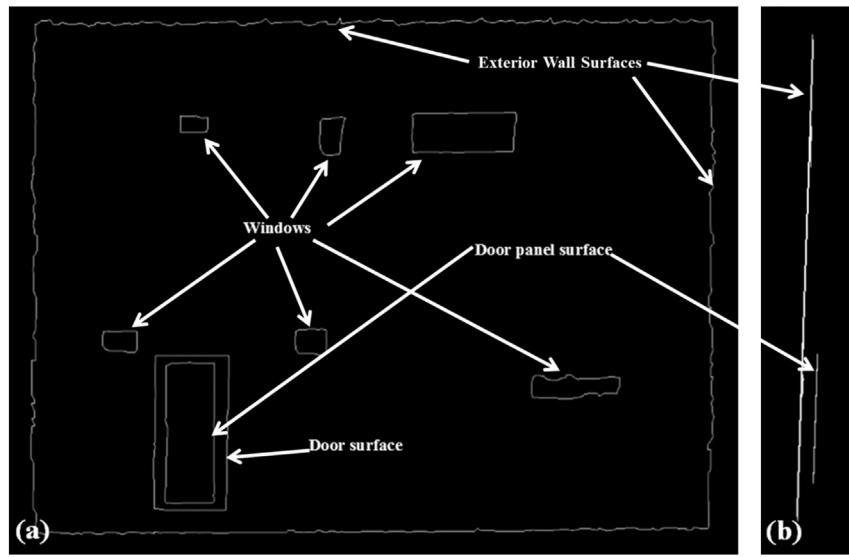


Fig. 6. Exterior wall surface and door panel surface. (a) Front view. (b) Side view.

and building geometry is successfully extracted. Fig. 2 shows the flowchart of our proposed method. The four steps are explained in detail in the ensuing sub-sections.

3.1. Data pre-processing

The point cloud data collected by a laser scanner from an outdoor environment usually contain noise, which can result in a failure or inaccurate result if not been reduced or eliminated. A tensor voting algorithm [51], a non-parametric algorithm that can infer local data geometric structure, was employed in this paper to distinguish and remove the isolated points from the collected point cloud. The standard tensor voting algorithm is composed of two phases: tensor encoding and tensor voting. Information (e.g., curve, surface, intersections) is collected from a tensor through tensor encoding for each 3D point, and in the tensor voting phase a tensor passes its information to its neighbors via a

predefined tensor field. Each neighbor point collects all the votes and then encodes them into a new tensor. Finally, after the surface feature map is generated, noise points can be eliminated by removing those having the lower surface feature values. The tensor voting algorithm considers outlier noise explicitly, but may result in serious problems if the inlier data is also noisy.

Next step is the data downsizing, the goal of which is to increase the data processing speed by reducing the amount of overly dense data being processed. The raw point cloud data are imported into a 3D space where the data structure is a 3D uniform voxel grid (Fig. 3(a)). Each voxel has its own specific boundary according to the size set up. After they are placed in their corresponding voxels, all the points present in the same voxel are removed and a centroid point for the point group is created [42] (Fig. 3(b), (c)). Thus, the bigger the voxel is, the more points are eliminated. The newly downsized data are then passed to the next step as input.

3.2. Region growing plane segmentation

In this paper, existing residential buildings or small commercial buildings are mainly studied. Due to a difficulty of foundation form

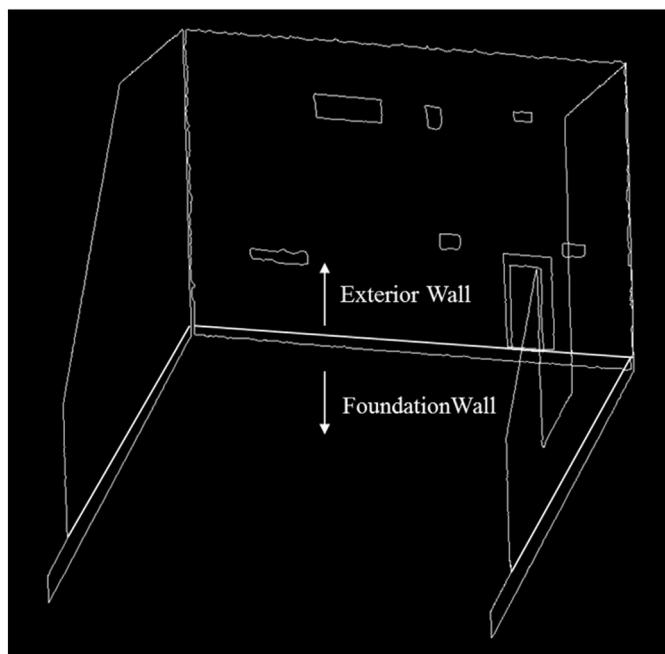


Fig. 7. Exterior wall and foundation wall.

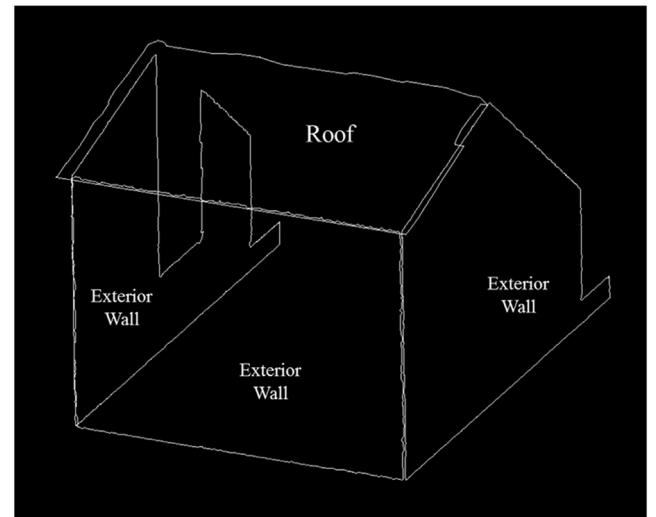


Fig. 8. Roof classification.

Table 1
Proposed classification rules.

