LiDAR Point Clouds to 3-D Urban Models: A Review

Ruisheng Wang, Jiju Peethambaran, and Dong Chen

***Abstract*—Three-dimensional (3-D) urban models are an integral part of numerous applications, such as urban planning and performance simulation, mapping and visualization, emergency response training and entertainment, among others. We consolidate various algorithms proposed for reconstructing 3-D models of urban objects from point clouds. Urban models addressed in this review include buildings, vegetation, utilities such as roads or power lines and free-form architectures such as curved buildings or statues, all of which are ubiquitous in a typical urban scenario. While urban modeling, building reconstruction, in particular, clearly demand specific traits in the models, such as regularity, symmetry, and repetition; most of the traditional and state-of-the-art 3-D reconstruction algorithms are designed to address very generic objects of arbitrary shapes and topology. The recent efforts in the urban reconstruction arena, however, strive to accommodate the various pressing needs of urban modeling. Strategically, urban modeling research nowadays focuses on the usage of specialized priors, such as global regularity, Manhattan-geometry or symmetry to aid the reconstruction, or efficient adaptation of existing reconstruction techniques to the urban modeling pipeline. Aimed at an in-depth exploration of further possibilities, we review the existing urban reconstruction algorithms, prevalent in computer graphics, computer vision and photogrammetry disciplines, evaluate their performance in the architectural modeling context, and discuss the adaptability of generic mesh reconstruction techniques to the urban modeling pipeline. In the end, we suggest a few directions of research that may be adopted to close in the technology gaps.**

***Index Terms*—Airborne laser scanning (ALS), airborne light detection and ranging, façade modeling, light detection and ranging (LiDAR), mobile laser scanning (MLS), mobile LiDAR, point clouds, powerline (PL) reconstruction, road modeling, rooftop modeling, surface reconstruction, terrestrial laser scanning (TLS), tree modeling, urban models.**

Manuscript received April 8, 2017; revised July 12, 2017 and November 22, 2017; accepted December 4, 2017. Date of publication January 9, 2018; date of current version February 12, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 41301521, in part by the Open Funds of the State Key Laboratory of Remote Sensing Science under Grant OFSLRSS201516, in part by the China Scholarship Council under Grant 201506710075, and in part by the Eyes-High Postdoctoral fellowship at the University of Calgary. *(R. Wang and J. Peethambaran are co-first authors.) (Corresponding author: Ruisheng Wang.)*

R. Wang is with Beijing Advanced Innovation Center for Imaging Technology, Capital Normal University, Beijing 100000, China, and also with the Department of Geomatics Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada (e-mail: ruiswang@ucalgary.ca).

J. Peethambaran is with the Department of Geomatics Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada (e-mail: jijupnair2000@gmail.com).

D. Chen is with the Nanjing Forestry University, Nanjing 210037, China, and also with the Department of Geomatics Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada (e-mail: chendong@njfu.edu.cn; dong.chen@ucalgary.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTARS.2017.2781132

I. Introduction

THE advances in depth-based scanning technologies and the large-scale availability of point clouds of urban scenes have triggered an ever-increasing demand for accurate and efficient reconstruction of urban models from these acquired data. Over the past few decades, point clouds from light detection and ranging (LiDAR) have been a major data source for mapping applications in the photogrammetry and remote sensing communities. In recent years, processing large-scale geospatial data, especially point clouds, has also drawn considerable attention from the computer vision, computer graphics, and robotics communities. There exist state-of-the-art surveys on building reconstruction and modeling from images and LiDAR [1]–[3], where a large portion of the content is dedicated to image-based approaches. Reviews on other urban objects, such as trees, utilities, and sculptures, are rarely found as most of the surveys on surface reconstruction deal with very generic objects of arbitrary shapes and topology [4]. Furthermore, the surveys from graphics communities [1], [2] often ignore a vast amount of work in the photogrammetry and remote sensing communities. A review on reconstruction of various urban models from point clouds (e.g., buildings, trees, utilities, and sculptures) from all the related fields is desired. In this paper, we review the existing urban reconstruction algorithms, prevalent in photogrammetry, computer graphics, and computer vision disciplines, evaluate their performance in the architectural modeling context and discuss the adaptability of generic mesh reconstruction techniques to the urban modeling pipeline.

Urban reconstruction algorithms from point clouds can be grouped into automatic and interactive based on the level of user interaction required. Interactive applications are mostly designed to deal with sparsity and occlusions in the data. We can also classify various urban reconstruction techniques based on the input data or the output model. For example, algorithms designed to work on airborne laser scanning (ALS) considerably differ from those developed for mobile laser scanning (MLS) and terrestrial laser scanning (TLS). Similarly, while most of the reconstruction algorithms generate triangular meshes, there also exist algorithms that construct parametric models consisting of planar or Bezier surfaces. To date, the reconstruction techniques are vast in number and the lines between input- or output-based classification are often blurred. Therefore, we classify the reconstruction algorithms into different groups based on targeted object types and multiplicity. A typical urban environment consists of buildings, roads, bridges, power lines (PL), trees, and sculptures. We review reconstruction algorithms for each of these urban objects, focusing on point cloud data. For each of the urban objects, we again group the methods into different categories according to different algorithmic designs. An outline of the survey is depicted in Fig. 1.

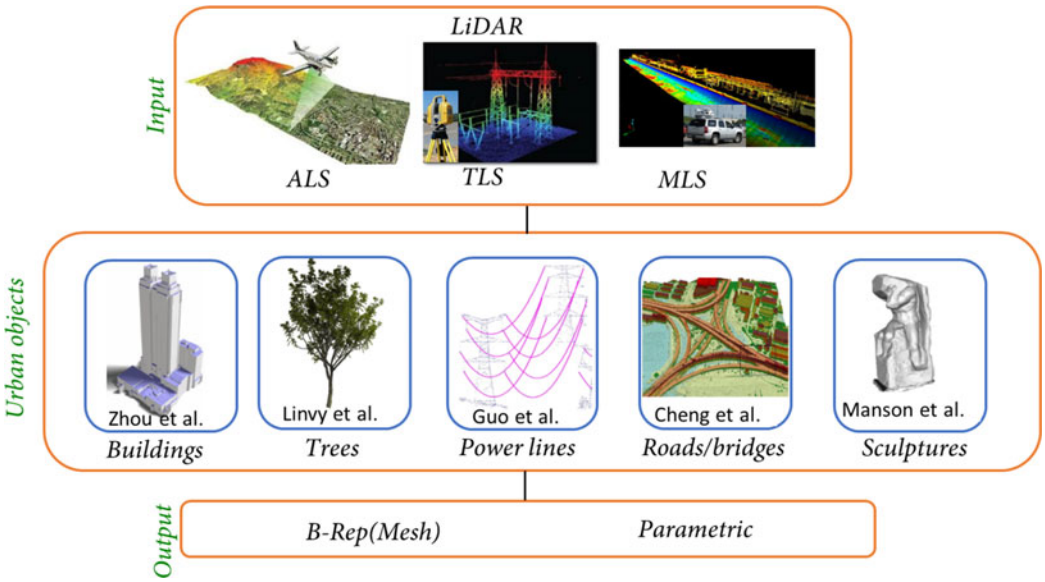


Fig. 1. Outline of the survey. We organize and review various LiDAR-based urban modeling algorithms based on the reconstructed objects.

II. Reconstruction of Building Roofs and Façades

Reconstruction of building models is an active area of research among photogrammetry, computer graphics, computer vision, and remote sensing communities. The topic has been surveyed [1], [3], [5], [6] on several occasions by various communities. To the best of our knowledge, the most recent and comprehensive survey on urban reconstruction was published in 2013, compiled by Musialski *et al.* [1], which mainly focused on state-of-the-art methods in computer graphics and computer vision, briefly touching upon the methods from photogrammetry and remote sensing. The last couple of years have witnessed several new developments in this direction and we primarily focus on updating with these recent algorithms, especially the reconstruction from LiDAR data.

To date, we have urban buildings conforming to a rich variety of architectural styles ranging from ancient Greek to postmodern. Undoubtedly, the design of any building reconstruction algorithm has to consider the architectural style, the input data, and the target applications. For example, algorithms dealing with Manhattan [7] buildings mostly rely on orthogonality and rectilinearity rules. Reconstruction of Roman or Parisian architecture has to account for a distinctive style consisting of arches, vaults, and domes, which is extremely difficult. Similarly, each architectural style exemplifies its own set of rules and these features have to be considered while designing the reconstruction algorithms. With few exceptions, most reconstruction works are not very explicit on the architectural styles that they deal with. Instead, the main focus is to adapt the algorithmic design to exploit the geometry available in the input data to generate buildings.

A. Modeling From Airborne Data

In general, the methods for modeling building rooftops from ALS data can be categorized into data-driven, model-driven, and hybrid-driven, as illustrated in Fig. 2. Data-driven methods for three-dimensional (3-D) building reconstruction use a bottom-up approach that begins with the extraction of primitives, e.g., planes, cylinders, cones, spheres or tori, followed by analyzing primitive topology in 2-D or 3-D space. The geometric elements of the primitives, such as lines and critical vertices of the rooftops, are extracted and grouped to form building models. In contrast to data-driven methods, model-driven approaches involve a top–down strategy that usually begins with a hypothetical model library and then uses rooftop point clouds to search for optimal solutions of model composition from the model library [8] and corresponding parameters of the models. More precisely, model-driven methods require a primitive model library composed of basic building shapes, e.g., flat, gable, hip, shed, saltbox, mansard, pyramidal, and gambrel. The most appropriate models are selected from the library to best fit the point clouds [9], during which the corresponding model parameters are also simultaneously determined. Hybrid-driven methods take full advantage of data-and model-driven methods and consequently, avoid the deficiencies of these methods, if used alone. Practically, hybrid modeling often uses the ground plans [10] or sets of nonoverlapped and connected quadrilaterals [11] to divide the complex rooftop into subsets where the solution already exists. Another alternative hybrid representation has been achieved through rooftop topology graph (RTG) matching [12]–[18]. The RTG is anticipated to be constructed while the rooftop primitives are extracted and the adjacency relationships among these primitives are determined. Through combining data-driven and model-driven methods, the problem of rooftop modeling can be naturally converted to the problem of rooftop graph matching with basic topology graph elements in a model library. In the following subsections, typical rooftop reconstruction systems will be reviewed. We have no intention to provide an exhaustive list of all suggested methodologies, however, rather to provide some insight into the essence of the rooftop reconstruction methods by listing representative literatures.

1) Data-Driven Methods: Over the past two decades, there have been plenty of data-driven methods proposed for building detection and/or modeling from ALS data. These methods employ various techniques, such as pattern recognition, machine learning, and statistical analysis, to extract rooftop primitives.

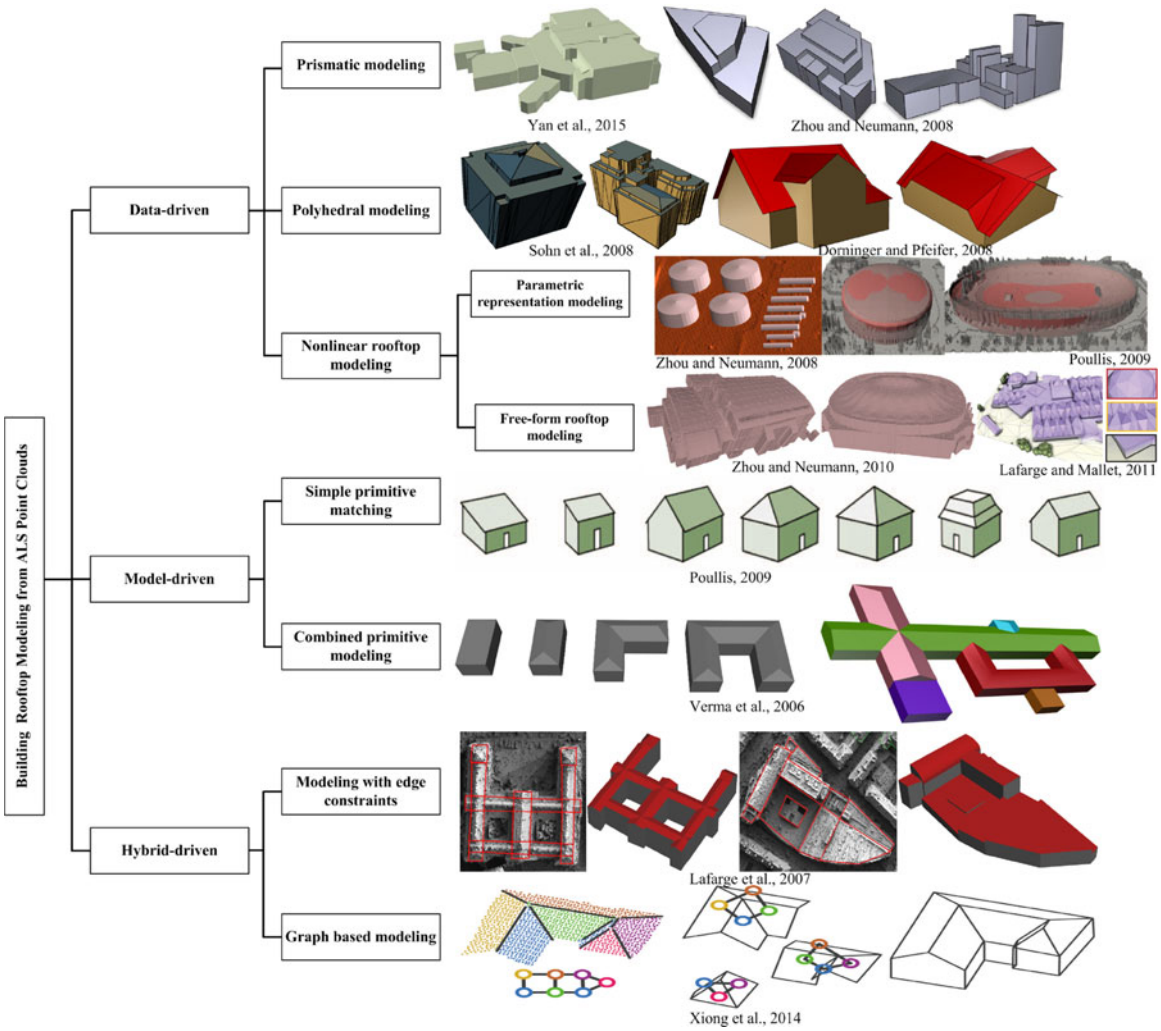


Fig. 2. Classification of rooftop modeling techniques and a few representative results.

Although numerous works are focused on building segmentation [19]–[22], rooftop primitive extraction [23], [24], or primitive boundary regularization [25], these methods fall outside the scope of this survey as they only focus on a single step in the entire modeling pipeline and do not consider 3-D rooftop reconstruction. The reader can refer to these relevant papers for details. In this section, we divide the data-driven approaches into three groups: prismatic modeling, polyhedral modeling, and nonlinear rooftop modeling.

a) Prismatic modeling: Prismatic modeling is suitable for flat or multilevel high-rise flat buildings, especially those in metropolitan downtown areas. For instance, Zhang *et al.* [26] present a prismatic building model reconstruction method. In this method the building outlines have been adjusted and refined by the building's dominant and orthogonal directions to form regularized structures. However, this method can only generate prismatic building models that comply to the criteria for the level of detail (LoD1) block model of CityGML. To model relatively complicated, multilevel, and high-rise flat buildings, Zhou and Neumann [27] present a representation of rooftop boundary for modeling buildings. The highlight of this method is the determination of the buildings' multiple dominant directions through statistical analysis of a group of local building boundaries. This regularity makes it flexible to determine the arbitrary building boundaries without presumption or reference for specific angles between adjacent boundaries. Since each primitive boundary is processed separately, their method cannot guarantee generation of compact and watertight models. This work is further extended and successfully incorporated into a streaming framework for processing large-scale ALS point clouds [28]. By utilizing out-of-core data management, this method overcomes the limitations of traditional methods based on the "divide-and-conquer" concept that divides large-scale data into multiple tiles and processes them individually. Thus, it avoids the artifacts alongside the boundaries between tiles. Similarly, Poullis and You [29] propose a framework for modeling multilevel flat buildings over large-scale areas, including rooftop segmentation, surface fitting, boundary simplification, boundary refinement, and 3-D building model generation. This method has been further enhanced by Poullis [30] to refine segmentation of roof primitives and building boundaries using a two-step unsupervised clustering algorithm followed by a fast energy minimization process. Nevertheless, the limitation of their method is that the boundaries of roof planes are extracted separately. Therefore, crack effects between reconstructed components of the building rooftop models are unavoidable. In addition, all the building boundaries are regarded as piecewise linear segments so that nonlinear boundaries cannot be handled accurately. In

a similar way, Matei *et al.* [31] design a pipeline for modeling buildings in large-scale rural and dense urban areas, including ground segmentation, building rooftop segmentation, and segment orientation, building model generation by simply stretching rooftop segments to the corresponding terrain height. Since the rooftop topology is often inconsistent, crack effects will be unavoidable. In order to overcome these adverse effects, Chen *et al.* [32] propose a multiscale method for reconstruction of multilevel flat buildings, in which the building roofs are accurately detected and segmented by combining two-level depth images, and then the rooftops are constructed by simply stretching patches to the ground. By transforming the point clouds into corresponding depth images, this method ensures topological consistency between rooftop components. Chen *et al.* [33], [34] use the subgraph of Voronoi diagram to maintain the topological relationship among primitives for reconstructing watertight and compact multilevel flat buildings.

b) Polyhedral modeling: Polyhedral modeling is the most commonly used data-driven rooftop modeling technique and can be adapted for generating building models with both simple and complex topology of roofs. Vosselman [35] presents a rooftop modeling approach based on detecting and outlining roof primitives from laser scanning point clouds. Ridge and jump edges are found by intersection of the rooftop faces and analysis of height discontinuities. In order to obtain accurate jump edges, the orientation of the main building is derived and used as a constraint for jump edge orientation. The rooftop topology is finally maintained by analyzing the relationship of linear features and bridging the gaps between the detected adjacent edges. Mass and Vosselman [36] also use the intersection of planar faces from triangulated point clouds to construct relatively complex roofs. Sohn *et al.* [37] introduce a Binary Space Partitioning (BSP) tree to recover the rooftop topology between adjacent planar or linear segments. Based on this correct topology, their method has the capability to generate different representations of a polyhedral roof model at different levels of detail by means of controlling the depth of the BSP tree. In another approach, Dorninger and Pfeifer [8] analyze the topological relationship among adjacent rooftop primitives to produce intersecting edges combined with regular building outlines for generating watertight polyhedral rooftop models. Although detailed 3-D building models with rooftop overhangs are generated, manual interventions are required during the postprocessing steps to eliminate possible artifacts from the building models. Moreover, for complex building rooftop structures, the interior structure lines are not easily extracted by simply using the intersection operation between the planar primitives and its neighbor primitives. Sampath and Shan [38] propose an approach for reconstruction of polyhedral rooftop in which an improved fuzzy k-means algorithm is employed for rooftop primitive segmentation. The interior vertices of the roofs are restored by using the adjacency matrix of the primitives. Although most vertices of the rooftops are obtained by the adjacency matrix, the vertical walls and building boundaries are also needed as *a priori* knowledge to determine the vertices on the outlines.

