Lightweight Reconstruction of Urban Buildings: Data Structures, Algorithms, and Future Directions

Vivek Kamra, Prachi Kudeshia, Somaye ArabiNaree, Dong Chen, Yasushi Akiyama, and Jiju Peethambaran, *Member, IEEE*

***Abstract*—Commercial buildings as well as residential houses represent core structures of any modern day urban or semiurban areas. Consequently, 3-D models of urban buildings are of paramount importance to a majority of digital urban applications, such as city planning, 3-D mapping and navigation, video games and movies, and construction progress tracking, among others. However, current studies suggest that existing 3-D modeling approaches often involve high computational cost and large storage volumes for processing the geometric details of the buildings. Therefore, it is essential to generate concise digital representations of urban buildings from the 3-D measurements or images so that the acquired information can be efficiently utilized for various urban applications. Such concise representations, often referred to as "lightweight" models, strive to capture the details of the physical objects with less computational storage. Furthermore, lightweight models consume less bandwidth for online applications and facilitate accelerated visualizations. With many emerging digital urban infrastructure applications, lightweight reconstruction is poised to become a new area of research in the urban remote sensing community. We aim to provide a thorough review of data structures, representations, and state-of-the-art algorithms for lightweight 3-D urban reconstruction. We discuss the strengths and weaknesses of key lightweight urban reconstruction techniques, ultimately providing guidance on future research prospects to fulfill the pressing needs of urban applications.**

***Index Terms*—3-D urban models, data structures, deep learning, geometric abstractions, level of detail (LOD) modeling, lightweight reconstruction, point cloud, procedural encoding.**

I. Introduction

Rapid growth in the urbanization with changing size of cities has introduced uncertainty in the understanding of the city dynamics. A solution to this ever-increasing problem is 3-D city models which enable the smart city paradigm. These digital models support a wide variety of applications, such as 3-D mapping and navigation, urban planning and smart cities, augmented/virtual reality (AR/VR), emergency response training, movies and video games, and mining, among others. Typically, point clouds of outdoor or indoor scenes and objects are acquired through 3-D sensors, such as light detection and ranging (LiDAR), and converted to triangular meshes using specialized urban reconstruction algorithms before being used in different real-world applications. These meshes usually contain hundreds of millions of triangles, putting an enormous burden on rendering, data transfer, and storage of applications. The massiveness of the 3-D mesh volume can be understood such that for a simple point cloud of a Stanford bunny of around 2 MBs, a typical reconstructed mesh uses at least 20 MB of memory [1]. The ongoing studies related to geometric modeling show that millions of triangles are required to represent the details of the model. For instance, the city scene in Fig. 1 with 5061 distinct buildings' meshes comprise 671 million triangles that require 37 GB of storage space. This leads to the issues of large memory size (number of faces increases the requirements of the meshes in memory space) and computing capacity for a large-scale urban scene models.

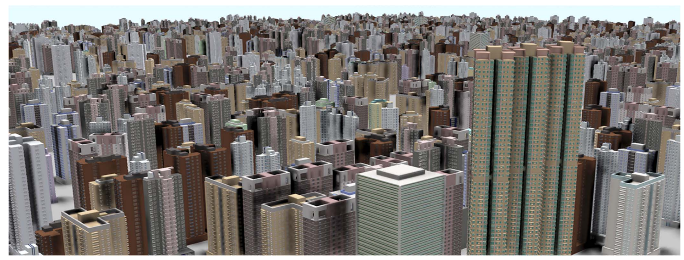


Fig. 1. Mesh-based model of a large-scale city scene with 5061 distinct buildings that consume a storage space of 37 GB [2].

Modern day web-based urban applications demand concise digital representations and, hence, favor lightweight reconstruction. Since this whole article is built around a comprehensive review of lightweight reconstruction, we attempt to provide a formal definition for the term lightweight reconstruction in the context of 3-D reconstruction.

Manuscript received 11 August 2022; revised 21 October 2022 and 2 December 2022; accepted 16 December 2022. Date of publication 28 December 2022; date of current version 5 January 2023. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada Discovery under Grant #2019-05092 and in part by the National Natural Science Foundation of China under Grant #42271450. *(Vivek Kamra and Prachi Kudeshia are co-first authors.) (Corresponding author: Jiju Peethambaran.)*

Vivek Kamra, Prachi Kudeshia, Yasushi Akiyama, and Jiju Peethambaran are with the Saint Mary's University, Halifax, NS TW1 4SX, Canada (e-mail: vivek.kamra@smu.ca; prachi.kudeshia@smu.ca; yasushi.akiyama@smu.ca; jiju.poovvancheri@smu.ca).