Component	Classification rules	
Wall	Exterior wall	Vertical surfaces
	Foundation wall	Vertical surfaces, below a door surface
Door	Panel door	Bottom of the opening close to the boundary of the wall, panel surface behind the opening
	Glass door	Bottom of the opening close to the boundary of the wall, no panel recognized
Window	Blinded window	Non-door opening, blind surface behind the opening
	Clear window	Non-door opening, non-blinded window
Roof	Above and adjacent to the exterior wall, non-vertical	
	Raised floor	Horizontal, below door surface
	Shade	Surface not adjacent to any spaces

design and cost, most of the residential building envelope components have plane surfaces. Thus, a plane segmentation algorithm is then applied to pre-processed data to segment it into a set of disjoint point clouds which are located on the same plane. The region growing plane segmentation algorithm [44,45] was chosen in this research because of its desirable properties, such as conceptually simple and allowing applications in a wide range of settings. This algorithm can merge the points that are close enough to each other in terms of the smoothness constraint into one plane cluster. The algorithm sorts the points by their curvature value, and the region begins its growth from point P with a minimum curvature value. This point P is chosen and added to the set called seed points. For each seed point chosen, the algorithm finds its neighbor points $\{P_N\}$ and tests each neighbor point $N \in \{P_N\}$ for the angle between its normal and the normal of the current seed point. The current seed point is added to the current region if the angle is less than the threshold value θ_{th} . Further, the curvature value of its neighbor point is compared with the value of the seed point. If the curvature value is less than the threshold value C_{th} , this neighbor point is added to the set of seed points and the current tested seed point is removed from the set. The algorithm repeats this process until the set of seed points is empty, signifying that the algorithm has grown the entire region and all points have been labeled. The output of this segmentation algorithm is a set of segmented point cloud clusters, where points in the same cluster are considered to be part of the same plane (see Fig. 4).

3.3. Edge and boundary point extraction

Point cloud data cannot be collected from materials that have low reflectivity, such as black objects and glass, owing to the characteristics of the laser beam. Consequently, there is no point showing in the window glass area. The edge points of the window frames can be separated from

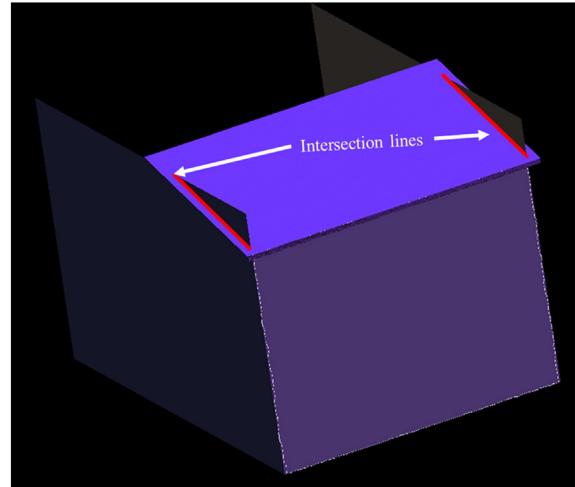


Fig. 10. Intersection lines extracted after surface extension.

the joined boundary points on the basis that the boundary points of the window frame surround an empty window glass area. In the third step, an edge and boundary detection algorithm [43,46] is applied to isolate edge and boundary points from the rest. The results of the region growing plane segmentation process are a set of segmented point cloud clusters, in which each point contains x , y , z coordinates together with its normal and curvature flatness. As illustrated in [46], the edges of the objects can be extracted based on the curvature information because they are characterized by high changes in curvature. However, the boundary points residing on the outer border of the point cloud cannot be found based on curvature data as there is no change for these points. Since all points in each cluster are on the same plane, we can project the point cloud onto a 2D plane. In 2D plane, the boundary points can be easily identified because the maximal angle formed by the vectors towards the neighboring points is larger for boundary points than for points that are on the inside of the object. For point cloud data of buildings, the edge points and boundary points are correspondingly referred to the edge of the openings and the boundaries of walls or roofs (see Fig. 5). On the edge and boundary points of all clusters being recognized, all the component surfaces can be created by applying 2D concave hull algorithm [52].

3.4. Rule-based building envelope component classification

In the final step of our proposed method, the building envelope components were automatically identified via the boundary points obtained

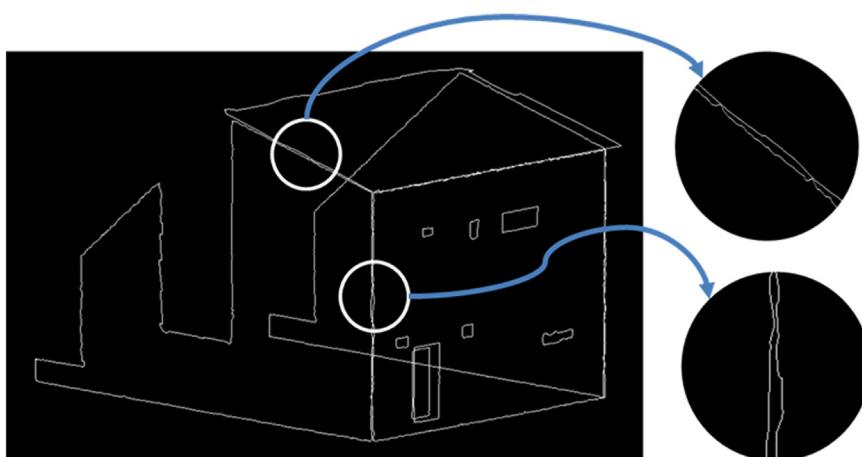


Fig. 9. Gaps between surfaces.

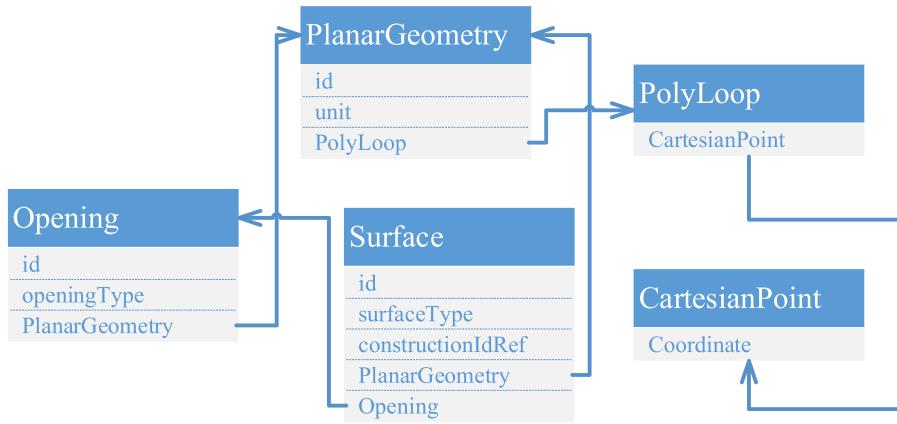


Fig. 11. The gbXML schema of the elements used in data exchange.

in the previous sub-section. All surfaces recognized from the previous step were processed through a rule-based classification system. The following rules were developed based on common knowledge, and only the building components covered in gbXML schema were considered in this research to be an object to recognize. We first defined all vertical surfaces as wall components, then separated doors and windows from the recognized openings of all wall components. In this paper, it was assumed that all openings were closed when the data was collected. For each opening, if there was a same size of surface parallel and close to it, then this paralleled surface can either be a door panel or window-blinds. Together with the location of the opening, the opening was labeled as a door if it was close to the bottom boundary of its wall surface, otherwise it was recognized as a window. The door components were further categorized into normal door and glass door, according to the existence of the door panel. The window components were also categorized into clear window and blinded window based on the existence of a window-blinds. Fig. 6 shows an example of the recognized wall, window, door, and door panel surfaces. Then, the wall category was divided into two classes (exterior wall and foundation wall) by the rule that the foundation wall surface was below a door surface, and the exterior wall

surface was not (Fig. 7). The partial foundation wall surface could also be completed according to the user input. Because the roof was usually above the walls and adjacent to at least one exterior wall, it can be recognized once the exterior wall components are defined (Fig. 8). Lastly, the unclassified surfaces were categorized into the raised floor and shade based on the rules that raised floor surface was horizontal and below a door surface, and the shade surface was not adjacent to the space formed by wall surfaces. Table 1 shows the organized classification rules.