Kim and Shan [39] propose a polyhedral rooftop reconstruction based on the level set method, determining the interior and boundary vertices by adjacency relationship of primitives and building footprints, respectively. The order of these critical vertices is restored by traversing the boundary of each primitive on a binary image. To make the polyhedral model visually convincing, Zhou and Neumann [40] propose a framework of rooftop reconstruction by introducing a variety of global regularities, including roof–roof regularity, roof–boundary regularity, and boundary–boundary regularity. The principle of this method is that global regularity reveals the primitives' topological relations and similarities for intrinsic structure of building models arising from architectural designs. Unlike previous hard constraints, global regularity provides a relatively flexible representation of global knowledge in rooftop models. The algorithm is suitable for handling complex rooftops and improving model quality in terms of both fitting errors and visual effects. To simplify the maintenance of rooftop's topology, Yan *et al.* [41] introduce a 2-D snake algorithm to provide a tradeoff between parallelism of the rooftop's primitive boundary and deviation from its original position. In their method, the problem of complex rooftop reconstruction in 3-D space has been naturally transformed into a problem of maintaining 2-D primitive's topology. Similarly, Chen *et al.* [33], [34] use ESRI polygon Z data structure to maintain the final rooftop vertices and their topological relationships, which makes it easy to delineate and render 3-D building rooftop models through triangulation.

c) Nonlinear rooftop modeling: In general, these types of methods mainly deal with complicated free-form rooftop structures. Vertex decimation [42], vertex clustering [43], and/or edge collapse methods [44], [45] have been frequently employed to eliminate redundant points, edges, or triangle meshes. In theory, these methods can handle rooftops with arbitrary complexity and always result in topologically consistent representations that are geometrically close to the raw data. In practice, however, the following issues should be considered.

1) The reconstructed rooftop models do not ensure the characteristics of model compactness. That means the models still contain many triangular meshes, which is not suitable for model storage, web transmission, and acceleration of rendering. Actually, the models with fewer triangles tend to be more compact.
2) The reconstructed models often do not contain semantic information. Recently, well-known commercial photogrammetric software packages such as Acute3D,[1] Pix4D,[2] and SURE[3] have included the capability of modeling large-scale 3-D urban scenes from point clouds derived from the stereo image pairs using photogrammetry algorithms. However, the whole reconstructed model is a giant mesh. This general model representation is often deprived of semantic information.
3) The geometric error criterion, commonly used for rooftop simplification, is insufficient to maintain the regularity of buildings, especially the sharp boundaries of building outlines.

In order to generate lightweight building models at different LoDs, many researchers employed Quadtree simplification [46], tetrahedral mesh simplification [47], and Octree 3-D geometry simplification [48]. To obtain the semantic information, the simplification procedure usually needs additional constraints by adding specific semantic components, e.g., planes, cylinders, cones, and tori [45]. These semantic components are usually detected by shape detection algorithms, such as RANSAC [49] or Hough transform [50]. In particular, Lafarge and Mallet [51] propose a complete, realistic, and semantically rich descrip-

[1]http://www.acute3d.com
[2]http://pix4d.com
[3]http://www.nframes.com

tion of urban scenes by simultaneously reconstructing buildings, trees, and topologically complex ground surfaces. To do so, a 2.5 D planimetric map is employed for labeling different semantic urban scenes (tree, building, and ground) through solving a nonconvex energy minimization problem. By using the 2.5 D planimetric map for representation of complex urban scenes, the problem of maintaining complex topology of rooftop primitives is transformed to a labeling problem. This method reconstructs rooftop models composed by not only multiple semantic components but also free-form components.

To ensure building regularity, topological constraints are introduced to preserve the topological properties of man-made objects to guarantee regularity and high fidelity of the final models. Zhou and Neumann [46] propose a general building representation using a mesh simplification procedure based on a 2.5 D dual contouring. The point clouds are first converted into Hermite data of surface and boundary over a 2-D grid. A simultaneous optimization over 3-D surface and 2-D boundary of roof layers is implemented by minimizing quadratic error functions to generate a hyper-point in each grid cell, containing a set of 3-D points with the same x and y coordinates but different z values. These hyper-points are connected to meshes by generating roof and wall polygons. The reconstructed models are guaranteed to be crack-free with small fitting errors for preserving sharp features. This method is further extended to model building rooftops in residential areas where most of the rooftops are partially missing due to vegetation occlusions [52]. The membrane interpolation is applied to the 2.5 D Hermite data with nonempty Hermite points as boundary conditions. Although 2.5 D dual-contouring can deal with arbitrary rooftops, the optimization does not take the building's topology into account, resulting in distorted rooftop outlines with numerous insignificant triangles along thin, long rooftops. To overcome these drawbacks, Zhou and Neumann [53] extend the 2.5 D dual contouring algorithm into a 2.5 D modeling method with topology control. The reconstructed models are compact with less triangles, preserving the topological features of each building. Essentially, the 2.5 D dual contouring algorithm is actually a remeshing method, which resamples or reorganizes the mesh vertices to meet specific quality requirements.

Besides modeling rooftops by nonparametric meshes, there are cases of using parametric representations for nonlinear modeling. For instance, to detect oil tins and tanks in industrial sites, Zhou and Neumann [27] employ interactive operations to select specific object primitives and then determine their parameters using the RANSAC algorithm. Poullis and You [54] use a semi-automatic approach for recognition and reconstruction of nonlinear rooftops such as dome-like and stadium-like structures by fitting an ellipsoidal primitive to the data. Lafarge and Mallet [51] combine geometric 3-D-primitives, such as planes, cylinders, cones, and spheres along with mesh-patch representation to jointly represent complex building rooftops. These nonlinear primitives are detected by an iterative nonlinear minimization, e.g., Levenberg–Marquardt optimization.

In summary, the problem of building rooftop reconstruction based on a data-driven framework is often transformed into a problem of consistency maintenance of topological relationships among rooftop primitives, primitive boundaries or their combinations. In addition, data-driven methods normally can create complicated building models, and are not restricted to shape primitives. In theory, they can be used to model any type of rooftops, including curved and free-formed rooftops. In practice, however, the computational cost is usually high. The data-driven methods are sensitive to the local point density, noise, outliers, and data missing, especially for point clouds derived from image-matching techniques.

2) Model-Driven Methods: In contrast to the data-driven methods, model-driven methods involve a top–down approach that usually begins with a hypothetical model and then uses point clouds to verify that model. These methods require a model library composed of basic building shapes, e.g., flat, gable, shed, saltbox, mansard, pyramidal, and gambrel. The most appropriate model is then selected from the library to best fit the point clouds. For example, Maas and Vosselman [36] employ the invariant moment technique for modeling a symmetric gable roof. This technique allows modeling of asymmetric deviations like dorms on roofs, and closed solutions can be formulated for determining parameters of simple gable roof models. Poullis and You [54], [55] develop a model-based method for photorealistic reconstruction of large-scale virtual environments. They parameterize geometric primitives of rooftops by exploring symmetry constraints that commonly exist in man-made architectures. The optimal parameterized model variables are obtained by solving a quasi-Newton minimization for the bound-constrained problems. This method determines the types of roof models and associated parameters simultaneously, which significantly reduces the number of model parameters and increases the computational efficiency of the reconstruction process. Huang *et al.* [56], [57] present a generative model for automatic reconstruction of building rooftops from ALS data. The selection of roof primitives, as well as the sampling of their parameters is driven by a variant of Markov Chain Monte Carlo technique with a specified jump mechanism. By introducing a combination and merging mechanism, the modeling scheme allows overlapping of primitives during the reconstruction process. Notably, this method can generate a series of rooftop models at different LoDs by controlling the jump mechanism. Henn *et al.* [58] combine RANSAC/MSAC (M-estimator sample consensus) with support vector machine to derive best fitted models from extremely sparse ALS point clouds (1.2 point/m^2). This method divides the building (footprints) into small rectangular cells and assumes that the footprints of the buildings are orthogonally dominant. Thus, nonorthogonal buildings (footprints) are not divided by the rectangle decomposition. Model-driven methods often suffer from the insufficiency of hypothetical models and possible mismatches between data and models. Although a "divide-and-conquer" strategy for generating complex rooftops or expanding the model libraries can partially solve this problem, it is still difficult to accurately match various building rooftops by using libraries with insufficient number of hypothetical building models.

3) Hybrid Modeling: Hybrid methods combine advantages of the data-driven and model-driven methods for minimizing disadvantages from either method alone. To accomplish this goal, two issues should be considered: How to incorporate the results of data-driven methods into model-driven methods and to what extent can we make this combination effective. The hybrid methods can be grouped into two categories: modeling with edge constraints and topology graph modeling.

a) Modeling with edge constraints: Since complex building rooftops usually cannot be exactly matched to the models available in the libraries, the constructive solid geometry (CSG) modeling technique is often employed to decompose complex modeling tasks into simple subtasks. After the con-

struction of appropriate sub-building models, they can be combined to form a complex building model by the CSG technique. In this section, the data-driven results including building footprints, jump edges, and ridge edges are used as constraints to partition a complex building into multiple components that can be easily processed by the model-driven methods.

For example, Haala and Brenner [59] and Brenner [60] use ground plans from cadastral maps as constraints and decompose the ground plans into basic units (i.e., 2-D primitives). Each 2-D primitive is the corresponding footprint of the 3-D primitive, and the roof type of 3-D primitives and their parameters are determined by model fitting according to the corresponding 2-D primitives. The primitive models in each 2-D primitive are organized to form a complete building rooftop model with CSG style, which is further refined by subsequent manual interactions. By integrating ground plans, the problem of constructing complex roofs has been transformed into a problem of constructing its basic primitives. The basic primitives have fixed or limited parameters, which simplifies the rooftop construction considerably. To obtain a better rooftop partition, Lafarge *et al.* [61] use the marked point process method to find a set of rectangles on 2-D space for each building. The merging of the neighboring rectangles refines the overlaps or gaps between rectangles. The jump edges are detected and projected onto these 2-D rectangles by analyzing discontinuity of the roof height. The parametric approach is chosen to determine rooftop and gutter height. The rooftop model is created by an assumption that the roof structures are constrained to be symmetric two planes. In order to construct more complex roof models such as dissymmetric or gambrel roofs, Lafarge *et al.* [11] extend this method by dividing the digital surface model into a set of connected blocks, i.e., quadrilaterals and/or triangles. Basic model libraries are established for each block and the Gibbs model is used to control block assemblage and data fitting. For each block, the optimal model parameters are obtained by using Bayesian posterior probabilities. This method can be applied to large-scale scenes and produce compact and watertight models. Kada and McKinley [62] divide the point clouds of buildings into nonoverlapping quadrilateral cells based on the corresponding ground plans. The point clouds are first classified into cells according to the orientations of their locally estimated planes and then the optimal model that best fits the points in each cell is obtained. However, this method requires various rigid constraints imposed on specific building rooftops, which reduces the accuracy of the modeling. Zheng and Weng [63] decompose the irregular building footprints into nonintersection and quadrangular blocks for the identification of most probable primitive models. The decision tree algorithm is used to determine the most appropriate type of the primitive model from the predefined primitive model libraries. Parameters including width, length, orientation, gutter height, and ridge height are determined from the point clouds in each block. All the block models are built in the Google SketchUp software and combined into complete building models. However, the assumptions of symmetric ridges in each block, the uniform height of gutter, and without modeling superstructure and dormers, limit this method when dealing with large-scale urban areas with numerous roof types.

b) Graph-based modeling: Graph-based modeling methods employ not only edges but also relationships among edges, primitives, or their combinations for rooftop modeling. A series of publications along this line [12]–[18] demonstrate that the roof topology graph (RTG), as an intermediate connector, can facilitate complex rooftop modeling.

Using the RTG for building reconstruction from images has been described earlier by Ameri and Fritsch [12]. Verma *et al.* [15] establish an RTG based on segmentation of the rooftop primitives, where the subgraphs representing structure of roof primitives are defined in the RTG model library. The complex RTG are recognized by searching subgraphs, starting with the most complex subgraphs. The parts of the RTG that do not match with the shape available in the RTG model library are considered as rectilinear objects. Elberink [16] and Elberink and Vosselman [17] add more subgraphs to the model library to match with relatively complex building rooftops. The advantage of this method is that the RTG constraints can transfer knowledge to raw point clouds implicitly for deciding optimal gutter height, ridge direction, and roof critical points. However, there exists a degenerated situation that the rooftop geometrically does not fit the constraints from the model library. Most of the cases are due to incomplete matching caused by over segmentation, limited target models, lacking topology relations, undetected segments, and spurious roof segments. In this situation, interactive post-processing is needed. Xiong *et al.* [14] demonstrate that any rooftop topologies are constituted by three basic elements, i.e., loose nodes, loose edge, and minimum cycles. So the subgraphs contained in the library can be defined according to these basic elements. The problem of rooftop reconstruction can be transformed into the problem of detecting the subgraphs by searching for these basic elements. For each detected subgraph, the roof plane and boundary constraints are introduced to regularize the graph geometry while minimizing least square (LS) fitting errors. These subgraph models are finally grouped into a complete building model. To eliminate erroneous rooftop topology, Xiong *et al.* [13] design the graph edit dictionary containing the common erroneous subgraphs and corresponding corrected ones to refine the rooftop model. The results show that this method can modify the erroneous models from an average of 25% to 50% for some test areas with only 15 entries in graph edit dictionary, improving the rooftop reconstruction rate to 95% for the experimental Enschede dataset. Recently, Wang *et al.* [18] fully explore the symmetry property of the rooftop planes from an attributed RTG by using semantic decomposition techniques and then combine the subgraph models into compound building models. In contrast to previous methods, the decomposition of the subgraph is defined in the high level of granularity, i.e., hipped, half-hipped, gamble subgraph with specific semantics. The reconstruction step is more flexible by allowing overlap of adjacent subgraphs.

B. Ground-Based Reconstruction

In recent years, the goal of further segmentation and fitting of parameterized high-level polyhedral models emerged. We will focus on those approaches in this section and consider the work using TLS and MLS data for outdoor environments. Unlike well-controlled laboratory environments,[4] outdoor scenes contain buildings, trees, cars, pedestrians, and many other man-made or natural objects. Due to the constraints of scanning distance, field of view, object self-occlusion and occlusion by others, it is always difficult to obtain a complete and sufficient sampling of all the building surface. For instance, the rooftops

[4]http://graphics.stanford.edu/projects/mich/

and back of the buildings often cannot be scanned by an MLS system. The sampling rate of surfaces also varies in terms of its distance and orientation to the scanner [65]. For instance, the point clouds from the upper levels of a high building are normally sparser than those from the lower level. Moreover, laser beams can penetrate transparent surfaces such as glass windows, doors and walls, which results in insufficient point clouds collected from these surfaces [66].

We consider a TLS system as a type of stationary scanning method, in which multiple scans at different locations are needed to obtain "complete" data. Because building roofs are still often missing due to viewpoint constraints, the meaning of "complete" is relative to the laser scanning data acquired from one location. To process multiple scans, range registration is needed to combine the separate scans into a complete dataset. A common 3-D-to-3-D registration method is the iterative closest point algorithm [67]; a typical 3-D building modeling method using ground-based LiDAR is presented in [68]. This is a bottom-up method that involves point-wise normal computation and classification using principle component analysis (PCA), cluster merging, surface fitting, and boundary extraction. Based on an assumption that the surfaces of a building can be represented as a bounded polyhedron, Chen and Chen [65] detect planar regions and their intersections to generate piecewise planar building models from sparse laser scanning data. A top-down approach is presented in [49], [69], which is based on a sequential RANSAC [70] strategy to detect planes, spheres, cylinders, cones, and tori from unorganized point clouds.