Somaye ArabiNaree is with the Saint Mary's University, Halifax, NS TW1 4SX, Canada, and also with the Kharazmi University, Tehran 15719-14911, Iran (e-mail: somaye.arabinaree@gmail.com).

Dong Chen is with the Nanjing Forestry University, Nanjing 210037, China (e-mail: chendong@njfu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3232758

Lightweight Reconstruction can be defined as the process of designing a reconstruction algorithm and/or a minimally complex and less heavy data structure, to digitally represent the physical object being modeled, that consumes minimum storage space, thus facilitating accelerated rendering and fast transmission of the model.

Lightweight models help in improving the overall efficiency of 3-D reconstruction systems, especially for the applications requiring real-time data transmission such as cloud-based 3-D printing services [1]. The aim of the lightweight reconstruction is to save memory without missing information and essential details. Therefore, the advantage of lightweight models over conventional heavy models is the preservation of the details while using less complicated structures in terms of memory usage. In some cases, the high quality of a reconstructed building model does not necessarily mean high geometrical accuracy. In contrast, it means lightweight building models are aligned with the specific needs of urban applications rather than serving visualization purposes only. For many graphics applications, high-quality textures for geometrical rendering are prioritized over the geometry detailing [3]. Hence, we need to sacrifice the geometric accuracy of the building models to make the reconstructed model lightweight and to be used for a wide range of purposes.

A. Scope of the Survey

Over the past years, numerous papers [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16] have been dedicated to review the existing developments in the area of 3-D city modeling. Wang et al. [4] briefly discuss fundamental approaches and principles of 3-D modeling for real as well as virtual cities. Berger et al. [9] categorize surface reconstruction techniques from point clouds for various applications, including urban environment reconstruction. This categorization comprises parameters including point cloud artifacts (noise, outliers, missing data), input requirements (oriented/unoriented normals), shape class (CAD, indoor scene objects, urban environment, architectural), and reconstruction output (mesh, point set, implicit field, volumetric segmentation). Xu et al. [10] explain various techniques for building and civil infrastructure reconstruction from point clouds, including 3-D point cloud acquisition and reconstruction methods. Similarly, a recent paper by Wang et al. [5] explores state-of-the-art deep learning-based 3-D urban modeling solutions that acquire data through mobile laser scanning along with mobile mapping system related applications. Xia et al. [11] discuss geometric primitives-based extraction methods from point clouds for urban reconstruction. These primitives are categorized into two classes, i.e., shape primitives (lines, surfaces, and volumetric shapes) and structure primitives (skeletons and edges). Musialski et al. [7] provide a comprehensive overview of urban reconstruction techniques based on input data including point clouds as well as images. In this survey, various methods are grouped under fundamental categories, including point clouds and cameras (multiview stereo (MVS) structure from motion), buildings and semantics [image-based modeling, LiDAR-based modeling, inverse procedural modeling (IPM)], façades and images (façade imagery, façade decomposition, and modeling), and blocks and cities (ground reconstruction, aerial reconstruction). Similarly, Wang et al. [12] discuss various LiDAR-based urban modeling techniques based on reconstructed objects, such as buildings, trees, power lines, roads/bridges, and sculptures. Further, Ying et al. [6] present the current stature and substantial future prospective of 3-D urban modeling methods used in between 2015 and 2020 based on their characteristics, data requirements, user and technology, and ethical considerations. However, despite a strong requirement for lightweight 3-D reconstruction, a survey consolidating various techniques and representations for lightweight reconstruction does not exist in the literature. Therefore, this article is timely aligned to fill this gap.

B. Review Topics and Contributions

In general, polygonal meshes are used to store accurate geometric details and hence constitute dense representations of models. However, these dense representations pose major challenges for storage, transmission speed, and object rendering, especially for web-based interactive applications. Thus, there is a requirement for the best representation of lightweight modeling approaches in this domain. As the urban model data structures and the modeling methodologies are closely related to the reconstruction of the lightweight models, we first review the state-of-the-art data representation methods of 3-D urban scenes comprising polygonal (triangular/quadrilateral/hexagonal) meshes, constructive solid geometry (CSG), B-Rep (Bezier patches, splines), and stellar decomposition. Additionally, we will review lightweight modeling approaches and their limitations in the context of urban scene. These algorithms include geometric abstractions, level of detail (LOD) modeling and mesh decimation through Gradient Tensors. Our specific contributions in this article are listed as follows.

1) Review and assessment of potential data structures and representations for lightweight models.
2) Consolidation and review of different algorithms that directly or indirectly focus on lightweight reconstruction.
3) Comprehensive list of future directions in the context of lightweight 3-D urban reconstruction.