3.5. Geometry size fitting

After categorizing the recognized building envelope components, a rough semantic building geometry model is created. The created model is not yet watertight due to the gaps existing between the component surfaces. There are several possible causes that may lead to the existence of the gaps. First, the limited scan resolution of the laser scanner inevitably generates the gaps between points either vertically or horizontally. The size of the gap can vary based on the scanning location, and the bigger it gets the further the scanner scans from. This gap in the

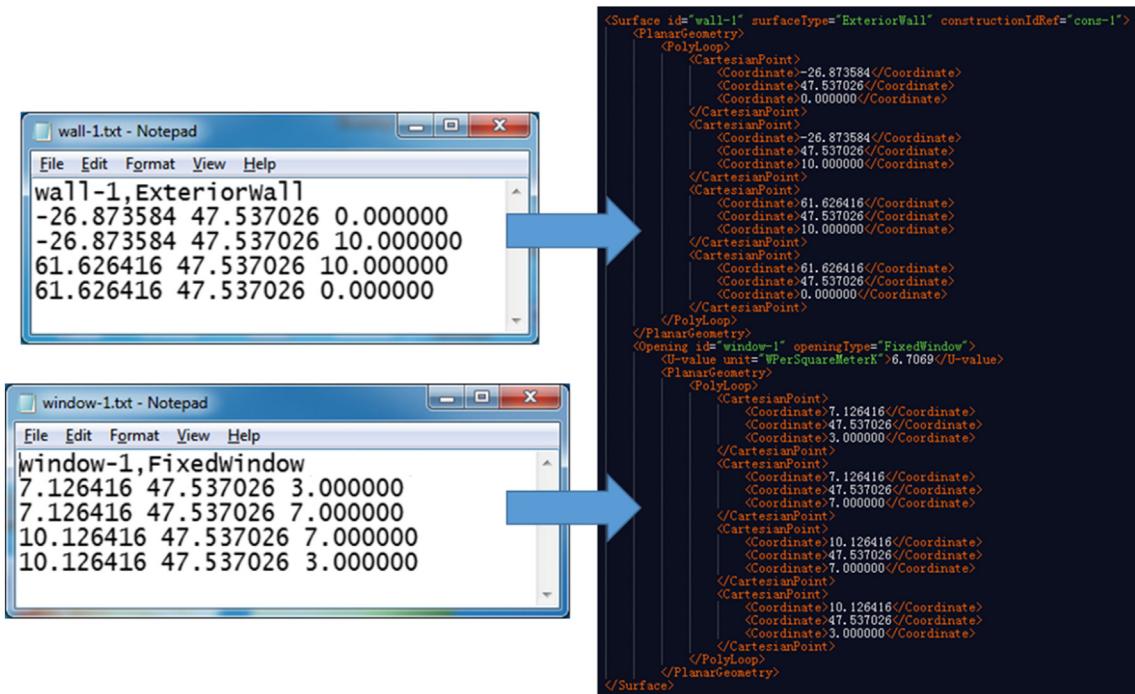


Fig. 12. Data exchange from text data (left) to gbXML data (right).

point cloud can make the collected as-is building data miss some features such as the edge of the components, and this will finally result in a surface gap after the proposed recognition and classification algorithms. Second, the roof thickness cannot be captured by the laser scanner due to its characteristics of only scanning the exterior surface of the object. In the raw data, the top surface of the roof is scanned which does

not contact the top of the exterior wall. Another possible reason is the algorithmic accuracy. For example, the data downsizing will result in a gap increase between the points although reducing the required data processing time; in the point cloud segmentation step, each point only belongs to one point cloud cluster, and this will sow the seed of creating the gaps in the following edge detection phase. In Fig. 9, it can be seen