MLS data are continuously collected from laser scanners mounted on a vehicle while driving at a posted speed. The work in [75] employs two 1-D laser scanners mounted on the roof of a vehicle, scanning the scene horizontally and vertically, respectively. Based on a flat terrain assumption, a 3-D model is reconstructed by registering and integrating horizontal and vertical scans. Through edge detection and linking from MLS point clouds, Xia and Wang [76] provide an effective way to achieve building abstraction and reconstruction. Frueh *et al.* [77], [78] develop a set of algorithms for generating textured façade meshes of cities from a series of vertical 2-D surface scans and camera images. They classified points into foreground and background layers using a concept similar to [79], and detected major building structures in the depth images. Large holes in the background layer are filled. In [80], Zheng *et al.* propose an interactive consolidation method to assist the façade modeling from partial missing and uniform density MLS points. In this method, the similar structural components are detected by interactive operations and then are registered together. Furthermore, the weighted L_1-median optimization is used to remove in-plane and off-plane outliers from each group of similar components. The consolidated point clouds are then used for detailed façade modeling.

The work in [81] introduces an interactive tool to create detailed building façades from partial or missing MLS data (possibly with ambiguous scenes) with the help of SmartBoxes from which the balconies of a building can be well modeled. The key idea is that the user-defined building blocks are automatically adjusted to fit the point data through a discrete optimization, considering the two force terms defined by data-fitting and contextual relationships with nearby similar boxes. Another iterative method is O-snap [82], an automatic polygonal reconstruction that can be interactively refined by the user. In [83], Sui *et al.* successfully transforms building reconstruction into an energy minimization problem under the Markov Random Field (MRF). Specifically, the authors reconstruct the entire building façades through reconstruction of multiple horizontal slices, among which dominant slices are constructed and propagated to other slices through MRF. The reconstructed dominant slices are then assembled into the whole buildings. The rapter [84] is a global algorithm to simultaneously select both planes along with their sparse inter-relations for modeling buildings as so-called regular arrangements of planes. The semantic modeling work [85] combines decomposition and reconstruction for modeling residential single-family houses. Particularly, the blocks in point clouds are defined by a series of assumptions, such as planarity, symmetry, and convexity, and organized by hierarchical tree structure. Then the potential candidate blocks are effectively detected in the decomposition step. In reconstruction step, the corresponding block models are selected and its parameters are determined by minimizing fitting errors. The reconstructed building models have obvious semantic labels, e.g., walls, roofs, columns, eaves, garages doors, and chimneys, which is beneficial to model editing and retargeting.

The façade parsing is a crucial step for ground-based façade modeling. For example in [86], the façade TLS points are first segmented into multiple 2.5 D depth planes, among which the façade elements are accurately detected by collectively using semantic segmentation, arrangement priors of façade elements, and machine learning recognition. The element boundaries are finally extracted for modeling 3-D façades. Another interesting line of work in this direction include transferring the geometric details of façades to coarse building models from the corresponding mobile laser scans. In [87], Peethambaran and Wang employ coded planar meshes and the MLS data to augment the quality of coarse building models available in 3-D maps, such as Google Earth. Their pipeline make use of a façade parsing algorithm based on a weighted point count function defined over the window or door boundaries. Adaptive partitioning described in [88] automatically derives a flexible and hierarchical representation of 3-D urban façades including the interleaved façades, and hence represents a useful tool that can be used in the parsing stage of façade modeling pipeline.

C. Summary

First, we reflect on various considerations to be taken into account while choosing a rooftop modeling technique for a particular application. A comprehensive building model consists of three elements: geometry, topology, and semantics. Geometry refers to building shape, size, and the spatial locations of the structural components. Topology refers to the existence of building structural components and their adjacency relationships and semantics represents the functional aspects of the building models. If the user primarily requires the rooftop models with high geometric accuracy and complete roof components, the methods based on data-driven framework often become the first choice. In this case, these methods will perform robustly to approximate most types of rooftops. If the user mainly focuses on generating watertight building models without artifacts or keeping building regularity typically present in man-made environments, the methods based on model-driven framework must be first taken into consideration. When a tradeoff between topology and geometry is necessary, the algorithms from hybrid-driven framework should be considered in the first place. Semantically, rich building models have explicit structural components.

TABLE I
LIST OF MAJOR TECHNIQUES ADDRESSING THE RECONSTRUCTION OF 3-D ROOFTOPS FROM AIRBORNE LIDAR

Technique	Model Rep.	Strengths/Robustness	Challenges/Limitations
Zhang *et al.* ([26])	Triangle mesh	Determination of the dominant directions of the footprint	Generate only LoD1 buildings
Zhou and Neumann ([27])	Triangle mesh and parametric	Detection of multiple dominant directions; oil tins and tanks detection	Crack effects exist in rooftop components
Poullis ([30])	Triangle mesh	Large-scale urban scene reconstruction	Only generate multi-level flat rooftops
Chen *et al.* ([33])	ESRI shape	Maintenance of the rooftop topology relationships	Only generate multilevel flat rooftops
Vosselman ([35]); Mass and Vosselman ([36])	Triangle mesh	Rooftop topology maintenance	Topology is sensitive to noise and data occlusion
Sohn *et al.* ([37])	Triangle mesh	Rooftop model presentation at different LoDs	Roof model is subject to the quality of linear features
Dorninger and Pfeifer ([8])	Triangle mesh	Rooftop model with detailed eave overhangs	Manual intervention
Sampath and Shan ([38])	Triangle mesh	Adjacency matrix for roof topology representation	*A priori* assumptions on the model vertices
Kim and Shan ([39])	Triangle mesh	Simultaneous segmentation of multiple rooftop primitives; Doubly connected edge list rooftop topology	Occurrence of trivial primitives when segmentation algorithm traps at the local minimum
Zhou and Neumann ([40])	Triangle mesh	Visually convincing roof models	Too many regularity assumptions,less flexibility
Yan *et al.* ([41])	ESRI shape	2-D roof topology optimization problem	Difficulty in determining effective energy functions for 2D snake algorithm
Kwak and Habib ([64])	Triangle mesh	Rectilinear roof primitive boundary extraction and optimization	Addresses only the buildings with multiple rectilinear primitives
Lafarge and Mallet ([51])	Triangle mesh	Complex rooftop representation; Large-scale rooftop modeling; Parallel processing schema	Heterogeneity and outliers due to SfM
Zhou and Neumann ([46])	Triangle mesh	2.5 D dual contouring roof representation	Rooftop model with numerous insignificant triangles
Maas and Vosselman ([36])	Parametric	Invariant moment modeling of symmetric gables with window dorms	Only simple rooftop structures
Poullis and You ([29], [55])	Parametric	Linear and nonlinear rooftop modeling	Heavily depends on symmetry
Huang *et al.* ([57])	Parametric	Different LoDs representation; Allows primitive model overlapping	The primitive library is limited
Haala and Brenner ([59]); Brenner ([60])	Parametric	CSG style modeling	Only construct simple roof model
Kada and McKinley ([62])	Parametric	CSG style modeling	Nonoverlapping quadrilateral cells partition
Verma *et al.* ([15])	Parametric	RTG modeling	Only planer-primitives
Oude Elberink and Vosselman ([17])	Parametric	RTG modeling; More subgraphs in model library	Difficulty in balancing between the incorrect matching rate and the number of primitives according to the building complexity
Xiong *et al.* ([13]); Xiong *et al.* ([14])	Parametric	RTG modeling; Definition graph edit dictionary	Difficulty in balancing between the incorrect matching rate and the number of primitives according to the building complexity

In the table, the methods in 2, 7, 19, 22, 23 rows are semiautomatic and the remaining ones are automatic.

Therefore, data-driven methods can generate the semantic information, e.g., planes, spheres, cylinders, cones, and tori at a fine-level scale. The model-driven methods have the capability to derive a coarse-level semantic information, e.g., primitive units, while the hybrid-driven methods can achieve a balance between fine- and coarse-level semantic information. Table I provides a quick reference to relevant urban reconstruction techniques along with their strengths and limitations.

A priori knowledge on the LoD required in the reconstructed building models is crucial in choosing the right input source as well as the computational requirements. LoDs are mainly determined based on the target applications and user needs. For a large scale rendering, we do not require high LoDs in the building models. However, for applications such as 3-D navigation or for landmark building, it is often desirable to have buildings with high LoDs. CityGML (Open Geospatial Consortium) is a widely accepted and standard model describing the representation, storage, and exchange of digital 3-D city and landscape models. In CityGML, 3-D objects are described with respect to their geometry, topology, semantics, and appearance, and defines five different LoD (see Fig. 4), i.e., LoD0 through LoD4. LoD0 is essentially a 2.5 D Digital Terrain Model (DTM) over which a 2-D map or an image may be draped. LoD1 is block models comprising prismatic buildings with flat roof structures. LoD2 buildings consists of piece-wise planar roofs and thematic façades. LoD3 provides additional details such as roof superstructures, doors, and windows and LoD4 buildings equip with interior walls and rooms. A recent work [89] addresses the LoD generation from raw meshes of urban scenes. As already mentioned in Section II-B, splitting of façade points into horizontal floors and tiles is

a crucial step in recovering the façade details. From our study on various façade parsing algorithms, we observe that adaptive partitioning [88], performs well for a range of façade structures including interleaved façades and hence represents a good choice for the parsing stage of the modeling pipeline.

III. Vegetation Modeling

Tree models are mainly used in computer graphics and forest management domains. In computer graphics and vision, tree modeling efforts have been mainly aimed at augmenting the visualization to add considerable realism to virtual urban environments. In the remote sensing community, 3-D models of trees are increasingly used for determining and predicting tree characteristics and quality attributes. Existing work on tree modeling can be classified into procedural approaches such as L-systems [90], [91], image-based approaches [92] and scan-based approaches [93]. The scope of this article allows us to review major tree modeling algorithms and frameworks that take laser scan or point clouds from multiview stereo as inputs. Please be aware that vegetation detection [94], individual tree segmentation and recognition [95] are frequently considered as an important and critical preprocessing step to accurately acquire the tree point clouds. Since the major focus of this review is modeling, we omit vegetation detection and segmentation methods from this survey.

A. Tree Crowns

Tree crowns mainly consist of branches and leaves. Complex topology and variable geometry of leaves and twigs makes tree modeling, in particular, tree crown reconstruction an extremely difficult task. A widely used approach to tree modeling is to infer vital clues on the main branching structure via skeletonizing the point clouds, complete the geometry through appropriate heuristic rules and, further add the leaves using L-systems [90]. For example, Xu *et al.* [96] extract the tree skeleton using Djikstra's shortest path algorithm on a weighted graph constructed from the input point clouds, employ tree allometric theory to convert the skeleton to the corresponding mesh and, subsequently, add the leaves using L-system. Linvy *et al.* [97] propose skeletal structures called branch structure graphs (BSG), which can be reconstructed through global optimizations to fit the point clouds. The BSG method is capable of reconstructing multiple overlapping trees without presegmentation.

In [98], alpha-shape [99] has been used to reconstruct coarse tree crown models. The crown geometry computed using alpha shape can also be used to increase the LoDs through texture lobes [72] (Fig. 3(b)). Cote *et al.* [71] (Fig. 3(a)), [100] utilize the reflectance properties of laser, i.e., the intensity differences between points from the branch surface and those from leaves, to reconstruct tree models. For range images, using local curvature directions and their distribution [101] for segmentation is a reasonable choice. Notably, the axis vectors at points from branches are regularly distributed, whereas the axis vectors at crown points are irregular (the reader may refer to Fig. 4(a) in [93] for qualitative details). Taking this idea further, Zhang *et al.* [93] put forward a data-driven technique based on a marching cylinder algorithm and particle flow simulation to model multilayer representation of trees. Dealing with inhomogeneous point clouds, Aiteanu and Klein [102] use principal curvatures of

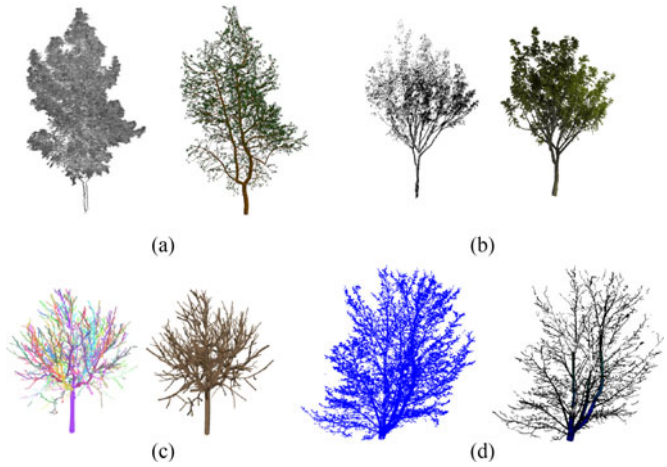


Fig. 3. (a) Example of a digital tree with crown [Image courtesy: [71]], (b) Texture-lobe-based reconstruction of tree crowns [Image courtesy: [72]], (c) B-spline lofted representation of branch structures [Image courtesy: [73]], and (d) Example of QSM [Image courtesy: [74]]. Each sub figure consists of the laser scan [segmented in (c)] and the corresponding reconstructed model.

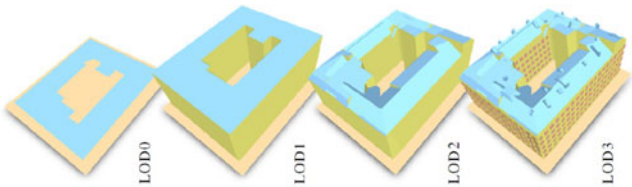


Fig. 4. LoDs used by CityGML. (Left to right): LoD0 represents the footprints of the building, LoD1 consists of prismatic blocks with flat roofs, LoD2 building with piece-wise planar rooftops, and LoD3 building with more details, such as windows, doors, or other roof structures. [Image courtesy due [89]]

points along with ellipse fitting and minimum spanning tree for reconstructing dense and sparse regions, respectively.

ALS point clouds are an ideal source to model tree crown in the large-scale scenes. Unlike MLS and TLS laser scans, ALS data captures very few samples on branches and twigs, making the branch and twig generation a difficult task. In [103], Lafarge and Mallet reconstruct the tree model from ALS points by matching simple ellipsoidal template to tree clusters. The centers of mass of trees are first detected by a watershed algorithm and the other parameters associated with templates, such as the height and the radius are simultaneously found by minimizing the distance from the points to an ellipse. To model complicated tree structures from ALS data in residential areas, in [104] the inscribed octahedrons of the ellipsoid fitted by the local points through PCA are adopted to represent the leaves. For generating large-scale tree models with light-weight characteristics, the semantic decomposition framework [85] employ the billboard representation for creating visually plausible tree models from MLS point clouds.

B. Skeletal Structures

A few approaches focus on reconstructing stems and branch structures without leaves or with sparse leaves. These approaches are mainly designed for addressing the following aspects. A majority of the trees that are scanned during winter contain only a few leaves, thus, making the interior branches of the trees visible to the scanning devices. Tree models with-

out leaves help in accurately estimating various geometric attributes such as taper, volume, diameter, and size, which are vital ingredients for the forestry and timber industry. Addressing coniferous trees and the trees scanned in winter, Pfeifer *et al.* [105] have utilized a volumetric method for extracting skeletons from points and built meshes for stems using cylinders. Tree branch skeletons can be constructed by cylinder fitting on point cloud segments generated via k-means clustering [73], (An example of the reconstructed model is shown in Fig. 3(c)) and the radius of each internode along the tree skeleton can be estimated [106] using axis directions and branch thickness. In [107], Wang *et al.* employed minimum distance spanning tree and structure-aware global optimization to extract the tree skeleton from TLS point clouds. The extracted skeleton is smoothed by a Laplacian function, inflated to a tree model with the help of radii estimated using a cylinder-fitting procedure and, subsequently, enriched with leaf models.

Tree skeletons can be derived from geometric and topological structures such as Voronoi diagrams and Reeb graphs [108]. Bucks *et al.* [109], [110] use an octree graph built from the point clouds to extract the skeleton and embed the skeleton into the point cloud. The authors show that the extracted skeleton is a super graph of a Reeb graph, a widely used topological structure in computer graphics and shape modeling. In [111], Voronoi skeletons obtained from the boundaries of surface patches are split into paths and a principal curve representing the component shape is computed from each path's point cluster.

Quantitative structure models (QSMs) are a hierarchical collection of cylinders that provide the volume and diameter of branch segments needed to estimate the biomass. QSMs help in accurately and nondestructively estimating the volume, branching structure and branch size distribution, and further aids in growth and change detection [112]. There exist several works addressing QSM generation from TLS point clouds. The skeleton modeling technique by Raumonen *et al.* [74] (Fig. 3(d)) is based on covering the point cloud data with small sets of corresponding connected patches on the tree surface and then, using the neighbor relationships and geometrical properties of these sets to reconstruct the cylindrical model of the tree. In 2015, the authors have extended the cover-set method to reconstruct QSM of every tree in a forest plot [113]. Belton *et al.* [114] extract features (eigenvectors and their ratios) from TLS point clouds through principal component analysis and separate leaf points from wood points in feature space via a Gaussian mixture model. Skeletons are then extracted using the centers of ellipses fitted onto the horizontal slices of woody parts, and thus, the skeletons obtained are utilized to generate cylindrical tree models. Li *et al.* [115] segment the point clouds into trunks, ground, or bush and scatter using normal direction at each point. Polygonal models of a trunk are then created via interpolating the circles fitted to the horizontal slices of the point clouds along the axis.