The rest of this article is organized as follows. Section II presents a review of potential data structures for lightweight reconstruction of 3-D urban scenes. Algorithms for lightweight reconstruction are discussed in Section III. Section IV describes some of the future directions in the context of lightweight 3-D reconstruction of urban scenes. Finally, the conclusion and insights for future work are provided in Section V.

II. Data Structures and Other Representations for Lightweight Models

In graphics and remote sensing applications, three-dimensional objects can be efficiently represented using different geometric structures such as triangular meshes or CSG in CAD systems. Geometric data structures specify the fundamental elements of a 3-D object, such as, surfaces, space, and scene structure. This section describes the most common lightweight

TABLE I
TYPICAL STORAGE REQUIREMENTS OF DIFFERENT MESH DATA STRUCTURES

Mesh Structure	Vertex Position Array	Indices Array	Repr. Edge	Total Storage
Indexed Mesh Storage	12 bytes/vert	24 bytes/vert	-	36 bytes/vert
Winged-Edge Structure	12 bytes/vert	96 bytes/vert	4 bytes/vert	112 bytes/vert
Half-Edge Structure	12 bytes/vert	96 bytes/vert	4 bytes/vert	112 bytes/vert

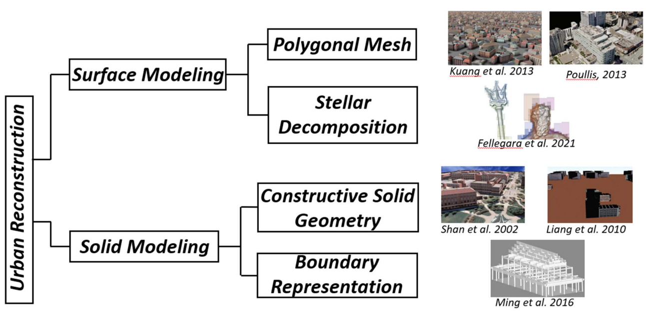


Fig. 2. Potential urban modeling data structures and representations.

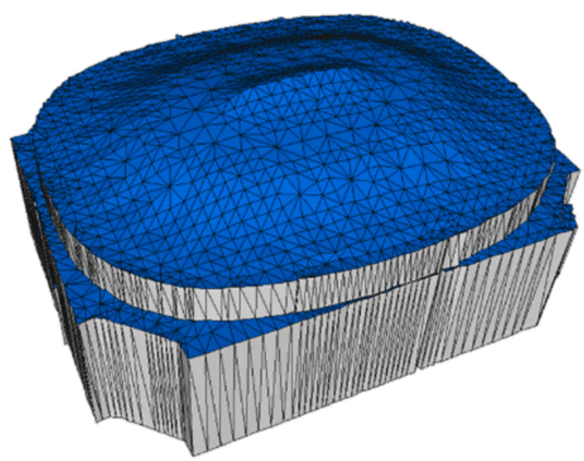


Fig. 3. Mesh model of a curved building from [17].

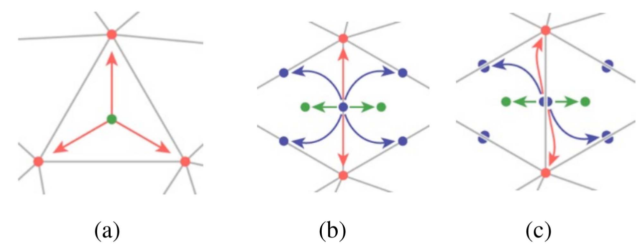


Fig. 4. Depiction of different mesh data structures using vertices (orange), faces (green), and edges (blue). (a) Indexed mesh structure. (b) Winged-edge structure where each vertex and face points to an edge. (c) Half-edge structure where each vertex and face points to a half-edge [20].

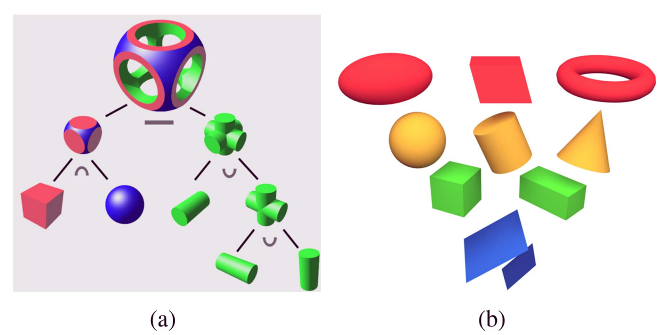


Fig. 5. CSG. (a) Example of CSG operations with expression [23]. (b) Four categories of geometric primitives showing level of complexity in different colors [24].

data structures used to represent 3-D objects, including polygonal (triangular/quadrilateral/hexagonal) meshes, CSG, B-Rep, and stellar decomposition.