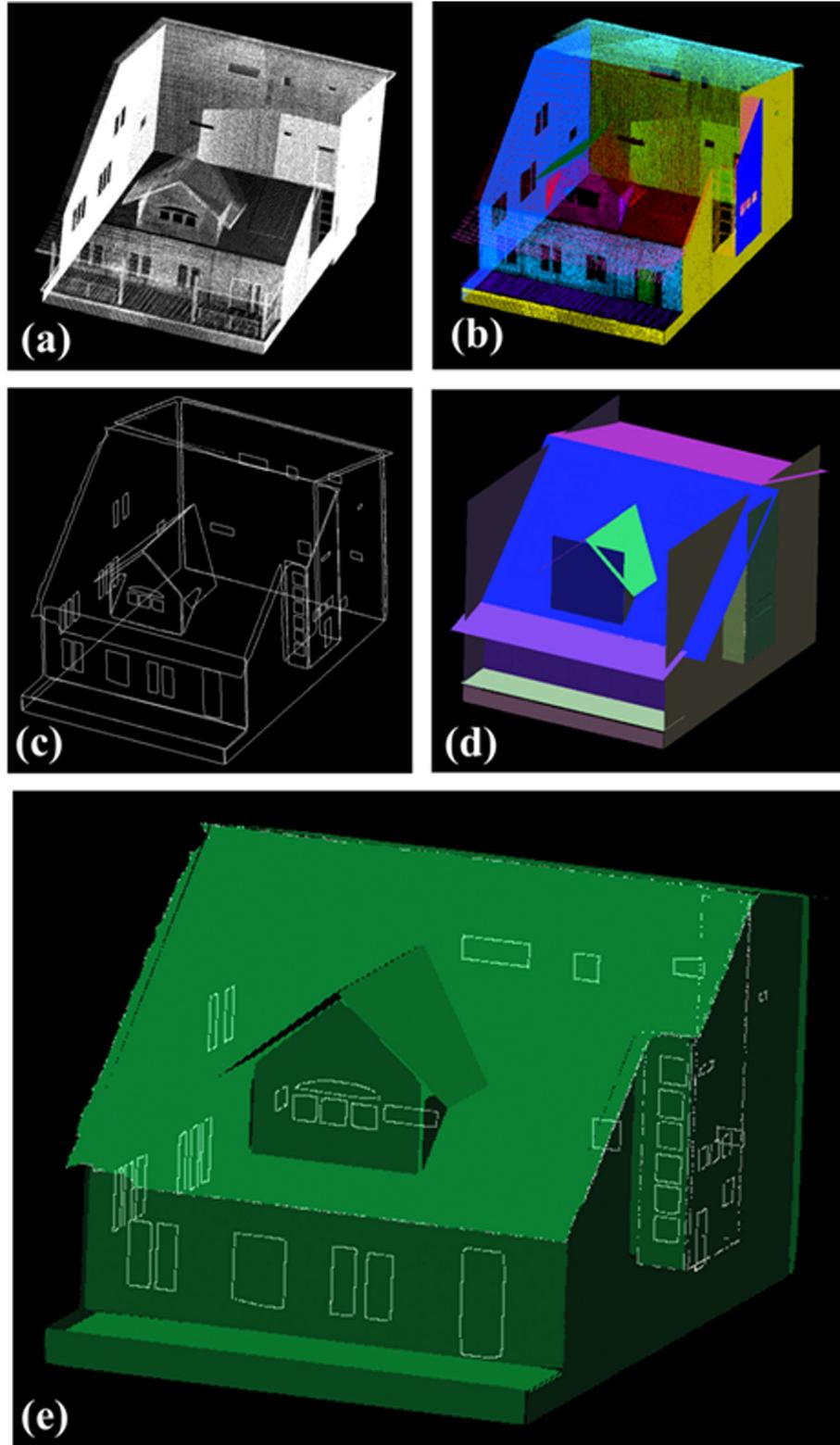


Fig. 13. Test results of case study 1 (ZNETH). (a) Raw data; (b) segmented point cloud clusters; (c) detected boundaries and edges; (d) geometry size fitting; and (e) created semantic model.

Table 2

Evaluation of the extracted envelope components for case study 1.

	TP	FP	FN	TN	Precision	Recall	Accuracy
Exterior wall	10	0	0	46	100.00%	100.00%	100.00%
Window	39	0	1	17	100.00%	97.50%	98.25%
Door	2	0	0	54	100.00%	100.00%	100.00%
Foundation wall	1	0	0	55	100.00%	100.00%	100.00%
Raised floor	1	0	0	55	100.00%	100.00%	100.00%
Roof	4	0	0	52	100.00%	100.00%	100.00%

(Precision = TP / (TP + FP), recall = TP / (TP + FN), accuracy = (TP + TN) / (TP + TN + FP + FN)).

Table 3

Comparison between the recognized and the manually measured envelope components for case study 1.

	Measured dimension (m ²)	Recognized dimension (m ²)	Error (m ²)	Error (%)
Exterior wall	355.25	363.95	8.71	2.45
Door	3.90	4.27	0.37	9.49
Window	18.48	15.29	3.19	17.26
Roof	156.74	143.48	13.26	8.46
Raised floor	19.74	16.41	3.33	16.87

that there are gaps between the recognized surfaces. Since the objective of this research is to create a model ready for energy simulation which requires a closed space as an input, there is a need for applying a geometry size fitting algorithm to fill in the gaps and refine the created semantic model. The proposed algorithm simply extends the surfaces of all walls, roofs, and raised floor both vertically and horizontally, and replaces their surface edges with the intersection lines created by any two extended surfaces (Fig. 10).

3.6. Data conversion

The output of the building component classification algorithm was a set of boundary points of the envelope components. For each individual component, all its boundary points were saved in a text file in which the first line of data was its surface ID, and followed by its surface type on the same line. Starting from the second line, there were three columns of data on each line, and they represented one point's x, y, and z coordinates. To be useful for energy simulation, the file has to be converted to another file format that can be imported. In this research, the gbXML open schema was chosen to help facilitate the transfer of the data to engineering analysis tools. Fig. 11 shows a structure chart of element "Surface" in gbXML schema (version 5.0.1). This element was used to

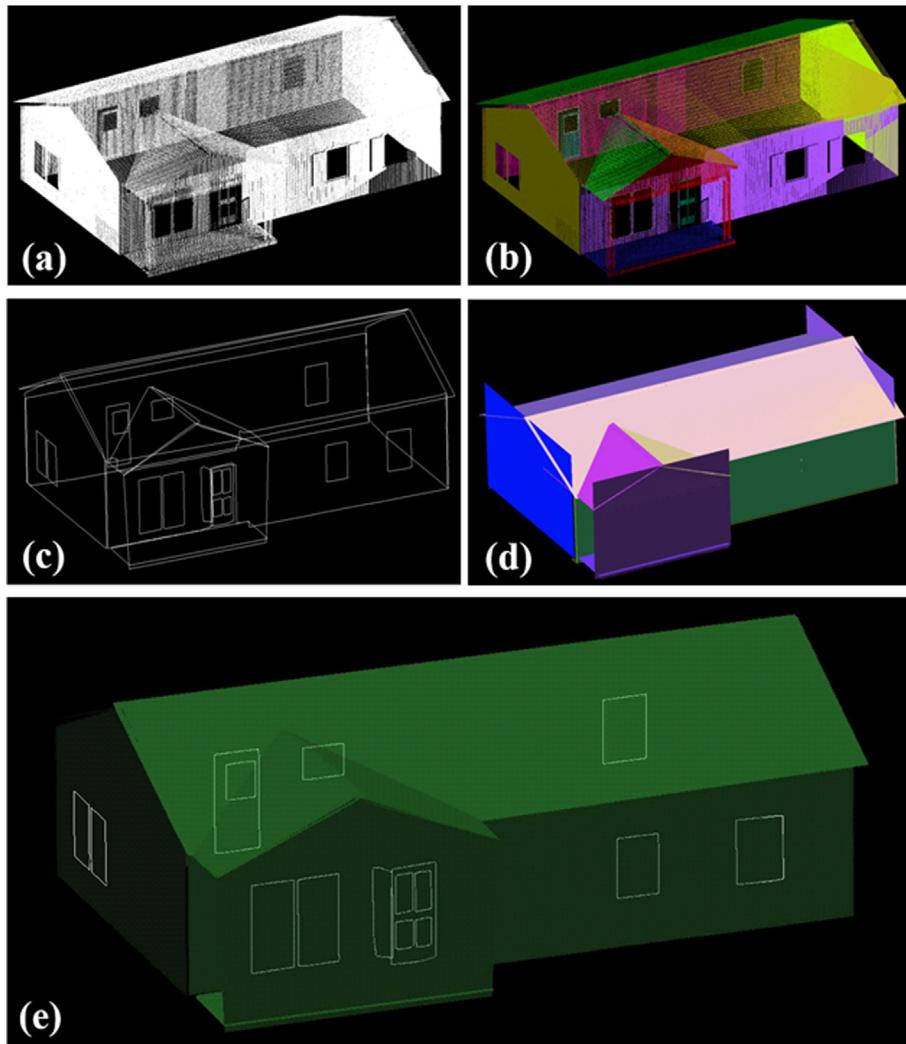


Fig. 14. Test results of case study 2 (ZNETH II). (a) Raw data; (b) segmented point cloud clusters; (c) detected boundaries and edges; (d) geometry size fitting; and (e) created semantic model.