C. Summary

We summarize different practices in tree modeling research. A widely adopted pipeline for tree modeling include skeletonization of the point clouds, infer the geometry and topology of the tree trunks and branches and then add the leaves and twigs via L-systems [90]. Skeletal structures, in particular, the QSMs represent an efficient representation for estimating the tree quality attributes in forestry and/or remote sensing applications.

TABLE II
OPEN SOURCE TREE MODELING PROJECTS

Software	Weblink
Simple Tree	http://www.simpletree.uni-freiburg.de/
Computree	http://computree.onf.fr/?lang=en
Pypetree	https://github.com/cjauvin/pypetree
OPenAlea	http://openalea.gforge.inria.fr/dokuwiki/doku.php

There are a few open source implementation to construct and customize tree models. Simple tree [116] and Computree are open source platforms utilizing TLS point clouds. They are capable of generating a DTM on a plot. Stems are detected by fitting circles with a Hough transformation (HF) into slices of the vegetation points above the DTM. Circles in a spatial neighborhood are combined to form stem models. Pypetree [117] is a tree perennial tissue modeling software based on the skeletal extraction algorithm devised by Verroust and Lazarus [118]. The algorithm finds implicit skeletal curves by computing four distinct graph structures, namely neighborhood graph, geodesic graph, level sets, and the skeleton, where each one is derived from its preceding structure. These software packages are available on the web links provided in Table II.

IV. MODELING URBAN UTILITIES

A. PL Reconstruction

3-D PL models are used for analyzing the structural stability of the power cables [119], thermal rating and detecting potential hazards, such as electromagnetic field pollution and vegetation encroachment [120]. Due to gravity, power cables hung between two adjacent pylons assume the shape of a catenary as shown in Fig. 5(a), which bears resemblance to parabolic curves. In mathematics, catenary [the word is derived from Latin term "catenaria" (chain)] is a curve that describes the shape of a flexible hanging chain (or cable). A catenary curve $\mathcal{C}(a, b, c)$ in the XZ plane is given by (1), where a and b are the parameters for translation of the origin, and c is a scaling factor denoted as the ratio between the tension and the weight of the hanging flexible wire per unit of length [121]

$$y = a + c\cosh\left(\frac{x-b}{c}\right). \tag{1}$$

1) Catenary Curve Fitting: Most of the PL reconstruction approaches start by classifying the LiDAR points into PL and non-PL points. The final model is then created by fitting catenary curves to the detected PL points. For example, Melzer and Briese [124] employ a 2-D HT to locate the groups of parallel PLs in the projection of points onto the XY plane. These detected lines are then grouped into corridor segments. The intersections of these corridor segments are computed and, subsequently, individual PLs within each corridor segment are modeled using a local analysis and catenary curve fitting. McLaughlin [122] computes local affine models of the sections of transmission lines within each ellipsoidal neighborhood. Starting with an arbitrarily chosen local affine model, other local affine models on the same span are detected by progressively refining the estimate of the catenary parameters. Individual transmission line spans are then modeled by aggregating all the local affine models determined for the corresponding span. Jwa *et al.* [125] propose a

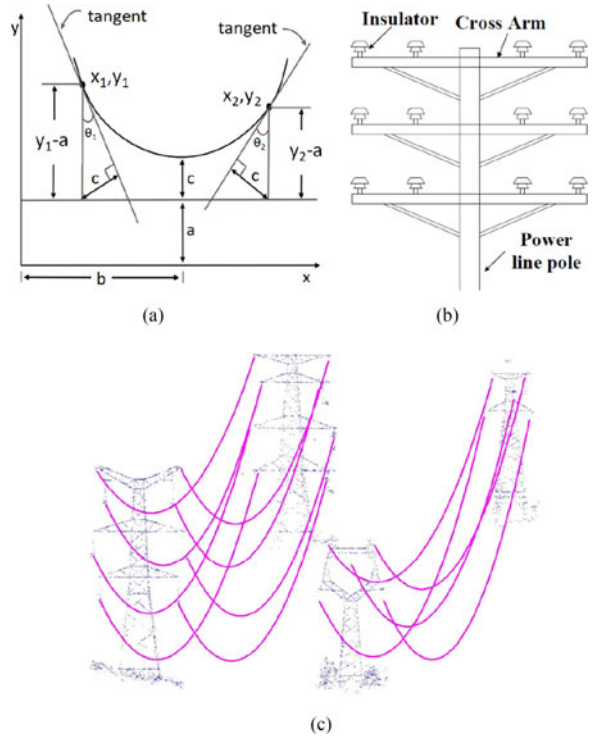


Fig. 5. (a) Example of catenary curve(Image courtesy: [122]), (b) prototype of multiple parallel PL structure (Image courtesy: [123]), and (c) PLs reconstructed using the technique in [119] (Image courtesy: [119]).

new method to detect and group PL points, called voxel-based piece-wise line detector, that reconstructs a PL by grouping similar features at different levels of information (i.e., points, segments, and lines). Reconstruction of a single PL model is then accomplished by applying a nonlinear adjustment for estimating catenary curve parameters to a piece wisely segmented voxel space. Jwa and Sohn [121] propose a piecewise catenary curve model growing algorithm for PL reconstruction. In their approach, potential candidate points of the PL, extracted via linear feature extraction operators, are used as the seed points to generate primitive models. These models are then progressively expanded through hypothetical model production and optimal model selection. Based on the piecewise model growing [121], Sohn *et al.* [126] employ an MRF classifier to extract PL features from the point clouds and localize pylons through investigating *a priori* knowledge of contextual relationships between PL and pylon. PL spans are modeled with catenary curves, after pylon localization is accomplished.

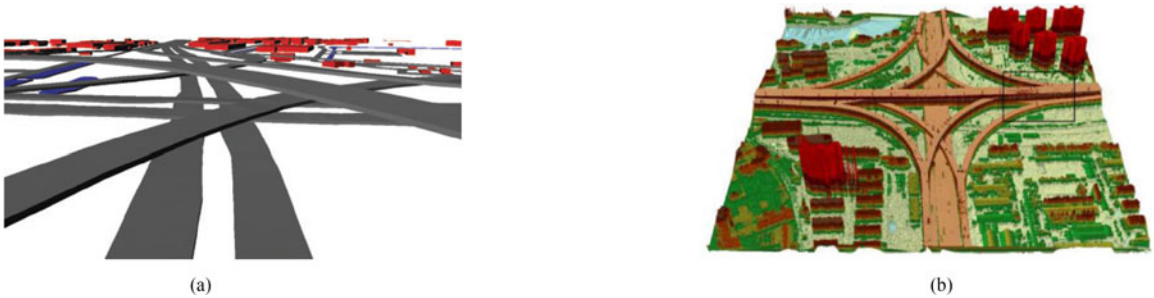
2) Other Parametric Models: Liang *et al.* [127] consider the Euclidean distance between the points to group them into individual PLs and then create the corresponding 3-D model by fitting quadratic polynomials to grouped points. In [128], Ritter and Benger employ eigenanalysis of a point distribution tensor to derive the major eigenvector and, thus, the dominant orientation of the distribution. The PL is then reconstructed by following the streamlines, which are computed along the dominant eigenvector field. In 2016, Guo *et al.* [119] present a RANSAC-based method that exploits the contextual pylon information and distribution property of PL group between two neighboring pylons to improve the PL reconstruction results. The main assumption is that PLs in one group are nearly parallel with the same direction and sag, and this information can be effectively used for setting initial parameters for the reconstruction. Cheng *et al.* [123] observe that high-rise buildings in the urban environment restrict a close distance (60–150 m) aerial scanning of PLs, which in turn affects the point sampling of the ALS. Moreover, ALS systems, obtaining data at a constant off-nadir angle, detect only the PL information of first cross-arm and, therefore, misses other parallel PLs [see Fig. 5(b) for a structure of parallel PLs]. To address these issues, the Cheng *et al.* [123] introduce a method based on MLS, in which the initial clustering and clustering recovery procedures are iteratively performed to identify each PL points and parabolic curves are then fitted to these points. In general, the accuracy of the PL modeling techniques is dependent on the scanning quality, which in turn is influenced by external factors, such as wind, electrical load, and solar radiation [129].

B. Roads and Bridges

Tons of work exists in the detection or extraction of road inventory, such as traffic signs, lamp poles, or road networks. Since, this study is aimed at the modeling part of the urban reconstruction pipeline, we recommend the reader to the recent surveys [130], [131] to get an in-depth treatment on the detection or extraction part.

Accurate, up-to-date and detailed 3-D models of roads are important for a number of geospatial applications, such as traffic monitoring, landscape design, and driver assistance systems. Commercial maps used for most of the existing road navigation systems are based on road center lines represented by polylines, which clearly lack accuracy, contents, and completeness at the subroad level. Two-dimensional roads are incapable of capturing complex road scenes, such as overpasses and multilevel road systems. Consequently, research in this direction started focusing on reconstructing 3-D road models that can capture complex road scenes. 3-D roads can be reconstructed using procedural rules [132], GIS data [133], aerial images [134], or LiDAR data [135] (An example reconstruction is shown in Fig. 6(a)). We focus on various LiDAR-based road reconstruction techniques in this section.

Vosselman [136] proposed a method to reconstruct single-layer road models by fusing LiDAR data and cadastral maps. In this method, LiDAR points lying inside manually edited road polygons are used to generate an initial road surface with a triangular irregular network (TIN) structure. Assuming that the road cross section exists without slope, curvature, or torsion, road surfaces are smoothed. The downside of this method is that bridges are not accurately modeled, i.e., appear as embedded canals. Using TIN structure for road surface modeling is quite common (e.g., [141]). Yu *et al.* [137] present a data acquisition and processing framework that incorporates multimodality sensors for road surface mapping and inspection applications. The proposed road surface modeling method provides detailed 3-D models that possess accurate depth information for characterization and visualization of road cracks of up to the size 10 cm in length and width. Road crack modeling with depth information seems to be a significant improvement over contemporary commercial video-based vision systems. In another study [142], Denis *et al.* describe a pipeline for extracting and modeling urban roads as 3-D surfaces, using laser points acquired from an MLS system and existing 3-D road axes derived from aerial imagery. Hervieu and Soheilian [138] propose a semiautomatic system for road and pavement modeling.



(a) (b)

Fig. 6. Examples of highways and bridges reconstructed by various algorithms. (a) Reconstruction of four levels of highways [Image courtesy [135]] and (b) interchange bridge model [Image courtesy [140]].

Their method first detects curbs and curb ramp, and the detection results are then used to model surfaces of both roads and walking areas. The method performs reasonably well under occlusions.

Three-dimensional interchange bridge models are extremely important for accurate navigation in urban environments. Elberink and Vosselman [135] propose an automatic method for modeling complex highway interchanges based on the fusion of LiDAR and 2-D topographic map data. Error propagation and measurement reference [143] are used in this method for quality assessment of 3-D model reconstruction. Sithole and Vosselman [139] use seed points to detect bridges based on ALS data. Fusion of large-scale topographic maps and ALS to reconstruct 3-D road models has been explored in [144]. Recently, Cheng *et al.* [140] propose a framework that divides an entire interchange bridge into many simple *structure units*, and then derive a complete 3-D model (refer to Fig. 6(b)) through reconstruction. The whole framework consists of point cloud extraction, connectivity-based segmentation, determination of structure units, occlusion detection and restoration, and 3-D modeling.

C. Summary

Table III summarizes the major techniques addressing the reconstruction of 3-D urban objects including trees, PLs, roads, and bridges from LiDAR data. From the Table III, one can easily make the following observations:

1) PL modeling techniques normally use ALS data for reconstruction and generate parametric representations for the PLs. The most common representations of PLs are catenary curves.
2) Different types of model representations including TIN, Delaunay mesh, or quadratic polynomials are used for road surfaces. Moreover, inputs to the road or bridge modeling techniques are either ALS or MLS.

According to Cheng *et al.* [140], research into 3-D interchange bridge reconstruction has a long way to go. Most of the research aims to reconstruct bridges with simple structures, and hardly account for 3-D reconstruction of large multilayer interchange bridges. Moreover, the techniques to restore obscured structures of large interchange bridges has not been sufficiently addressed. Very few works on 3-D interchange bridge reconstruction are based on ALS. There could be multiple reasons for this. The ALS data represents an abstraction of landscape, where the landscape is composed of the bare earth and objects. Objects are further categorized into "detached" (free of the bare earth and rise vertically, e.g., buildings or trees) or "attached" (connected to the bare earth, e.g., bridges). In reality, distinguishing the "attached" objects such as bridges from the bare earth is a challenge work, therefore, an accurate reconstruction of such objects becomes a hard problem. Another possible reason for this could be the occlusions occurring underneath the upper objects, leading to gaps in the airborne data [135]. In the presence of such occlusions, the auxiliary data (e.g., topographic maps) can substantially improve the reconstruction results. However, auxiliary data may not be always available. Hence, future research efforts have to account for such details while developing frameworks for the modeling of complex bridge structures.

V. Reconstruction of Free-Form Urban Objects

Free-form objects such as statues, fountains, and buildings with developable surfaces are quite common in urban environments. Typically, generic surface reconstruction algorithms such as the Poisson method [145] can reconstruct free-form objects with reasonable accuracy. There exist numerous other surface reconstruction techniques that are capable of addressing free-form objects. The literature on surface reconstruction is so vast that a review of all the papers is clearly beyond the scope of this paper. Moreover, we are interested in urban free-form object reconstruction that are often different from the objects being targeted by many surface reconstruction techniques in terms of size, structure, etc. Thus, this section is meant to introduce and analyze the major surface reconstruction techniques prevalent in the computer graphics and computational geometry domain to the researchers and practitioners of photogrammetry and remote sensing communities.

Most of the current surface reconstruction methods can be classified into two major categories: techniques that use implicit functions to represent surfaces and explicit methods that use Delaunay-/Voronoi-based triangular meshes to represent the surfaces. We review implicit and explicit methods for surface reconstruction, respectively, in Sections V-A and V-B. An expository treatment of implicit surface reconstruction techniques can be found in a recent survey by Berger *et al.* [146].

A. Implicit Functions

Given a finite set of distinct points P on a 3-D surface M, implicit methods attempt to formulate a smooth function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ such that the zero level set of f, i.e., $Z(f) = \{p \in \mathbb{R}^3 \mid f(p) = 0\}$ are close to P (refer to Fig. 7 for an illustration). The function f is determined based on the information derived from the point cloud P and the associated set of normals. Implicit surfaces can be directly visualized by using a ray tracer [147] or

TABLE III
LIST OF MAJOR TECHNIQUES ADDRESSING THE RECONSTRUCTION OF DIGITAL TREES, PLS AND ROADS/BRIDGES FROM LIDAR

Technique	Input	Interaction	Model Rep.	Strengths/Robustness	Challenges/Limitations
			Trees		
Xu *et al.* [96]	TLS	Semi-automatic	Mesh	Small branches and twigs from incomplete scan, tree allometry for branch dimensions	Parameters specific to tree species, visual differences from the original tree
Linvy *et al.* [97]	TLS	Automatic	Parametric	Sparse, incomplete and noisy data, multiple overlapping trees	Occluded branches of crowns, Random placement of leaves,
Wang *et al.* [107]	TLS	Automatic	Mesh	Reconstructs from incomplete scans, varying point density	Branch intersection, Completely occluded twigs
Zhu *et al.* [98]	TLS	Automatic	Mesh	Intermediate LOD model	Coarse branching structure
Linvy *et al.* [72]	TLS	Interactive	Cylindrical	Easy editing of tree models, large sets of trees	No tiny branches for unilateral scans, species library creation
Zhang *et al.* [93]	TLS	Semi-automatic	Mesh	Multilayer and consistent representation of trees	Distorted or wired branches, trees with large and thick leaves
Cote et al. [71]	TLS			Low spatial/angular resolutions, non-ideal conditions(wind & occlusion)	Termination criteria of foliage addition
Cote *et al.* [100]	TLS	Semi-automatic	L-architect	Occlusion, over-sampling, wind	Running time
Dai *et al.* [101]	TLS	Semi-automatic	Cylindrical	Heavily blocked branches and twig modeling	Selection of root point
Aiteanu and Klein [102]	TLS	Automatic	Cylindrical	Dense & sparse data	Running time, random orientations of leaves
pfeifer *et al.* [105]	TLS	Automatic	Mesh	Coniferous trees, Scanned in winter	Unrealistic outer hulls, Unable to reconstruct occluded stems
Yan *et al.* [73]	TLS	Automatic	Cylindrical	Complex branch structures	Reasonable timing, holes due to occlusions
Cheng *et al.* [106]	TLS	Interactive	Cylindrical	Works on single scan	Wrong radii due to incomplete data
Bucks *et al.* [110]	TLS	Automatic	Skeleton	Theoretical validation, under-sampling and varying density	Octree resolution
Schilling and Mass [111]	TLS	Automatic	3-D polylines	Explicit tree detection is not required	Partial tree skeleton
Raumonen *et al.* [74]	TLS	Automatic	QSM	Quantitative analysis of tree structure & size	Missing data are not reconstructed, noise
Li *et al.* [115]	TLS	Semi-automatic	Mesh	Trunks and ground reconstruction	Overlapping fissures in crown assignment
			PL		
Melzer *et al.* [124]	ALS	Automatic	Parametric	First use of catenary in PL	Corridor concatenation
McLaughlin [122]	ALS	Automatic	Parametric	Absence of false positives	Sparse data
Jwa *et al.* [125]	ALS	Automatic	Parametric	Robustness to vegetation interference	–
Liang *et al.* [127]	ALS	Automatic	Parametric	Simple method	Sparsity, breakage points
Ritter and Benger [128]	ALS	Semi-automatic	Parametric	–	neighborhood radius, weighing functions
Jwa and Sohn [121]	ALS	Automatic	Parametric	Reconstructs complex scenes, bundle wires	Initial growing cube dependence
Sohn *et al.* [126]	ALS	Automatic	Parametric	Complex scenes and Pylon reconstruction	–
Cheng *et al.* [123]	MLS	Automatic	Parametric	Addresses multiple parallel PL	Threshold setting
Guo *et al.* [119]	ALS	Automatic	Parametric	Absence of gaps in the span	Sparse data
			Roads/bridges		
Vosselman [136]	ALS	Automatic	Quadratic polynomial	Earliest attempt	Assumptions on curvature, slope and torsion, flaws in bridge models
Yu *et al.* [137]	MLS	Semi-automatic	Delaunay mesh	Crack detection & modeling	
Hervieu and Soheilian [138]	MLS	Semi-automatic	Quadratic polynomial	Handles occlusions	Fail at curb ramps and occlusions in bends
Elberink and Vosselman [135]	ALS	Automatic	TIN	Handles occluded bridges	
Sithole and Vosselman [139]	ALS	Automatic	–	adaptable to different designs, blind to bridge shapes	False bases for bridges over deep water bodies
Cheng et al. [140]	ALS	Automatic		Complete and geometrically accurate complex bridges	Decomposition into structure units always not possible

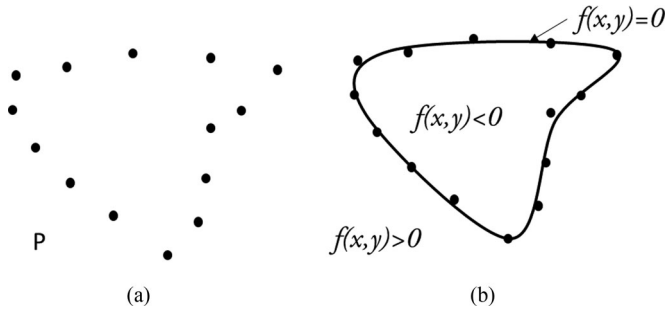


Fig. 7. Illustration of an implicit function in 2-D. (a) Point set. (b) Implicit curve.

Fig. 8. RBF approximation [152] of a noisy LiDAR scan of a fountain at Santa Barbara. The statue is approximately 2 m × 5 m. Image courtesy [152].

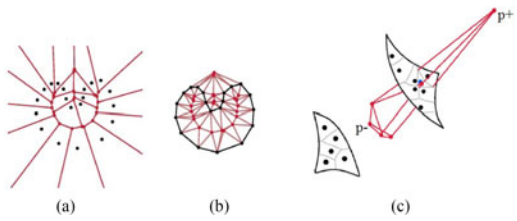


Fig. 9. Illustration of crust in 2-D. (a) Point set S, with the Voronoi diagram (red colored). (b) Crust of S(shown in black). (c) Poles (p^+ and p^-) illustrated in 3-D for a sample point (blue colored) [Image courtesy: [171]].

by polygonizing it using the well-known marching cube [148] algorithm. Implicit reconstruction methods mainly differ in the formulation of the function f. A typical choice for f is a signed distance function (SDF). SDF for an arbitrary point q is the distance between q and its nearest surface point and the sign component indicates the location of q with respect to the surface, i.e, inside or outside. One of the early and pioneering works in implicit surface reconstruction is the tangent plane estimation method by Hoppe *et al.* [149]. The method employs principal component analysis to estimate the normals and tangent planes at each data point, and then employs a neighborhood graph search to unify the inside/outside directions. The SDF is then computed using the oriented tangent planes. Another method using SDF is volumetric range image processing [150], which employs a volumetric method based on a cumulative SDF to average the range surfaces in a regular grid. While both these methods are robust to noise, they have difficulty in dealing with large-scale data. This is mainly because of the requirement that the implicit function representation has to reside in the memory [151].

1) Radial Basis Functions (RBFs): RBFs are well suited for the smooth interpolation of scattered data. In [152], Carr *et al.* estimate the function f as a linear sum of weighted and shifted radial functions, i.e., $f(p) = \sum_{i=1}^{n} w_i \phi(\|p - c_i\|)$, where the weights w_i are calculated by applying surface constraints at points c_i and then solving the resultant linear system. The choices for the basic function ϕ include Gaussian ($\phi(r) = \exp(-cr^2)$), multiquadric ($\phi(r) = \sqrt{c^2 + r^2}$), polyharmonic ($\phi(r) = r$ or $\phi(r) = r^2$), and thin-plate spline ($\phi(r) = r^2 \log r$). Although compactly supported basic functions such as Gaussian are faster, they are found to produce undesirable artifacts for irregular and nonuniformly sampled data. On the contrary, polyharmonic functions are a good choice for 3-D representation due to their energy minimization properties [152]. However, since the harmonic functions are noncompact, solving the linear system becomes computationally expensive. In [152], the computational performance of RBF interpolation has been enhanced by employing a fast multipole method to evaluate the RBF and a greedy algorithm for RBF fitting. The method is found to reconstruct free-form urban objects from a noisy LiDAR scan with reasonable accuracy, as shown in Fig. 8. Another approach in this direction by Turk and O'Brien [153] computes the implicit surface as an RBF that minimizes thin plate energy subject to a set of interior and exterior constraints.

2) Poisson Methods: The Poisson surface reconstruction method [145] solves for an indicator function for the solid, whose gradient best approximates the normal field N, i.e., $F = \text{argmin}_S \parallel \nabla_S - N \parallel_2^2$. This minimization problem leads to a Poisson equation, which is solved by a locally supported RBF on an adaptive octree. This is one of the most popular methods for surface reconstruction due to its scalability and efficiency. Moreover, Poisson reconstruction generates satisfactory results, preserving even minute details. In [154], a streaming approach for Poisson reconstruction has been proposed, which allows the algorithm to handle massive datasets on the order of hundreds of millions of point samples. In other related works, Fourier [155] and wavelet [151] basis have been employed to improve the computational overload of solving Poisson equations. Recently, Kazhdan *et al.* [156] generalized the underlying mathematical framework of [145] to screened Poisson equation along with numerous algorithmic improvements.

3) Moving LS Methods: In moving LS methods, any points near the point set are projected onto the surface and the underlying surface is defined by the points projected onto themselves. The moving LS projection [157] method first defines a local reference domain H for a point r to be projected and then fits a local polynomial approximation g to the weighted input points. Here, the weight for each input sample p_i is a function of its distance to the projected r on H. Once g is computed, the projection of r onto g represents the moving LS projection of r. Several approaches including moving-based surface resampling [158], provable moving LS surfaces [159], moving LS fitting on algebraic spheres [160], progressive point set surfaces [161], moving LS-based sharp feature reconstruction [162] have been proposed based on the moving LS projection idea. Moving LS approaches have an intrinsic capability to handle noisy point sets and hence can be used in the reconstruction of objects from noisy urban scans.

4) Multilevel Partition of Unity (MPU) Implicits: Ohtake *et al.* [163] present an MPU that basically decomposes the domain into subdomains and locally fits piece-wise quadratic functions to the data. A global approximation is then created by blending the local quadratic functions using a weighting function (the partitions of unity). In [164], an improvement of the

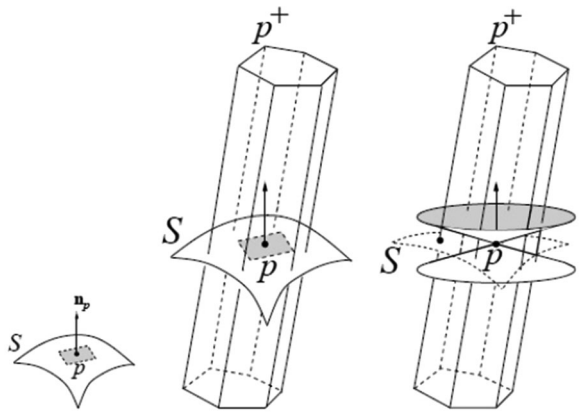


Fig. 10. Cocone Illustration. Voronoi cell V_p of a sample point p elongated along the direction of normal at p (n_p). Cocone is the region in V_p between two cones at p (extreme right) [Image courtesy: [175]].

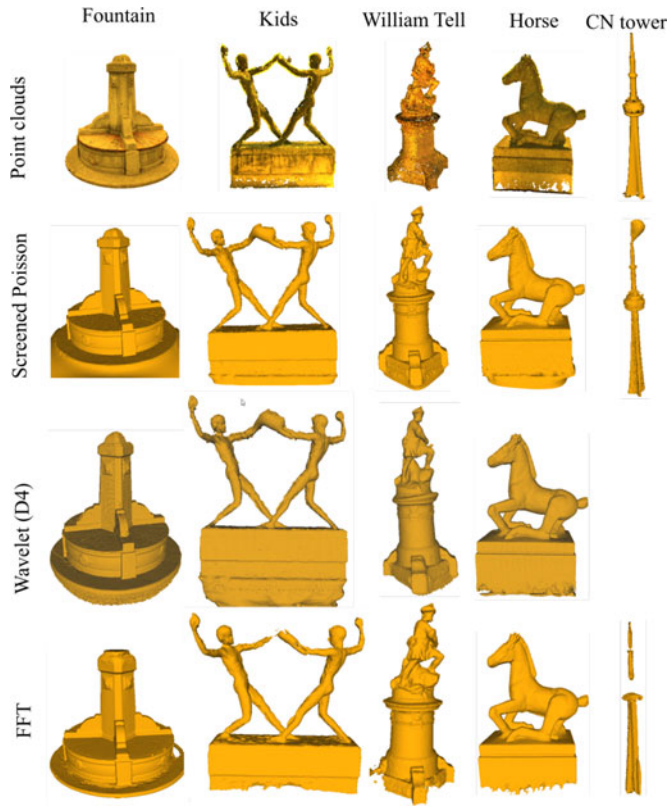


Fig. 11. Urban free-form models reconstructed by various implicit surface reconstruction techniques. In our experiment, wavelet (D4) method generated incomparable results for CN tower model and hence the corresponding result is omitted from the figure.

original MPU algorithm that combines an adaptive partition of unity with LS RBFs has been proposed. Gaussian process regression [165] is another way of obtaining the approximations to the local shapes in the MPU generated subdomains. In general, MPU implicits are simple and very fast, which makes it one of the obvious choices for the reconstruction of large free-form objects. To evaluate the performance of different implicit methods in urban modeling scenario, we experimented with a few urban models, and the reconstructed results are presented in Fig. 11.

B. Computational Geometry Approaches

There exist several Delaunay-/Voronoi-based surface reconstruction algorithms in the computational geometry domain [166], [167], most of which provide theoretical guarantees on the quality of reconstructed output given a specific sampling model. The underlying principle is that, when the point sampling is dense enough and free of noise and outliers, the neighboring points on the surface will also be neighbors in the space and can be captured with the help of geometric structures such as Delaunay triangulations (DT) or Voronoi diagrams. Research on Delaunay–based surface reconstruction starts with the sculpture algorithm by Boissonnat [168]. Later, 3-D α-shapes were proposed by Edelsbrunner and Mücke [169]. Following these two pioneering works, many computational geometry approaches are proposed, focusing on topology reconstruction and providing guarantees on the quality of output under a specific sampling model, for e.g., ϵ-samples [170]. We review the relevant methods in the subsequent paragraphs.

1) Power Crust: The Delaunay/Voronoi model together with ϵ-sampling (or a related sampling model) has been extensively researched. In the ϵ-sampling model [170], the point density depends on the local feature size, where a local feature size at a point p on the curve Σ, LFS(p) is the distance from p to the closest point on the medial axis of Σ. This sampling model is capable of quantifying the local LoD at each point on a smooth curve. Especially, the curve portions encapsulating more details are sampled densely and other portions are sampled sparsely. Hence, it generates a nonuniform sampling. Under the ϵ-sampling model, the first algorithm to provide theoretical guarantees for the reconstruction of smooth surfaces is crust [170], [172]. In two dimensions, crust consists of edges from the DT of input samples that can be enclosed by circles that are empty of samples points and Voronoi vertices [see Fig. 9 (b)]. The extension of crust to 3-D, uses the basic idea that Voronoi cells of points on the surface tend to be elongated in a direction perpendicular to the inferred surface. It uses *poles*, which are the farthest Voronoi vertices of a sample's Voronoi cell [see Fig. 9(c)]. For each sample, the algorithm picks two farthest Voronoi vertices as poles. The crust consists of all Delaunay triangles whose circumspeheres are empty of samples and poles. The power crust [173] (Fig. 12, upper row) is an extension of crust, more robust to realistic inputs, which instead relies on the power diagram: a weighted Voronoi diagram of the poles. The powercrust has been modified to improve the robustness to noise [174].

2) Cocone Family: Cocone algorithms use poles to compare facets normal with point normals (vector from the sample point to the pole) [176]. Cocone is the complement of a double cone centered at a sample point p with an opening angle $\frac{3\pi}{8}$ around the axis aligned with the normal at p [175] (see Fig. 10). Cocone algorithms proceeds as follows: Each sample point p, chooses a set of Delaunay triangles whose dual Voronoi edges intersect with the cocone defined at p. All such triangles chosen by all the samples represent candidate triangles and the final manifold is extracted out of the candidate triangles. Each candidate triangle has the property that its facet normal is oriented approximately in the same direction as the normals at its three sample points [176]. Several improvements over the basic cocone algorithm have been introduced. SuperCocone [177] employs an octree-based division of the point clouds to reduce the running time to compute DT thereby enhancing the performance of the entire

Fig. 12. Reconstruction results of different Delaunay-/Voronoi-based algorithm.

cocone algorithm. Dey and Goswami propose the TightCocone algorithm that produces a watertight surface that is free of holes [175]. The RobustCocone (Fig. 12, bottom row) generalizes the definition of poles to cope with a specific noise model [178]. A detailed discussion of Delaunay-based curve and surface reconstruction algorithms with mathematical analysis can be found in [167].

3) Sculpting Techniques: In computational geometry, sculpting refers to the process of creating shapes or surfaces through repeated elimination of tetrahedra from an initial tetrahedral mesh. The elimination of a tetrahedron T_i from the intermediate tetrahedral mesh is subject to the fundamental rules of tetrahedral removal and a few other algorithm specific constraints. Tetrahedral removal rules consider only the tetrahedra with one or two boundary faces for the removal. For a one boundary faced tetrahedron, the vertex opposite to the boundary face should lie in the interior of the intermediate surface. If the tetrahedron has two boundary faces, then the edge opposite to the edge being shared by the two boundary faces should lie in the interior of the intermediate surface. Basically, these rules are designed to produce a surface that is topologically equivalent to a sphere.

Sculpture algorithm [168] starts by constructing Delaunay tetrahedralization of the input points and iteratively removes tetrahedra from the intermediate boundary representation B, subjected to some fundamental rules and constructs the polyhedral approximation of the object implicitly represented by S. In each iteration, the noneliminated tetrahedra in B are visited in a certain order induced by a value $V(T_i)$ associated with each tetrahedron T_i. $V(T_i)$ is defined as the maximum distance between the faces of a tetrahedron, T_i on B and the associated parts of the circumsphere of T_i. Furthermore, a tetrahedron T_i on B may be removed, if it satisfies the tetrahedral removal rules. The algorithm terminates when no more tetrahedra can be removed from B and the set of noneliminated tetrahedra constitutes the polyhedral approximation of P. Veltkamp proposed a constriction algorithm based on two-parameter geometric graph, called γ-neighborhood graph [179], that generates a polyhedronization of S. Veltkamp introduced $\gamma(c_0^{'}, c_1^{'})$, which is a variant of a γ-neighborhood graph by providing a range of values for the parameters c_0 and c_1 (i.e., $c_0^{'} \in [c_0^0, c_0^1]$ and $c_0^{'} \in [c_1^0, c_1^1]$). DT is a member of the γ-neighborhood family with parameters $c_0 \in [-1, 1]$ and $c_1 \in [0, 1]$. Veltkamp's constriction algorithm starts with Del(S) ($\gamma([-1, 1], [0, 1])$) and uses the fundamental rules of tetrahedra removal to construct a pruned γ-graph.

Hybrid sculpting method [180] proposed by Attene *et al.* uses geometric graphs, such as Euclidean minimum spanning tree and extended Gabriel hypergraph (EGH) and the concept of "pseudoprism" to reconstruct a closed interpolating surface with zero or more through holes. The EGH is introduced in reconstruction in order to locate the Delaunay triangles, which have a high probability of belonging to the original surface. Gezahegne proposes a constrained sculpting [181] approach with topological guarantee for the reconstruction of a piece-wise linear watertight approximation of the original surface. The algorithm is a hybrid of sculpture [168] and cocone [176] algorithms. A recent sculpting approach namely shape-hull [182] eliminates boundary tetrahedra subject to the tetrahedral removal rules and tetrahedral circumcenter locations. For ϵ-sampled divergent concave curves and surfaces, shape-hull is guaranteed to generate a water-tight, piece-wise approximation to the original object.