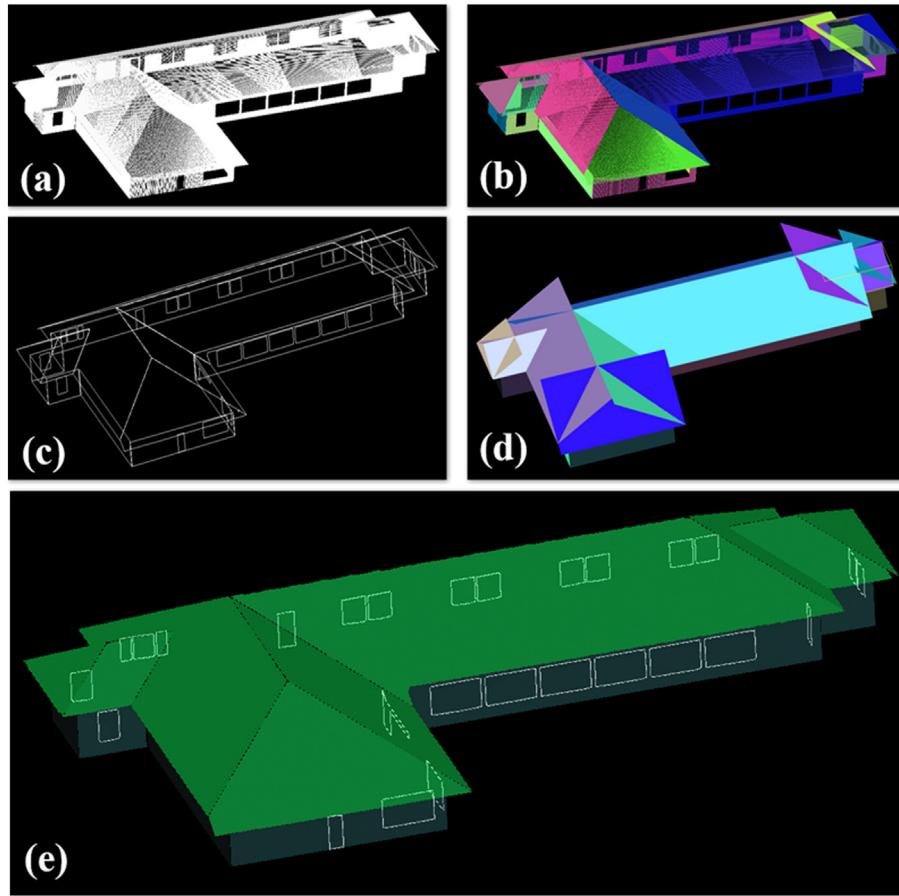


Fig. 15. Test results of case study 3 (Bank). (a) Raw data; (b) segmented point cloud clusters; (c) detected boundaries and edges; (d) geometry size fitting; and (e) created semantic model.

interpret the extracted components. Each surface requires a unique ID, surface type, and geometry. Surface type includes interior wall, exterior wall, roof, ceiling, and etc. In this paper, exterior wall and roof were assigned to corresponding surface. PlanarGeometry specifies the location of the surface, and lists all vertexes of the surface to define a loop. Attribute “Opening” is added if there is any opening in the surface. The extracted as-is data were first saved as text files, as shown in Fig. 12, and then were converted into a gbXML file according to the corresponding gbXML schema.

4. Implementation and test results

Validation of the proposed methodology was implemented on three small existing buildings, and two of which are residential houses, and one is a small bank building. In all case studies, the point clouds of the buildings were as completely as possible collected and used as raw input data. In the first case study, a Zero Net Energy Test House (ZNETH) was used as a test subject. The collected raw data (Fig. 13(a)) containing 1,061,637 points were first processed by the data downsizing

algorithm. In the algorithm, the leaf size of the vessel was set at 0.05 m which is five times of the resolutions (0.01 m) of the raw data. By utilizing data sizing algorithm, the data size was decreased to 541,003 points which is about half size of the raw data. The decreased data size can significantly reduce the processing time in the following processes. Then, the downsized point cloud data were segmented into a set of plane clusters (Fig. 13(b)). For each segmented point cloud cluster, the inner and outer boundary points were extracted by a boundary and edge point detection algorithm.

The output of the boundary point detection algorithm was a set of outer and inner boundary surfaces. Then, the rule-based building envelope component classification algorithm followed to categorize each boundary surface into its corresponding category. Fig. 13(c) shows the results of the proposed method. There were a total 2 door components, 39 window components, 4 roof components, 1 underground wall component, 1 raised floor component, and 10 exterior wall components being recognized from the set of boundary surfaces. Precision, recall and accuracy [47] were measured to evaluate the performance of the component classification (Table 2). In this case study, all recognized

Table 4

Evaluation of the extracted envelope components for case study 2.

	TP	FP	FN	TN	Precision	Recall	Accuracy
Exterior wall	4	1	0	20	80%	100%	96%
Window	14	0	0	11	100%	100%	100%
Door	2	1	0	22	67%	100%	96%
Roof	4	0	0	21	100%	100%	100%
Raised floor	1	0	0	24	100%	100%	100%

(Precision = TP / (TP + FP), recall = TP / (TP + FN), accuracy = (TP + TN) / (TP + TN + FP + FN)).

Table 5

Comparison between the recognized and the manually measured envelope components for case study 2.

	Measured dimension (m ²)	Recognized dimension (m ²)	Error (m ²)	Error (%)
Exterior wall	127.30	128.75	1.45	1.14
Door	2.97	3.89	0.92	3.10
Window	10.81	11.83	1.02	9.44
Roof	137.50	148.60	11.10	8.07
Raised floor	10.41	10.19	0.21	2.02

Table 6

Evaluation of the extracted envelope components for case study 3.