The sculpting algorithms are simple, relatively easy to implement, and guarantee to generate an output topologically equivalent to the original surface. However, the algorithms such as sculpture [168] suffers from some serious drawbacks such as existence of deadlocked tetrahedra and sensitivity to undersampling [181]. The constriction algorithm [179] is found to generate unaesthetic surfaces having long and thin tetrahedra [180]. Hybrid sculpting [180] has a simple selection criterion, i.e., tetrahedron with the longest edge, which may not capture the geometric property of a tetrahedron in its entirety. Algorithms such as hybrid sculpting and constraint sculpting [181] cycle between different stages to complete the reconstruction process. An important factor determining the geometrical accuracy of the sculpting method is the selection criteria, i.e., order of the tetrahedral removal. As opposed to a random selection of the tetrahedra for removal, criteria such as circumradius [182], γ-indicator [179], or the longest edge length of the boundary tetrahedron [180] are found to produce good results when combined with an appropriate algorithm. In practice, it is unknown what criterion will work well for a wide range of point sets. Hence, choosing the correct criterion represents one of the intriguing challenges in sculpting-based methods.

4) Other Methods: Topological equivalence between the reconstructed and the original surfaces has been ensured in many other Delaunay-based techniques. Gopi *et al.* [183] proposed a tangent plane based method that is guaranteed to construct a model homeomorphic to the original surface when a locally uniform sampling condition holds. Edelsbrunner's WRAP [184] algorithm is built upon the concepts of flow and stable manifolds where the *flow relation* has been formulated on the set of Delaunay simplices. Other Delaunay-based techniques including the greedy algorithm [185], ball pivoting [186], regular interpolant method [187], geometric convection [188], and Delaunay-based region growing [189] do not provide theoretical guarantees, however, are found to perform reasonably well for a range of point sets. Most of these methods are rather simplistic in nature and computationally efficient.

TABLE IV
SUBSET OF WELL-KNOWN ALGORITHMS FOR FREE-FORM SURFACE RECONSTRUCTION AND THEIR RELEVANT PROPERTIES

Algorithm	Input normals	Reconstruction type	Noise handling	Incomplete data	Large data	Parameters
FastRBF [152]	✓	Implicit	✓	✓	✓	d
Screened Poisson [156]	✓	Implicit	✓	✓	✓	d
Wavelet [151]	✓	Implicit	✓	✓	✓	d
FFT [155]	✓	Implicit	✓	✓	✓	d
MPU [163]	✓	Implicit	✓	✓	✓	d
MLS [157]	×	Implicit	✓	✓	✓	d
Powercrust [173]	×	Delaunay	×	Fragile	✓ (slow)	t, w, R
Robustcocone [178]	×	Delaunay	✓	Fragile	✓ (slow)	$thif, thff, bbr$

C. Summary

1) Normals and Noise: In general, Delaunay-/Voronoi-based approaches for surface reconstruction are found to produce good approximations for dense uniformly sampled point clouds. Work on input scans without point normals often computes output meshes with a complexity in the order of the input point set size. On the contrary, most of the implicit-based techniques use acquired or estimated point normals to facilitate the reconstruction process, and are robust against noise. Delaunay-based methods fail to reconstruct correct surfaces in the presence of noise. However, there are a few methods, such as modified powercrust [174] and robustcocone [178], that perform reasonably well under specific noise models.

2) Parameters: A vast majority of surface reconstruction algorithms rely on one or more parameters to reconstruct the final surface. Setting the right parameter for each model poses a real challenge, especially for different types of noisy and incomplete data. Since the isosurface is extracted using marching cubes on octrees, an appropriate octree depth (d) has to be determined and preset in the case of implicit methods. A more detailed mesh reconstruction is obtained for larger depth values, however, at the expense of increased computational time. Powercrust has three essential parameters, the angle between deep intersecting balls in the first (t) and second passes (w) and the sampling density (R). Two angle parameters ($thif$ and $thff$) and a circumball size (bbr) are required to be set for robustocone. We compare the parameters and properties of well-known free-form surface reconstruction algorithms in Table IV.

3) Accuracy: To verify the accuracy of various reconstruction techniques, we conducted an experiment in which the reconstructed mesh was evaluated against ground truth in terms of Hausdorff distance. Experimental data were taken from the statue repository, computer graphics, and geometry lab of EPFL[5] and an MLS scan of Toronto (CN tower). Each model has the images, point cloud, and a reconstructed mesh as shown in Figs. 11 and 12. The reconstructed meshes provided on the website have been considered as the reference mesh (ground truth) for our experiments. Fig. 13 shows the evaluation results using root mean square (rms) of Hausdorff distances computed with metro tool [190]. For the tested urban models, the accuracy (in terms of rms) of the results found to vary between different models. Fast Fourier transform (FFT) [155] generated a close approximation to the horse reference model, whereas, for William statue, wavelet [151] method reconstructed a better approximation. Compared to the horse model, all the methods generated reasonably close approximations in the case of William statue.

[5]http://lgg.epfl.ch/statues_dataset.php

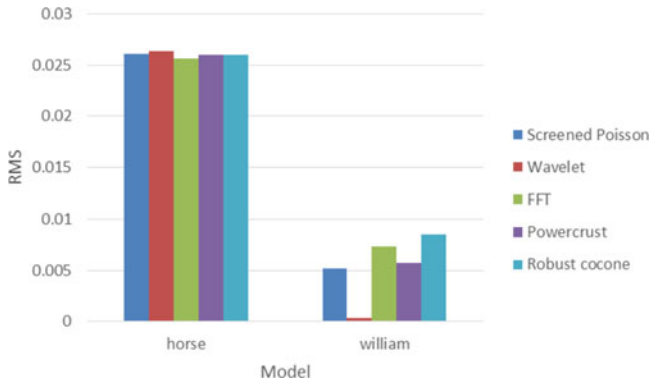


Fig. 13. Bar chart representing the the rms distances between the reconstructed and the reference meshes.

TABLE V
RUNNING TIMES (IN SECONDS) OF DIFFERENT SURFACE RECONSTRUCTION TECHNIQUES FOR THE TESTED MODELS

Model	Scan size	Poisson	Wavelet	FFT	Robust cocone
CN tower	694 042	1.74	7.21	14.01	879.16
Fountain	1 829 707	8.4	20.56	14.93	–
Kids	208 321	5.74	8.73	14.04	133.67
William	167 614	2.92	3.34	14.07	52.88
Horse	1 199 020	7.28	5.24	14.44	1011.8

In the experiment, Robust cocone exited while constructing DT1 for fountain model.

This clearly suggests that the accuracy depends on the object surface characteristics and the input data. In particular, to the urban context and the conducted experiment, the implicit method outperformed the computational geometry method in terms of reconstruction accuracy.

4) Time and Space: Table V reports the computational time incurred by various surface reconstruction techniques for the tested models. Comparatively, implicit methods are very fast in estimating the surface. As a result, implicit methods can be preferred over Delaunay-based techniques for reconstructing large sized urban free-form objects that include curved buildings, statues, or fountains.

VI. CONCLUDING REMARKS

Rooftop Modeling: Most of the existing methods employ limited assumptions and geometric constraints that may not be suitable for all complex buildings in real-world environments. For instance, boundaries are commonly fitted by piece-wise linear

segments no matter what the real boundary type is. We believe that multiple assumptions and constraints are useful to model complex buildings. Providing a comparison between the flexibility of the reconstruction methods and the quality of the reconstructed building models is really difficult and hence represents a topic to explore. Integration of semantic information into the reconstruction procedure is another possible research direction as most of the existing reconstruction methods do not use semantic information. A very recent work [191] employs semantic labeling information for 3-D building reconstruction. The approach to make a unified framework to model buildings with many different types of rooftops is also an interesting topic to explore.

Detailed Façades: Highly successful web-based 3-D visualization applications, such as Google Earth, Bing maps, and Apple maps, encourage urban modeling researchers to focus on 3-D models with rich geometric details. Many urban applications, such as 3-D navigation or emergency response training, require highly detailed 3-D models. Recent efforts in this direction use interactive systems with predesigned 3-D [192] or 2-D [87] templates to model urban façades. Though impressive 3-D models are produced by these interactive systems, the frameworks are designed under Manhattan geometry for symmetric façades. Reconstruction of non-Manhattan structures and nonsymmetric façades is highly desirable. Further investigations have to be carried out to enhance the quality of the visualization without compromising the algorithmic scalability and computational performance.

Level of Details: Creating 3-D models of a region with multiple LoDs [89], [193] is another potential direction to explore. LoDs in building models are often determined by various factors, such as data acquisition cost, labor expense, and most importantly, the target applications. For instance, a 3-D outdoor navigation application may require simple building blocks, whereas applications such as emergency response training demand more detailed building blocks with, for example, the emergency exit doors. Recent research efforts in this direction focus on real-time rendering of multiple LoDs. Even though LoDs embedded with the models increases usability, it requires large data storage. Hence, a very intriguing question that is being studied these days is how to represent multiple LoDs with minimum possible storage. A related topic is urban-scale space exploration [194] that deals with exploring the solution space across scales and automatically selecting the best structure for an application. Such exploration with respect to the geometric primitives and the structural relationships will help in finding a tradeoff between the complexity (enumerating the structural rules and the corresponding parameters) and the accuracy of the model creation.

Urban Dynamics and Evolution: Modeling of urban environments should also account for the constant evolution of urban structures over time. The change can be over short intervals, e.g., moving cars or walking pedestrians or long term including various architectural styles introduced to buildings as part of renovations, newly built buildings, seasonal changes of trees, etc. Updating the city models addressed in the literature seems a cumbersome and time consuming task for incorporating such urban dynamics. A challenging yet rewarding task in urban dynamics is to predict and anticipate the geometric evolution of urban structures and apparently design the system to reflect such changes [194]. Another promising direction is to design systems reflecting the urban structure evolution at multiple levels of detail by integrating the spaceborne TomoSAR point clouds [195], [196]. More precisely, at a world-scale level, the high-resolution TomoSAR point clouds can represent the whole world revolutions and this revolutions can be further refined in local urban scale by collectively using the ALS, MLS, and TLS point clouds.

Crowdsourcing: An obvious future direction in urban reconstruction is to increase the level of granularity, i.e., shifting focus toward developing large geographical regions including nations. However, this demands huge data management, several man-years to acquire the input data and considerable computational time. Crowdsourcing on web-based 3-D urban modeling and visualization systems [197] would be an appropriate choice to realize such large systems. In a nutshell, users who work on particular classes of urban objects of certain geographical area, e.g., trees or buildings may upload the reconstructed models into a common database under the corresponding city or nation. The high-quality reconstructed models uploaded to the web-based system can be automatically or manually placed at the appropriate location of the city and can be used for various applications. Although, such crowdsourcing urban systems facilitates many urban applications, e.g., autonomous driving and urban planning based on deep learning framework, its 3-D visualization calls for exemplary research on highly responsive and immersive web-based user interfaces and high Internet bandwidth.

REFERENCES

[1] P. Musialski, P. Wonka, D. G. Aliaga, M. Wimmer, L. Gool, and W. Purgathofer, "A survey of urban reconstruction," *Comput. Graphics Forum*, vol. 32, pp. 146–177, 2013.

[2] C. A. Vanegas, D. G. Aliaga, P. Wonka, P. Müller, P. Waddell, and B. Watson, "Modeling the appearance and behavior of urban spaces," *Comput. Graphics Forum*, vol. 29, no. 1, pp. 25–42, 2010.

[3] R. Wang, "3D building modeling using images and LiDAR: A review," *Int. J. Image Data Fusion*, vol. 4, no. 4, pp. 273–292, 2013.

[4] M. Berger *et al.*, "A survey of surface reconstruction from point clouds," *Comput. Grapics Forum*, vol 36, no. 1, pp. 301–329, 2017.

[5] N. Haala and M. Kada, "An update on automatic 3D building reconstruction," *J. Photogramm. Remote Sens.*, vol. 65, no. 6, pp. 570–580, 2010.

[6] C. Brenner, "Building reconstruction from images and laser scanning," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 6, no. 3/4, pp. 187–198, 2005.

[7] C. A. Vanegas, D. G. Aliaga, and B. Benes, "Automatic extraction of manhattan-world building masses from 3D laser range scans," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 10, pp. 1627–1637, Oct. 2012.

[8] P. Dorninger and N Pfeifer, "A comprehensive automated 3D approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds," *Sensors*, vol. 8, no. 11, pp. 7323–7343, 2008.

[9] P. Musialski, P. Wonka, D. G. Aliaga, M. Wimmer, L. van Gool, and W. Purgathofer, "A survey of urban reconstruction," *Comput. Graph. Forum*, vol. 32, no. 6, pp. 146–177, May 2013.

[10] G. Vosselman and S. Dijkman, "3D building model reconstruction from point clouds and ground plans," *Int. Arch. Photogrammetry Remote Sens. Spatial Inf. Sci.*, vol. 34, no. 3/W4, pp. 37–44, 2001.

[11] F. Lafarge, X. Descombes, J. Zerubia, and M. Pierrot-Deseilligny, "Structural approach for building reconstruction from a single DSM," *IEEE Tran. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 135–147, Jan. 2010.

[12] B. Ameri and D. Fritsch, "Automatic 3D building reconstruction using plane-roof structures," in *Proc. Amer. Soc. Photogramm. Remote Sens., Washington, DC, USA*, 2000, pp. 22–26.

[13] B. Xiong, S. O. Elberink, and G. Vosselman, "A graph edit dictionary for correcting errors in roof topology graphs reconstructed from point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 93, pp. 227–242, Jul. 2014.

[14] B. Xiong, M. Jancosek, S. O. Elberink, and G. Vosselman, "Flexible building primitives for 3D building modeling," *ISPRS J. Photogramm. Remote Sens.*, vol. 101, pp. 275–290, Mar. 2015.

[15] V. Verma, R. Kumar, and S. Hsu, "3D building detection and modeling from aerial LIDAR data," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 2213–2220, vol. 2, 2006.

[16] S. Oude Elberink, "Target graph matching for building reconstruction," *ISPRS - Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 38, pp. 49–54, 2009.

[17] S. Oude Elberink and G. Vosselman, "Building reconstruction by target based graph matching on incomplete laser data: Analysis and limitations," *Sensors*, vol. 9, no. 8, 2009, Art. no. 6101.

[18] H. Wang, W. Zhang, Y. Chen, M. Chen, and K. Yan, "Semantic decomposition and reconstruction of compound buildings with symmetric roofs from LiDAR data and aerial imagery," *Remote Sens.*, vol. 7, no. 10, pp. 13 945–13 974, Oct. 2015.

[19] S. Sun and C. Salvaggio, "Aerial 3D building detection and modeling from airborne LiDAR point clouds," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 3, pp. 1440–1449, Jun. 2013.

[20] C. Liu, B. Shi, X. Yang, N. Li, and H. Wu, "Automatic buildings extraction from LiDAR data in urban area by neural oscillator network of visual cortex," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 4, pp. 2008–2019, Aug. 2013.

[21] Z. Zhang *et al.*, "A multilevel point-cluster-based discriminative feature for ALS point cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3309–3321, Jun. 2016.

[22] Z. Zhang, L. Zhang, X. Tong, B. Guo, L. Zhang, and X. Xing, "Discriminative-dictionary-learning-based multilevel point-cluster features for ALS point-cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7309–7322, Dec. 2016.

[23] D. Chen, L. Zhang, J. Li, and R. Liu, "Urban building roof segmentation from airborne LiDAR point clouds," *Int. J. Remote Sens.*, vol. 33, no. 20, pp. 6497–6515, Oct. 2012.

[24] M. Awrangjeb, "Automatic segmentation of raw LiDAR data for extraction of building roofs," *Remote Sens.*, vol. 6, no. 5, pp. 3716–3751, 2014.

[25] M. Awrangjeb, "Using point cloud data to identify, trace, and regularize the outlines of buildings," *Int. J. Remote Sens.*, vol. 37, no. 3, pp. 551–579, 2016.

[26] K. Zhang, J. Yan, and S.-C. Chen, "Automatic construction of building footprints from airborne LIDAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 9, pp. 2523–2533, Sep. 2006.

[27] Q. Y. Zhou and U. Neumann, "Fast and extensible building modeling from airborne LiDAR data," in *Proc. 16th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2008, pp. 7–15.

[28] Q. Y. Zhou and U. Neumann, "A streaming framework for seamless building reconstruction from large-scale aerial LiDAR data," *IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2759–2766.

[29] C. Poullis and S. You, "Automatic reconstruction of cities from remote sensor data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2775–2782.

[30] C. Poullis, "A framework for automatic modeling from point cloud data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2563–2575, Nov. 2013.

[31] B. C. Matei, H. S. Sawhney, S. Samarasekera, J. Kim, and R. Kumar, "Building segmentation for densely built urban regions using aerial LIDAR data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[32] Y. Chen, L. Cheng, M. Li, J. Wang, L. Tong, and K. Yang, "Multiscale grid method for detection and reconstruction of building roofs from airborne lidar data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 10, pp. 4081–4094, Oct. 2014.

[33] D. Chen, L. Zhang, P. T. Mathiopoulos, and X. Huang, "A methodology for automated segmentation and reconstruction of urban 3-D buildings from ALS point clouds," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 10, pp. 4199–4217, Oct. 2014.

[34] D. Chen, R. Wang, and J. Peethambaran, "Topologically aware building rooftop reconstruction from airborne laser scanning point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7032–7052, 2017.

[35] G. Vosselman, "Building reconstruction using planar faces in very high density height data," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. 32, no. Part 3/ 2W5, pp. 87–92, 1999.

[36] H.-G. Maas and G. Vosselman, "Two algorithms for extracting building models from raw laser altimetry data," *ISPRS J. Photogramm. Remote Sens.*, vol. 54, no. 2/3, pp. 153–163, Jul. 1999.

[37] G. Sohn, X. Huang, and V. Tao, "Using a binary space partitioning tree for reconstructing polyhedral building models from airborne LiDAR data," *Photogramm. Eng. Remote Sens.*, vol. 74, no. 11, pp. 1425–1438, Nov. 2008.

[38] A. Sampath and J. Shan, "Segmentation and reconstruction of polyhedral building roofs from aerial LiDAR point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 3, pp. 1554–1567, Mar. 2010.

[39] K. Kim and J. Shan, "Building roof modeling from airborne laser scanning data based on level set approach," *ISPRS J. Photogramm. Remote Sens.*, vol. 66, no. 4, pp. 484–497, Jul. 2011.

[40] Q. Y. Zhou and U. Neumann, "2.5d building modeling by discovering global regularities," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 326–333.

[41] J. Yan, K. Zhang, C. Zhang, S.-C. Chen, and G. Narasimhan, "Automatic construction of 3-D building model from airborne LiDAR data through 2-D snake algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 3–14, Jan. 2015.

[42] W. J. Schroeder, J. A. Zarge, and W. E. Lorensen, "Decimation of triangle meshes," *ACM SIGGRAPH Comput. Graph.*, vol. 26, no. 2, pp. 65–70, Jul. 1992.

[43] J. Rossignac and P. Borrel, "Multi-resolution 3D approximations for rendering complex scenes," in *Modeling in Computer Graphics*. New York, NY, USA: Springer-Verlag, 1993, pp. 455–465.

[44] M. Algorri and F. Schmitt, "Mesh simplification," *Comput. Graph. Forum*, vol. 15, no. 3, pp. 77–86, 1996.

[45] N. Haala and M. Kada, "An update on automatic 3D building reconstruction," *ISPRS J. Photogramm. Remote Sens.*, vol. 65, no. 6, pp. 570–580, Nov. 2010.

[46] Q. Y. Zhou and U. Neumann, "2.5 D dual contouring: A robust approach to creating building models from aerial LiDAR point clouds," in *Proc. 11th Eur. Conf. Comput. Vis. Conf. Comput. Vis.: Part III*, 2010, pp. 115–128.

[47] H. Vo, "Streaming simplification of tetrahedral meshes," *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 1, pp. 145–155, Jan. 2007.

[48] T. Ju, F. Losasso, S. Schaefer, and J. Warren, "Dual contouring of hermite data," *ACM Trans. Graph.*, vol. 21, no. 3, Jul. 2002.

[49] R. Schnabel, R. Wahl, and R. Klein, "Efficient RANSAC for point-cloud shape detection," *Comput. Graph. Forum*, vol. 26, no. 2, pp. 214–226, Jun. 2007.

[50] D. Borrmann, J. Elseberg, K. Lingemann, and A. NÃijchter, "The 3D hough transform for plane detection in point clouds: A review and a new accumulator design," *3D Res.*, vol. 2, no. 2, pp. 1–13, Jun. 2011.

[51] F. Lafarge and C. Mallet, "Creating large-scale city models from 3d-point clouds: A robust approach with hybrid representation," *Int. J. Comput. Vis.*, vol. 99, no. 1, pp. 69–85, Feb. 2012.

[52] Q. Y. Zhou and U. Neumann, "Complete residential urban area reconstruction from dense aerial LiDAR point clouds," *Graph. Models*, vol. 75, no. 3, pp. 118–125, May 2013.

[53] Q. Y. Zhou and U. Neumann, "2.5 D building modeling with topology control," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun 2011, pp. 2489–2496.

[54] C. Poullis and S. You, "Photorealistic large-scale urban city model reconstruction," *IEEE Trans. Vis. Comput. Graph.*, vol. 15, no. 4, pp. 654–669, Jul. 2009.

[55] C. Poullis, S. You, and U. Neumann, "Rapid creation of large-scale photorealistic virtual environments," in *Proc. IEEE Virtual Reality Conf.*, 2008, pp. 153–160.

[56] H. Huang, C. Brenner, and M. Sester, "3D building roof reconstruction from point clouds via generative models," in *Proc. 19th ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst.*, 2011, pp. 16–24.

[57] H. Huang *et al.*, "A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data," *ISPRS J. Photogramm. Remote Sens.*, vol. 79, pp. 29–43, 2013.

[58] A. Henn *et al.*, "Model driven reconstruction of roofs from sparse LiDAR point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 76, pp. 17–29, 2013.

[59] N. Haala and C. Brenner, "Virtual city models from laser altimeter and 2D map data," *Photogramm. Eng. Remote Sens.*, vol. 65, no. 7, pp. 787–795, 1999.

[60] C. Brenner, "Interactive modelling tools for 3D building reconstruction," in, *Photogrammetric Week*, D. Fritsch and R. Spiller, Eds. Osnabrück, Germany: Wichmann, 1999.

[61] F. Lafarge *et al.*, "Automatic building extraction from DEMS using an object approach and application to the 3D-city modeling," *ISPRS J. Photogramm. Remote Sens.*, vol. 63, no. 3, pp. 365–381, 2008.

[62] M. Kada and L. McKinley, "3D building reconstruction from LiDAR based on a cell decomposition approach," in *Proc. Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 38, 2009, Paper W4.

[63] Y. Zheng and Q. Weng, "Model-driven reconstruction of 3-D buildings using LiDAR data," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 7, pp. 1541–1545, Jul. 2015.

[64] E. Kwak and A. Habib, "Automatic representation and reconstruction of DBM from LiDAR data using recursive minimum bounding rectangle," *ISPRS J. Photogramm. Remote Sens.*, vol. 93, pp. 171–191, Jul. 2014.

[65] J. Chen and B. Chen, "Architectural modeling from sparsely scanned range data," *Int. J. Comput. Vis.*, vol. 78, pp. 223–236, Jul. 2008.

[66] R. Wang, L. Xie, and D. Chen, "Modeling indoor spaces using decomposition and reconstruction of structural elements," *Photogramm. Eng. Remote Sens.*, vol. 83, no. 12, pp. 827–841, 2017.

[67] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[68] I. Stamos and P. K. Allen, "Geometry and texture recovery of scenes of large scale," *Comput. Vis. Image Understanding*, vol. 88, pp. 94–118, Nov. 2002.

[69] R. Schnabel, R. Wahl, and R. Klein, "RANSAC based out-of-core point-cloud shape detection for city-modeling," in *Proc. Terrestrisches Laserscan. (TLS 2007)*, Beiträge zum 74, DVW-Seminar, Fulda, Germany, 2007.

[70] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[71] J. F. Côté, J. L. Widlowski, R. A. Fournier, and M. M. Verstraete, "The structural and radiative consistency of three-dimensional tree reconstructions from terrestrial lidar," *Remote Sensing Environ.*, vol. 113, no. 5, pp. 1067–1081, 2009.

[72] Y. Livny *et al.*, "Texture-lobes for tree modelling," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 53:1–53:10, Jul. 2011.

[73] D. Yan, J. Wintz, B. Mourrain, W. Wang, F. Boudon, and C. Godin, "Efficient and robust reconstruction of botanical branching structure from laser scanned points," in *Proc. 11th Int. Conf. Comput.-Aided Des. Comput. Graph.*, Huangshan, China, Aug. 19–21, 2009, pp. 572–575.

[74] P. Raumonen *et al.*, "Fast automatic precision tree models from terrestrial laser scanner data," *Remote Sens.*, vol. 5, no. 2, 2013, Art. no. 491.

[75] H. Zhao and R. Shibasaki, "Reconstructing urban 3d model using vehicle-borne laser range scanners," in *Proc. 3rd Int. Conf. 3-D Digit. Imag. Modeling*, 2001, pp. 349–356.

[76] S. Xia and R. Wang, "A fast edge extraction method for mobile LiDAR point clouds," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 8, pp. 1288–1292, Aug. 2017.

[77] C. Früeh and A. Zakhor, "Constructing 3D city models by merging ground-based and airborne views," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2003, pp. 562–569.

[78] C. Früeh, S. Jain, and A. Zakhor, "Data processing algorithms for generating textured 3D building facade meshes from laser scans and camera images," *Int. J. Comput. Vis.*, vol. 61, pp. 159–184, Feb. 2005.

[79] N. L. Chang and A. Zakhor, "Constructing a multivalued representation for view synthesis," *Int. J. Comput. Vis.*, vol. 45, pp. 157–190, Nov. 2001.

[80] Q. Zheng *et al.*, "Non-local scan consolidation for 3D urban scenes," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 94:1–94:9, Jul. 2010.

[81] L. Nan, A. Sharf, H. Zhang, D. Cohen-Or, and B. Chen, "Smartboxes for interactive urban reconstruction," *J. ACM Trans. Graph.,* vol. 29, no. 4, 2010, Art. no. 93.

[82] M. Arikan, M. Schwärzler, S. Flöry, M. Wimmer, and S. Maierhofer, "O-snap: Optimization-based snapping for modeling architecture," *ACM Trans. Graph.*, vol. 32, no. 1, pp. 6:1–6:15, Feb. 2013.

[83] W. Sui, L. Wang, B. Fan, H. Xiao, H. Wu, and C. Pan, "Layer-wise floorplan extraction for automatic urban building reconstruction," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 3, pp. 1261–1277, Mar. 2016.

[84] A. Monszpart, N. Mellado, G. Brostow, and N. Mitra, "RAPter: Rebuilding man-made scenes with regular arrangements of planes," *J. ACM Trans. Graph.*, vol. 34, 2015, Art. no. 103.

[85] H. Lin *et al.*, "Semantic decomposition and reconstruction of residential scenes from LiDAR data," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 66:1–66:10, Jul. 2013.

[86] Z. Li *et al.*, "A hierarchical methodology for urban facade parsing from tls point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 123, pp. 75–93, 2017.

[87] J. Peethambaran and R. Wang, "Enhancing urban facades via LiDAR based sculpting," *Comput. Graphics Forum*, vol. 36, pp. 511–528, 2017.

[88] C. H. Shen, S. S. Huang, H. Fu, and S. M. Hu, "Adaptive partitioning of urban facades," *ACM Trans. Graph.*, vol. 30, no. 6, 2011, Art. no. 184.

[89] Y. Verdie, F. Lafarge, and P. Alliez, "LOD generation for urban scenes," *ACM Trans. Graph.*, vol. 34, no. 3, 2015, Art. no. 30.

[90] A. Lindenmayer, "Mathematical models for cellular interactions in development II. Simple and branching filaments with two-sided inputs," *J. Theoretical Biol.*, vol. 18, no. 3, pp. 300–315, 1968.

[91] P. Prusinkiewicz and A. Lindenmayer, *The Algorithmic Beauty of Plants*, New York, NY, USA: Springer-Verlag, 1990.

[92] P. Tan, G. Zeng, J. Wang, S. B. Kang, and L. Quan, "Image-based tree modeling," *ACM Trans. Graph.*, vol. 26, no. 3, Jul. 2007, Art. no. 87.

[93] X. Zhang, H. Li, M. Dai, W. Ma, and L. Quan, "Data-driven synthetic modeling of trees," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 9, pp. 1214–1226, Sep. 2014.

[94] Z. Wang *et al.*, "A multiscale and hierarchical feature extraction method for terrestrial laser scanning point cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2409–2425, May 2015.

[95] L. Zhong, L. Cheng, H. Xu, Y. Wu, Y. Chen, and M. Li, "Segmentation of individual trees from TLS and MLS data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 2, pp. 774–787, Feb. 2017.

[96] H. Xu, N. Gossett, and B. Chen, "Knowledge and heuristic-based modeling of laser-scanned trees," *ACM Trans. Graph.*, vol. 26, no. 4, Oct. 2007, Art. no. 19.

[97] Y. Livny, F. Yan, M. Olson, B. Chen, H. Zhang, and J. El-Sana, "Automatic reconstruction of tree skeletal structures from point clouds," *ACM Trans. Graph.*, vol. 29, no. 6, pp. 151:1–151:8, Dec. 2010.

[98] C. Zhu, X. Zhang, B. Hu, and M. Jaeger, *Reconstruction of Tree Crown Shape From Scanned Data*, Berlin, Germany: Springer-Verlag, 2008, pp. 745–756.

[99] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel, "On the shape of a set of points in the plane," *IEEE Trans. Inf. Theory*, vol. 29, no. 4, pp. 551–559, Jul. 1983.

[100] J. F. Côté, R. A. Fournier, and R. Egli, "An architectural model of trees to estimate forest structural attributes using terrestrial LiDAR," *Environ. Model. Softw.*, vol. 26, no. 6, pp. 761–777, 2011.

[101] M. Dai, H. Li, and X. Zhang, *Tree Modeling Through Range Image Segmentation and 3D Shape Analysis*. Berlin, Germany: Springer-Verlag, 2010, pp. 413–422.

[102] F. Aiteanu and R. Klein, "Hybrid tree reconstruction from inhomogeneous point clouds," *Visual Comput.*, vol. 30, no. 6, pp. 763–771, 2014.

[103] F. Lafarge and C. Mallet, "Building large urban environments from unstructured point data," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1068–1075.

[104] Q. Y. Zhou and U. Neumann, "Complete residential urban area reconstruction from dense aerial LiDAR point clouds," *Graph. Models*, vol. 75, no. 3, pp. 118–125, 2013.

[105] N. Pfeifer and B. Gorte, and W. D, "Automatic reconstruction of single trees from terrestrial laser scanner data," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. 35, 2004, pp. 114–119.

[106] Z. L. Cheng, X. P. Zhang, and B. Q. Chen, "Simple reconstruction of tree branches from a single range image," *J. Comput. Sci. Technol.*, vol. 22, no. 6, pp. 846–858, 2007.

[107] Z. Wang *et al.*, "A structure-aware global optimization method for reconstructing 3-D tree models from terrestrial laser scanning data," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 9, pp. 5653–5669, Sep. 2014.

[108] G. Reeb, "Sur les points singuliers d'une forme de Pfaff complètement intégrable ou d'une fonction numérique," *Comptes Rendus Acad. Sci.*, vol. 222, pp. 847–849, 1946.

[109] A. Bucksch and R. Lindenbergh, "A skeletonization method for point cloud processing," *J. Photogramm. Remote Sens.*, vol. 63, no. 1, pp. 115–127, 2008.

[110] A. Bucksch, R. Lindenbergh, and M. Menenti, "Skeltre," *Visual Comput.*, vol. 26, no. 10, pp. 1283–1300, 2010.

[111] A. Schilling and H.-G. Maas, "Automatic reconstruction of skeletal structures from TLS forest scenes," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. II-5, pp. 321–328, May 2014.

[112] S. Kaasalainen *et al.*, "Change detection of tree biomass with terrestrial laser scanning and quantitative structure modelling," *Remote Sens.*, vol. 6, no. 5, pp. 3906–3922, 2014.

[113] P. Raumonen, E. Casella, K. Calders, S. Murphy, M. Akerblom, and M. Kaasalainen, "Massive-scale tree modelling from TLS data," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. II-3/W4, pp. 189–196, 2015.

[114] D. Belton, S. Moncrieff, and J. Chapman, "Processing tree point clouds using Gaussian mixture models," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, no. 2, pp. 43–48, Oct. 2013.

[115] H. Li, X. Zhang, M. Jaeger, and T. Constant, "Segmentation of forest terrain laser scan data," in *Proc. 9th ACM SIGGRAPH Conf. Virtual-Reality Continuum Appl.Ind.*, New York, NY, USA, 2010, pp. 47–54.

[116] J. Hackenberg, H. Spiecker, K. Calders, M. Disney, and P. Raumonen, "Simpletree—An efficient open source tool to build tree models from TLS clouds," *Forests*, vol. 6, no. 11, pp. 4245–4294, 2015.

[117] S. Delagrange, C. Jauvin, and P. Rochon, "Pypetree: A tool for reconstructing tree perennial tissues from point clouds," *Sensors*, vol. 14, no. 3, pp. 4271–42789, 2014.

[118] A. Verroust and F. Lazarus, "Extracting skeletal curves from 3D scattered data," *Vis. Comput.*, vol. 16, no. 1, pp. 15–25, Feb. 2000.

[119] B. Guo, Q. Li, X. Huang, and C. Wang, "An improved method for power-line reconstruction from point cloud data," *Remote Sens.*, vol. 8, no. 1, 2016, Art. no. 36.

[120] S. J. Mills *et al.*, "Evaluation of aerial remote sensing techniques for vegetation management in power-line corridors," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 9, pp. 3379–3390, Sep. 2010.

[121] Y. Jwa and G. Sohn, "A piecewise catenary curve model growing for 3d power line reconstruction," *Photogramm. Eng. Remote Sens.*, vol. 78, no. 12, pp. 1227–1240, 2012.