	TP	FP	FN	TN	Precision	Recall	Accuracy
Exterior wall	14	0	0	42	100%	100%	100%
Window	27	0	3	26	100%	90%	95%
Door	3	0	0	53	100%	100%	100%
Roof	12	0	0	44	100%	100%	100%

(Precision = $TP / (TP + FP)$, recall = $TP / (TP + FN)$, accuracy = $(TP + TN) / (TP + TN + FP + FN)$).

components except one window were correctly categorized. The area dimensions of the recognized components were also compared with the manually measured area dimensions of the house, and the absolute difference was calculated for each recognized component. Table 3 shows the comparison results of the recognized geometry of each envelope component. The door category was the most accurately recognized in terms of the area size. The roof and exterior wall categories have a lower accuracy because of the incompleteness of the raw data.

To further validate the robustness of the proposed methodology, two more case studies were conducted. As shown in Figs. 14(a) and 15(a), one is a one-story residential house, the other is a one-story bank building. Following the same process, the results of these two case studies were correspondingly visualized in Figs. 14 and 15. The evaluation results of the component recognition were also shown in Tables 4–7. In case study 2, one exterior wall and one door were falsely classified, and most components were recognized with around 1 m^2 error. There was 11.10 m^2 difference between the measured and the recognized roof components. This is also caused by the data incompleteness. In case study 3, the tested bank building has a more complicated roof containing 12 plane segments. Based on the evaluation results shown in Table 6, 3 out of the 24 windows were not successfully recognized from the point cloud data. The dimension evaluation results in Table 7 show that the recognized exterior wall and roof categories had greater absolute area difference compared with the manually measured one.

In Fig. 16, all recognized component categories in three case studies were put together to analyze the relationship between the error and the measured area size of the component. It can be summarized from Fig. 16 that the greater errors mostly came from the greater size of the component. Fig. 17 shows the error range frequency, and a total of 50% of the recognized component categories had less than 2.5 m^2 error, and a total of about 71% had less than 10 m^2 error. Fig. 17 shows the error range frequency, and a total of 50% of the recognized component categories had less than 2.5 m^2 error, and a total of about 71% had less than 10 m^2 error. Through a joint analysis with Tables 3, 5 and 7, it can be seen that the recognized component categories with greater than 10 m^2 errors are roof and exterior wall. This is because the point cloud data are usually difficult to be completely collected from these two components due to the building height or occlusion. Besides the absolute error, the relative error percentage was also calculated to evaluate the performance of the proposed method because a small absolute error of a small component can possibly yield a bigger relative error percentage. For example in the window components in case studies 1 and 2, the absolute errors are 3.19 m^2 and 1.02 m^2 correspondingly, but

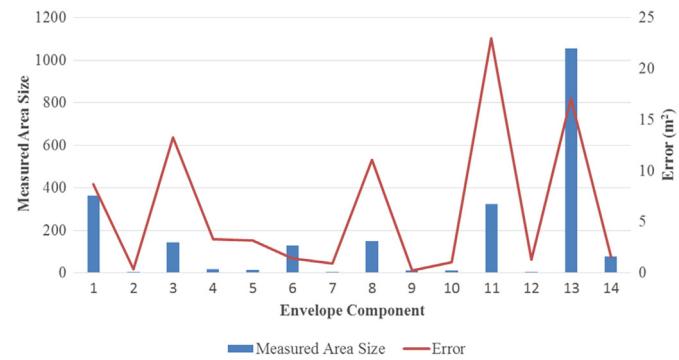


Fig. 16. Summary of the relationship between the error and the measured area size.

the relative error percentages are both the highest among all their corresponding components. There are two causes that may lead to these results: First, the edges of the windows are defined by recognizing the boundary points of the empty areas. Since the point cloud cannot be collected from the window glass, the boundary points collected are usually from the window frames. Therefore, using the window frames as the edges of the windows slightly increased the errors. Second, the component recognition precision also has an effect on the corresponding error percentage. In Table 2, it shows that there is one window not being successfully recognized ($FN = 1$), and this will result in an error increase because, in Table 3, the measured dimension includes both recognized and unrecognized components.

5. Feasibility validation

In previous sections, this paper discussed about how to automatically extract building envelope geometry from the point cloud data. The output from the previous sections was an auto-generated gbXML file. The intent of this section was to validate the feasibility of using the auto-generated gbXML file as an input in the energy simulation tools. Fig. 18 shows the preliminary result that the auto-generated gbXML file of the case study 1 was successfully imported into a building energy simulation tool (Autodesk Ecotect Analysis 2011 was tested for validation in this study).

6. Conclusions

In this paper, we proposed and demonstrated a method for automatic building geometry extraction from unorganized point clouds collected from a 3D laser scanner. In the proposed method, raw data are first eliminated to reduce the data size so as to increase the processing speed while maintaining accuracy. The downsized data are then processed through boundary detection algorithms, and building components



Fig. 17. Error range frequency.

Table 7

Comparison between the recognized and the manually measured envelope components for case study 3.

	Measured dimension (m ²)	Recognized dimension (m ²)	Error (m ²)	Error (%)
Exterior wall	347.70	324.74	22.97	6.61
Door	4.63	5.92	1.29	27.86
Window	76.01	77.53	1.52	2.00
Roof	1036.90	1054.00	17.11	1.65