[122] R. A. McLaughlin, "Extracting transmission lines from airborne LiDAR data," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 2, pp. 222–226, Apr. 2006.

[123] L. Cheng, L. Tong, Y. Wang, and M. Li, "Extraction of urban power lines from vehicle-borne LiDAR data," *Remote Sens.*, vol. 6, no. 4, 2014, Art. no. 3302.

[124] T. Melzer and C. Briese, "Extraction and modeling of power lines from als point clouds," in *Proc. 28 Workshop Austrian Assoc. Pattern Recognit.*, Hangenberg, Austria, 2004, pp. 47–54.

[125] Y. Jwa, G. Sohn, and H. B. Kim, "Automatic 3D powerline reconstruction using airborne LiDAR data," in *Proc. Int. Arch. Photogramm. Remote Sens.*, 2009, pp. 105–110.

[126] G. Sohn, Y. Jwa, and H. B. Kim, "Automatic powerline scene classification and reconstruction using airborne LiDAR data," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. I-3, pp. 167–172, Jul. 2012.

[127] J. Liang, J. Zhang, K. Deng, Z. Liu, and Q. Shi, "A new power-line extraction method based on airborne LiDAR point cloud data," in *Proc. Int. Symp. Image Data Fusion*, Tengchong, China, 2011, pp. 1–4.

[128] M. Ritter and W. Benger, "Reconstructing power cables from LiDAR data using eigenvector streamlines of the point distribution tensor field," *J. WSCG*, vol. 20, pp. 223–230, 2012.

[129] M. Lu and Z. Keiloch, "Accuracy of transmission line modeling based on aerial LiDAR survey," *IEEE Trans. Power Del.*, vol. 3, no. 3, pp. 1655–1663, Jul. 2008.

[130] H. Guan, J. Li, S. Cao, and Y. Yu, "Use of mobile LiDAR in road information inventory: A review," *Int. J. Image Data Fusion*, vol. 7, no. 3, pp. 219–242, 2016.

[131] L. J. Quackenbush, J. Im, and Y. Zuo, "Road extraction: A review of 1895 LiDAR-focused studies," in *Remote Sensing Natural Resources*, G. Wang and Q. Weng, Eds. Boca Raton, FL, USA: CRC Press, 2013, pp. 155–169.

[132] G. Chen, G. Esch, P. Wonka, P. Mueller, and E. Zhang, "Interactive procedural street modeling," *ACM Trans. Graph.*, vol. 27, no. 3, 2008, Art. no. 103.

[133] H. Nguyen, B. Desbenoit, and M. Daniel, "Realistic road path reconstruction from GIS data," *Comput. Graph. Forum*, vol. 33, no. 7, pp. 259–268, 2014.

[134] C. Zhang, "Towards an operational system for automated updating of road databases by integration of imagery and geodata," *J. Photogramm. Remote Sens.*, vol. 58, no. 3/4, pp. 166–186, 2004.

[135] S. J. Oude Elberink and G. Vosselman, "3D information extraction from laser point clouds covering complex road junctions," *Photogramm. Rec.*, vol. 24, no. 125, pp. 23–36, 2009.

[136] G. Vosselman, "3D reconstruction of roads and trees for city modelling," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci,* Dresden, Germany, vol. 34, 2003, Art. no. 3.

[137] S. J. Yu, S. R. Sukumar, A. F. Koschan, D. L. Page, and M. A. Abidi, "3D reconstruction of road surfaces using an integrated multi-sensory approach," *Opt. Lasers Eng.*, vol. 45, no. 7, pp. 808–818, 2007.

[138] A. Hervieu and B. Soheilian, "Semi-automatic road/pavement modeling using mobile laser scanning," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, no. 3, pp. 31–36, Oct. 2013.

[139] G. Sithole and G. Vosselman, "Bridge detection in airborne laser scanner data," *ISPRS J. Photogramm. Remote Sens.*, vol. 61, no. 1, pp. 33–46, 2006.

[140] L. Cheng, Y. Wu, Y. Wang, L. Zhong, Y. Chen, and M. Li, "Three-dimensional reconstruction of large multilayer interchange bridge using airborne LiDAR data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 2, pp. 691–708, Feb 2015.

[141] A. Jaakkola, J. Hyyppä, H. Hyyppä, and A. Kukko, "Retrieval algorithms for road surface modelling using laser-based mobile mapping," *Sensors*, vol. 8, no. 9, pp. 5238–5249, 2008.

[142] E. Denis, R. Burck, and C. Baillard, "Towards road modelling from terrestrial laser points," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 38, pp. 293–298, 2010.

[143] S. Oude Elberink and G. Vosselman, "3D modelling of topographic objects by fusing 2D maps and LiDAR data," in *Proc. ISPRS TC-IV Int. Symp. Geospatial Databases Sustain. Develop.*, Goa, India, 2006, pp. 17–30.

[144] L. C. Chen and C. Y. Lo, "3D road modeling via the integration of large-scale topomaps and airborne LiDAR data," *J. Chin. Inst. Eng.*, vol. 32, no. 6, pp. 811–823, 2009.

[145] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Proc. 4th Eurograph. Symp. Geom. Process.*, Aire-la-Ville, Switzerland, 2006, pp. 61–70.

[146] M. Berger, J. A. Levine, L. G. Nonato, G. Taubin, and C. T. Silva, "A benchmark for surface reconstruction," *ACM Trans. Graph.*, vol. 32, no. 2, pp. 20:1–20:17, Apr. 2013.

[147] J. Bloomenthal and B. Wyvill, Eds.,*Introduction to Implicit Surfaces*. San Francisco, CA, USA: Morgan Kaufmann, 1997.

[148] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 163–169, Aug. 1987.

[149] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," *SIGGRAPH Comput. Graph.*, vol. 26, no. 2, pp. 71–78, Jul. 1992.

[150] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Techn.*, New York, NY, USA, 1996, pp. 303–312.

[151] J. Manson, G. Petrova, and S. Schaefer, "Streaming surface reconstruction using wavelets," in *Proc. Symp. Geom. Process.*, Aire-la-Ville, Switzerland, 2008, pp. 1411–1420.

[152] J. C. Carr *et al.*, "Reconstruction and representation of 3D objects with radial basis functions," in *Proc. 28th Annu. Conf. Comput. Graph. Interactive Techn.*, New York, NY, USA, 2001, pp. 67–76.

[153] G. Turk and J. F. O'Brien, "Modelling with implicit surfaces that interpolate," *ACM Trans. Graph.*, vol. 21, no. 4, pp. 855–873, Oct. 2002.

[154] M. Bolitho, M. Kazhdan, R. Burns, and H. Hoppe, "Multilevel streaming for out-of-core surface reconstruction," in *Proc. 5th Eurograph. Symp. Geom. Process.*, Aire-la-Ville, Switzerland, 2007, pp. 69–78.

[155] M. Kazhdan, "Reconstruction of solid models from oriented point sets," in *Proc. 3rd Eurograph. Symp. Geom. Process.*, Aire-la-Ville, Switzerland, 2005, Paper 73.

[156] M. Kazhdan and H. Hoppe, "Screened poisson surface reconstruction," *ACM Trans. Graph.*, vol. 32, no. 3, pp. 29:1–29:13, Jul. 2013.

[157] D. Levin, *Mesh-Independent Surface Interpolation*. Berlin, Germany: Springer-Verlag, 2004, pp. 37–49.

[158] M. Alexa, J. Behr, D. Cohen-or, S. Fleishman, D. Levin, and C. T. Silva, "Computing and rendering point set surfaces," *IEEE Trans. Vis. Comput. Graph.*, vol. 9, no. 1, pp. 3–15, Jan.–Mar. 2003.

[159] R. Kolluri, "Provably good moving least squares," *ACM Trans. Algorithms*, vol. 4, no. 2, pp. 18:1–18:25, May 2008.

[160] G. Guennebaud and M. Gross, "Algebraic point set surfaces," *J. ACM Trans. Graph.*, vol. 26, no. 3, 2007, Art. no. 23.

[161] S. Fleishman, D. Cohen-Or, M. Alexa, and C. T. Silva, "Progressive point set surfaces," *ACM Trans. Graph.*, vol. 22, no. 4, pp. 997–1011, Oct. 2003.

[162] S. Fleishman, D. Cohen-Or, and C. T. Silva, "Robust moving least-squares fitting with sharp features," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 544–552, Jul. 2005.

[163] Y. Ohtake, A. Belyaev, M. Alexa, G. Turk, and H.-P. Seidel, "Multi-level partition of unity implicits," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 463–470, Jul. 2003.

[164] Y. Ohtake, A. Belyaev, and H.-P. Seidel, "Sparse surface reconstruction with adaptive partition of unity and radial basis functions," *Graph. Models*, vol. 68, no. 1, pp. 15–24, 2006.

[165] M. G. López, B. Mederos, and O. Dalmau, *GP-MPU Method for Implicit Surface Reconstruction*. Cham, Switzerland: Springer-Verlag, 2014, pp. 269–280.
[166] F. Cazals and J. Giesen, "Delaunay triangulation based surface reconstruction: Ideas and algorithms," in *Effective Computational Geometry for Curves and Surfaces.*, New York, NY, USA: Springer-Verlag, 2006, pp. 231–273.
[167] T. K. Dey, *Curve and Surface Reconstruction: Algorithms With Mathematical Analysis (Cambridge Monographs on Applied and Computational Mathematics)*. New York, NY, USA: Cambridge Univ. Press, 2006.
[168] J. D. Boissonnat, "Geometric structures for three-dimensional shape representation," *ACM Trans. Graph.*, vol. 3, no. 4, pp. 266–286, Oct. 1984.
[169] H. Edelsbrunner and E. P. Mücke, "Three-dimensional alpha shapes," *ACM Trans. Graph.*, vol. 13, no. 1, pp. 43–72, Jan. 1994.
[170] N. Amenta, M. Bern, and D. Eppstein, "The crust and the beta-skeleton: Combinatorial curve reconstruction," *Graph. Models Image Process.*, vol. 60, 1998, pp. 125–135.
[171] N. Amenta, M. Bern, and M. Kamvysselis, "A new voronoi-based surface reconstruction algorithm," in *Proc. 25th Annu. Conf. Comput. Graph. Interactive Techn.* New York, NY, USA, 1998, pp. 415–421.
[172] N. Amenta and M. Bern, "Surface reconstruction by voronoi filtering," in *Proc. 14th Annu. Symp. Comput. Geom.* New York, NY, USA, 1998, pp. 39–48.
[173] N. Amenta, S. Choi, and R. K. Kolluri, "The power crust," in *Proc. 6th ACM Symp. Solid Model. Appl.*, New York, NY, USA, 2001, pp. 249–266.
[174] B. Mederos, N. Amenta, L. Velho, and L. H. de Figueiredo, "Surface reconstruction from noisy point clouds," in *Proc. 3rd Eurographics Symp. Geom. Process.*, Aire-la-Ville, Switzerland, 2005, pp. 11–21.
[175] T. K. Dey and S. Goswami, "Tight cocone: A water-tight surface reconstructor," in *Proc. 8th ACM Symp. Solid Modeling Appl.*, New York, NY, USA, 2003, pp. 127–134.
[176] N. Amenta, S. Choi, T. K. Dey, and N. Leekha, "A simple algorithm for homeomorphic surface reconstruction," in *Proc. 16th Annu. Symp. Comput. Geom.*, New York, NY, USA, 2000, pp. 213–222.
[177] T. K. Dey, J. Giesen, and J. Hudson, "Delaunay based shape reconstruction from large data," in *Proc. IEEE Symp. Parallel Large-Data Vis. Graph.*, San Diego, CA, USA, Oct. 22–23, 2001, pp. 19–27.
[178] J. Boissonnat, J. Snoeyink, T. K. Dey, and S. Goswami, "Special issue on the 20th ACM symposium on computational geometry provable surface reconstruction from noisy samples," *Comput. Geom.*, vol. 35, no. 1, pp. 124–141, 2006.
[179] R. C. Veltkamp, *Closed Object Boundaries From Scattered Points*, (Lecture Notes in Computer Science), vol. 885. New York, NY, USA: Springer-Verlag, 1994.
[180] M. Attene and M. Spagnuolo, "Automatic surface reconstruction from point sets in space," *Comput. Graph. Forum*, vol. 19, pp. 457–465, 2000.
[181] A. Gezahegne, *Surface Reconstruction With Constrained Sculpting*. Davis, CA, USA: Univ. California, 2005.
[182] J. Peethambaran and R. Muthuganapathy, "Reconstruction of water-tight surfaces through Delaunay sculpting," *Comput.-Aided Des.*, vol. 58, pp. 62–72, 2015.
[183] M. Gopi, S. Krishnan, and C. Silva, "Surface reconstruction based on lower dimensional localized Delaunay triangulation," *Comput. Graph. Forum*, vol. 19, pp. 467–478, 2000.
[184] H. Edelsbrunner, "Surface reconstruction by wrapping finite sets in space," in *Discrete and Computational Geometry* (Algorithms and Combinatorics), B. Aronov, S. Basu, J. Pach, and M. Sharir, Eds., vol. 25. Berlin, Germany: Springer-Verlag, 2003, pp. 379–404.
[185] D. Cohen-Steiner and F. Da, "A greedy delaunay-based surface reconstruction algorithm," *Vis. Comput.*, vol. 20, no. 1, pp. 4–16, Apr. 2004.
[186] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin, "The ball-pivoting algorithm for surface reconstruction," *IEEE Trans. Vis. Comput. Graph.*, vol. 5, no. 4, pp. 349–359, Oct. 1999.
[187] S. Petitjean and E. Boyer, "Combinatorial curves and surfaces regular and non-regular point sets: Properties and reconstruction," *Comput. Geom.*, vol. 19, no. 2, pp. 101–126, 2001.
[188] R. Chaine, "A geometric convection approach of 3-D reconstruction," in *Proc. Eurographics/ACM SIGGRAPH Symp. Geom. Process.*, Aire-la-Ville, Switzerland, 2003, pp. 218–229.
[189] C. C. Kuo and H. T. Yau, "A delaunay-based region-growing approach to surface reconstruction from unorganized points," *Comput.-Aided Des.*, vol. 37, no. 8, pp. 825–835, 2005.
[190] P. Cignoni, C. Rocchini, and R. Scopigno, "Metro: Measuring error on simplified surfaces," *Comput. Graph. Forum*, vol. 17, no. 2, pp. 167–174, 1998.
[191] M. Blaha, C. Vogel, A. Richard, J. D. Wegner, T. Pock, and K. Schindler, "Large-scale semantic 3D reconstruction: An adaptive multi-resolution model for multi-class volumetric labeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit., , Las Vegas, NV, USA, June 27–30, 2016*, pp. 3176–3184.
[192] L. Nan, C. Jiang, B. Ghanem, and P. Wonka, "Template assembly for detailed urban reconstruction," *J. Comput. Graphics Forum*, vol. 34, no. 2, pp. 217–228, 2015.
[193] K. Arroyo Ohori, H. Ledoux, F. Biljecki, and J. Stoter, "Modeling a 3D city model and its levels of detail as a true 4D model," *ISPRS Int. J. Geo-Inf.*, vol. 4, no. 3, pp. 1055–1075, Sep. 2015.
[194] F. Lafarge, "Some new research directions to explore in urban reconstruction," in *Proc. Joint Urban Remote Sens. Event*, Mar. 2015, pp. 1–4.
[195] M. Shahzad and X. X. Zhu, "Robust reconstruction of building facades for large areas using spaceborne tomosar point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 752–769, Feb. 2015.
[196] D. Reale, G. Fornaro, A. Pauciullo, X. Zhu, and R. Bamler, "Tomographic imaging and monitoring of buildings with very high resolution SAR data," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 661–665, Jul. 2011.
[197] Z. Lu, P. Guerrero, N. J. Mitra, and A. Steed, "Open3d: Crowd-sourced distributed curation of city models," in *Proc. 21st Int. Conf. Web3D Technol.*, New York, NY, USA, 2016, pp. 87–94.

Ruisheng Wang received the B.Eng. degree in photogrammetry and remote sensing from the Wuhan University, Wuhan, China, the M.Sc.E degree in geomatics engineering from the University of New Brunswick, Fredericton, NB, Canada, and the Ph.D. degree in computer engineering from the McGill University, Montreal, QC, Canada.

He is currently an Associate Professor with the Department of Geomatics Engineering, University of Calgary, Calgary, AB, Canada. His research interests include geomatics and computer vision, especially point cloud processing.

Jiju Peethambaran received the Bachelor's degree in information technology from the University of Calicut, Malappuram, India, the Master's degree in computer science from the National Institute of Technology, Karnataka, Mangalore, India, and the Ph.D. degree in computational geometry from Indian Institute of Technology, Madras, Chennai, India.

He is currently a Postdoctoral Researcher with the Department of Geomatics Engineering, University of Calgary, Calgary, AB, Canada. His research interests include computational geometry, 3-D graphics, real-time geometry processing, and possible applications including LiDAR-based urban modeling.

Dong Chen received the Ph.D. degree in geographical information sciences from Beijing Normal University, Beijing, China, in 2013.

He is an Assistant Professor with Nanjing Forestry University, Nanjing, China. He currently works as a Postdoctoral fellow with the Department of Geomatics Engineering, University of Calgary, Calgary, AB, Canada. His research interests include image- and LiDAR-based segmentation and reconstruction, full-waveform LiDAR data processing, and related remote sensing applications in the field of forest ecosystems.