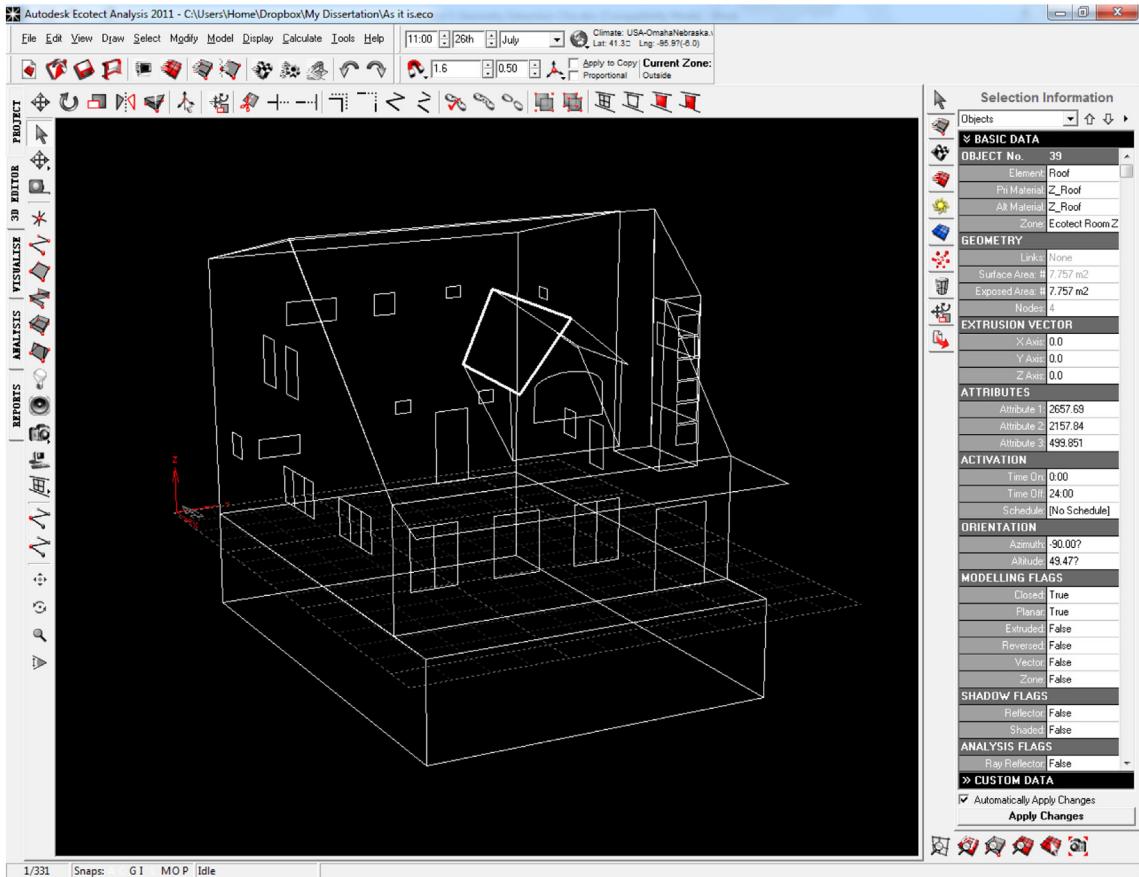


Fig. 18. Auto-generated gbXML file imported into Autodesk Ecotect.

finally recognized by processing the boundary points. We tested and validated our proposed method on three collected as-is building data. The test results show that the proposed method can successfully extract semantic information from the raw point cloud data, and convert the extracted data into a gbXML format that can be imported into the energy simulation tools. Precision, recall, accuracy of the component recognition algorithm, and dimension error of each component were all evaluated in this paper.

In future work, we plan to focus on improving the accuracy of processing incomplete data because it was identified that accuracy primarily relies on the integrity of the data. The data downsizing process can cause errors because it replaces the points in each voxel with an estimated point. Incomplete data is another factor that can reduce accuracy (e.g., the incomplete roof of the residential house and the parts blocked by trees). Therefore, how to complete the data and filter the unrelated data will also be an emerging topic. We also plan to further enhance the robustness of our proposed method so that it can be implemented and used for several different types of object recognition and extraction activities for as-built modeling in the AEC/FM domain.

Acknowledgment

Portions of this work are based on work supported by the National Science Foundation (award #: CMMI-1055788). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

References

- [1] M. Golparvar-Fard, J. Bohn, J. Teizer, S. Savarese, F. Peña-Mora, Evaluation of image-based modeling and laser scanning accuracy for emerging automated performance monitoring techniques, *Autom. Constr.* 20 (2011) 1143–1155.
- [2] Y. Cho, C. Haas, K. Liapi, S. Sreenivasan, A framework for rapid local area modeling for construction automation, *Autom. Constr.* 11 (6) (2002) 629–641.
- [3] B. Akinci, F. Boukamp, C. Gordon, D. Huber, C. Lyons, K. Park, A formalism for utilization of sensor systems and integrated project models for active construction quality control, *Autom. Constr.* 15 (2) (2006) 124–138.
- [4] L. Brass, A glimpse of the energy future, *Oak Ridge Natl. Lab. Rev.* 40 (2) (2007) 2–7.
- [5] F. Bosche, C.T. Haas, Automated retrieval of 3D CAD model objects in construction range images, *Autom. Constr.* 17 (4) (2008) 499–512.
- [6] P. Tang, B. Akinci, Automatic execution of workflows on laser-scanned data for extracting bridge surveying goals, *Adv. Eng. Inform.* 26 (4) (2012) 889–903.
- [7] U.S. Department of Energy, 2011 U.S. DOE Buildings Energy Databook, <http://buildingsdatabook.eren.doe.gov/>, Apr. 15 2012.
- [8] Energy Information Agency (EIA), Annual Energy Review 2008, DOE/EIA-0384 (2008), U.S. Department of Energy, June 2009.
- [9] X. Xiong, A. Adan, B. Akinci, D. Huber, Automatic creation of semantically rich 3D building models from laser scanner data, *Autom. Constr.* 31 (2013) 325–337.
- [10] L. Diaz-Vilarin, S. Laguela, J. Arnesto, P. Arias, Semantic as-built 3D models including shades for the evaluation of solar influence on buildings, *Sol. Energy* 92 (2013) 269–279.
- [11] M. Golparvar-Fard, F. Peña-Mora, S. Savarese, D4AR— a 4-dimensional augmented reality model for automating construction progress data collection, processing and communication, *J. Inf. Technol. Constr.* (ITcon) 14 (2009) 129–153 (Special Issue Next Generation Construction IT: Technology Foresight, Future Studies, Roadmapping, and Scenario Planning, <http://www.itcon.org/2009/13>).
- [12] M. Golparvar-Fard, F. Peña-Mora, C. Arboleda, S. Lee, Visualization of construction progress monitoring with 4D simulation model overlaid on time-lapsed photographs, *ASCE J. Comput. Civ. Eng.* 23 (4) (2009) 391–404 (Special Ed. on Graphical 3D Visualization in AEC).
- [13] J. Bohn, J. Teizer, Benefits and barriers of construction project monitoring using high-resolution automated cameras, *J. Constr. Eng. Manag.* 136 (6) (2010) 632–640.
- [14] E.B. Anil, P. Tang, B. Akinci, D. Huber, Deviation analysis method for the assessment of the quality of the as-is Building Information Models generated from point cloud data, *Autom. Constr.* 35 (2013) 507–516.
- [15] S.W. Kwon, F. Bosche, C. Kim, C.T. Haas, K.A. Liapi, Fitting range data to primitives for rapid local 3D modelling using sparse range point clouds, *Autom. Constr.* 13 (1) (2004) 67–81.
- [16] A. Bhatia, S. Choe, O. Fierro, F. Leite, Evaluation of accuracy of as-built 3D modeling from photos taken by handheld digital cameras, *Autom. Constr.* 28 (2012) 116–127.
- [17] F. Bosche, C.T. Haas, Towards automated comparison of 3D sensed and 3D designed data, International Workshop on Computing in Civil Engineering, Pittsburgh, PA, USA 2007, pp. 548–556.

- [18] S. El-Omari, O. Moselhi, Integrating 3D laser scanning and photogrammetry for progress measurement of construction work, *J. Autom. Constr.* 18 (1) (2008) 1–9.
- [19] D. Rebolj, N.C. Babic, A. Magdic, P. Podbreznik, M. Psunder, Automated construction activity monitoring system, *Adv. Eng. Inform.* 22 (4) (2008) 493–503.
- [20] C. Gordon, B. Akinci, Technology and process assessment of using LADAR and embedded sensing for construction quality control, *Construction Research Congress*, San Diego, CA, USA, 2005.
- [21] G.S. Cheok, W.C. Stone, R.R. Lipman, C. Witzgall, Ladars for construction assessment and update, *Autom. Constr.* 9 (5–6) (2000) 463–477.
- [22] I. Heinz, F. Hartl, C. Frohlich, Semi-automatic 3D CAD model generation of as-built conditions of real environments using a visual laser radar, *Robot and Human Interactive Communication, Proceedings. 10th IEEE International Workshop on 2001*, pp. 400–406.
- [23] C. Kim, C.T. Haas, K.A. Liapi, Rapid, on-site spatial information acquisition and its use for infrastructure operation and maintenance, *Autom. Constr.* 14 (5) (2005) 666–684.
- [24] E.B. Anil, P. Tang, B. Akinci, D. Huber, Assessment of quality of as-is building information models generated from point clouds using deviation analysis, *Proceedings of SPIE*, San Jose, California, USA, 2011.
- [25] A. Adan, D. Huber, 3D reconstruction of interior wall surfaces under occlusion and clutter, *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, Hangzhou, China, 2011.
- [26] S.J. Gordon, D.D. Lichti, M.P. Stewart, J. Franke, Modeling point clouds for precise structural deformation measurement, *Proceedings of the XXth ISPRS Congress, Istanbul, Turkey 2004*, pp. 954–959.
- [27] E.J. Jaselskis, Z. Gao, R.C. Walters, Improving transportation projects using laser scanning, *J. Constr. Eng. Manag.* 131 (3) (2005) 377–384.
- [28] E.J. Jaselskis, E.T. Cackler, R.C. Walters, J. Zhang, M. Kaewmoracharoen, Using Scanning Lasers for Real-time Pavement Thickness Measurement, CTRE Project 05-205, Iowa State University, Ames, IA, USA, 2006.
- [29] H. Son, C. Kim, C. Kim, 3D reconstruction of as-built industrial instrumentation models from laser-scan data and a 3D CAD database based on prior knowledge, *Autom. Constr.* 49, Part B (2015) 193–200.
- [30] P. Tang, B. Akinci, Automated measurement extraction from laser scanned point clouds to support bridge inspection, *IABSE Symposium*, Weimar, Germany, 2008.
- [31] P. Tang, B. Akinci, H. James, J. Garrett, Laser scanning for bridge inspection and management, *IABSE Symposium*, Weimar, Germany 2007, pp. 206–207.
- [32] P. Tang, D. Huber, B. Akinci, Quantification of edge loss of laser scanned data at spatial discontinuities, *J. Comput. Civ. Eng.* 25 (1) (2011) 31–42.
- [33] T. Hinks, Geometric Processing Techniques for Urban Aerial Laser Scan Data Doctoral Dissertation University College Dublin, Dublin, Ireland, 2013.
- [34] Leica Geosystems, http://www.leica-geosystems.us/en/Leica-CloudWorx_60696.htm, Jan. 27 2014.
- [35] Intergraph, http://www.intergraph.com/products/ppm/smart_3d/plant/default.aspx, Jan. 27 2014.
- [36] Autodesk, <http://www.autodesk.com/products/autodesk-autocad-plant-3d/overview>, Jan. 27 2014.
- [37] Kubit, <http://us.kubit-software.com/CAD/Products/PointSense/index.php>, Jan. 27 2014.
- [38] AVEVA Continual Progression, http://www.aveva.com/en/Products_and_Services/Product_Finder.aspx#open:0F6798A9-0B3A-4E72-88C3-4BEC11228F5E, Jan. 27 2014.
- [39] Trimble, <http://www.trimble.com/3d-laser-scanning/realworks.aspx?dtID=overview&>, Jan. 27 2014.
- [40] ClearEdge3D, <http://www.clearedge3d.com/>, Jan. 27 2014.
- [41] S. Pu, G. Vosselman, Knowledge based reconstruction of building models from terrestrial laser scanning data, *ISPRS J. Photogramm. Remote Sens.* 64 (2009) 575–584.
- [42] H. Moravec, Robot Spatial Perception by Stereoscopic Vision and 3D Evidence Grids, 1996.
- [43] R. Rusu, N. Blodow, Z. Marton, A. Soos, M. Beetz, Towards 3D Object Maps for Autonomous Household Robots, *International Conference on Intelligent Robots and Systems*, IEEE, San Diego, CA, Oct. 29–Nov. 2 2007, pp. 319–3198.
- [44] R. Farid, C. Sammut, A relational approach to plane-based object categorization, *RSS 2012 Workshop on RGB-D Cameras*, Jul. 2012 ([Online], Available: <http://www.cs.washington.edu/ai/MobileRobotics/rbgd-workshop-2012/papers/farid-rbgd12-object-categorization.pdf>).
- [45] R. Farid, C. Sammut, Plane-based object categorization using relational learning, *Mach. Learn.* 94 (1) (2014) 1–21.
- [46] K.-H. Bae, D.D. Lichti, Automated registration of unorganized point clouds from terrestrial laser scanners, *International Archives of Photogrammetry and Remote Sensing (IAPRS)2004*, 222–227.
- [47] D.L. Olson, D. Delen, *Advanced Data Mining Techniques*, 1st edition, 2008. 138 ISBN 3540769161.
- [48] V. Bazjanac, Space boundary requirements for modeling of building geometry for energy and other performance simulation, *Proceedings of the CIB W78 2010: 27th International Conference*, Cairo, Egypt, 2010.
- [49] I. Howell, B. Batcheler, Building information modeling two years later – huge potential, Some Success and Several Limitations, *Newforma White Paper2004*.
- [50] V. Bazjanac, Implementation of semi-automated energy performance simulation: building geometry, *CIB W78*, Proc. 26th Conf., Managing IT in Construction. Istanbul, TK, CRC Press, ISBN: 978-0-415-56744-2 2009, pp. 595–602.
- [51] C. Kim, H. Son, C. Kim, Fully automated registration of 3D data to a 3D CAD model for project progress monitoring, *Autom. Constr.* 35 (2013) 587–594.
- [52] P. Zhou, *Computational Geometry: Analysis and Design on the Algorithms*, Second edition Tsinghua University Press, Beijing, 2005.
- [53] C. Wang, Y. Cho, Automated 3D building envelope recognition from point clouds for energy analysis, *Construction Research Congress*, West Lafayette, IN 2012, pp. 1155–1164.