A. Polygonal Meshes

Polygonal meshes are useful representations with many applications in computer graphics, geometric modeling, mechanical engineering, architecture, etc. As shown in Fig. 3, polygonal mesh consists of information about the geometry and topology of the 3-D shape in terms of vertices, edges, and faces that helps in defining the shape of a polyhedral object. Although triangles and quadrilaterals are the most common polygons used for surface representation of real-world models, several conceptual architectural structures benefit from free-form meshes with planar hexagonal faces. The most common data structures to represent the meshes are indexed lists, winged edge [18], and half-edge [19] data structures (see Fig. 4). These data structures allow the efficient traversal of meshes. Table I lists the space requirements for these structures.

B. Constructive Solid Geometry

CSG is a modeling technique used in CAD systems to create the complex geometry by combining primitives. As shown in Fig. 5(a), the CSG representation (or commonly referred as CSG tree) is an ordered binary tree where terminal nodes represent primitives and each internal node defines a Boolean set operation (union, intersection, and difference) applied to left and right nodes or corresponding child nodes [12]. Transformations such as scaling, rotation, and translation can be applied at any node of the CSG tree. Some graph-based modeling methods [21], [22] consider relationships between edges, primitives, or both for rooftop modeling. Complex rooftop can be modeled through roof topology graph, where rooftop modeling can be seen as rooftop graph matching, with fundamental topology graph elements in a model library.

Geometric primitive is an integral part of the CSG trees and is being used in many existing works [25], [26] to reconstruct 3-D model of building roofs and façades from airborne laser scanning (ALS) point clouds. It is defined as the simplest 3-D geometric shape which is used to construct complex shapes by assembling with others. Global intrinsic parameters of a

Fig. 6. Example of CSG model constructed from urban point cloud [27].

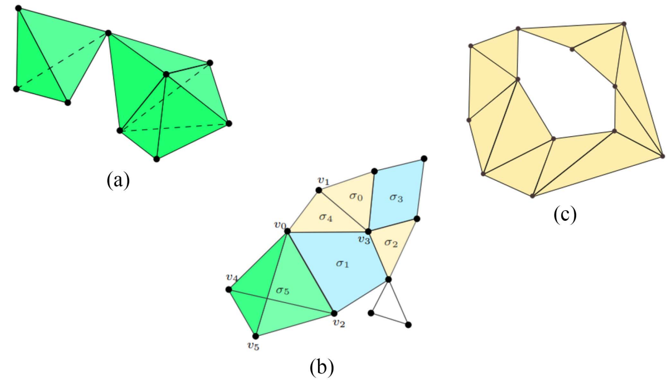


Fig. 7. Examples of CP complexes. (a) Pure simplicial three-complex having all top cells as tetrahedra. (b) CP complex containing top edges, triangles (in yellow), quads (in blue), and a tetrahedron (in green). (c) Pseudomanifold having triangles [30].

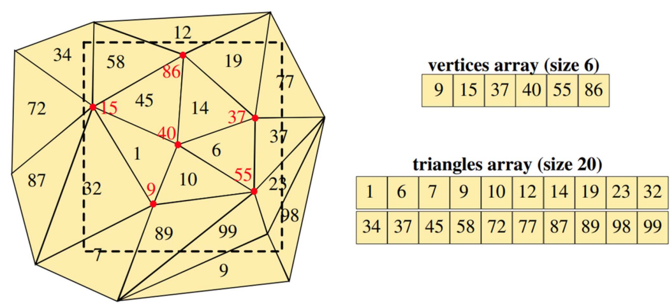


Fig. 8. Example of stellar decomposition encoding for a region with 6 vertices and 20 triangles. [30].

primitive are fixed and they are used to define the global size, orientation, and position of the shape. In addition, primitives are symmetric and all the primitives, except torus, are convex. These primitives, as shown in Fig. 5(b), can be classified into four categories [24], including planes, cube and cuboid, sphere, cylinder, and cone, and other shapes such as ellipsoids, torus, and nonrectangular parallelepipeds. Thus, CSG modeling often comes in handy to divide complex modeling tasks into different subtasks. Fig. 6 shows a CSG construction for 3-D point cloud of urban buildings.

C. Boundary Representation (B-Rep)

Boundary representation is a method for representing a shape using its boundaries. With this data structure, a solid structure can be depicted as a collection of connected surface elements defining the demarcation between interior and exterior points [28] in terms of surfaces, curves, and points based on its topology [29]. There are several types of surface representation including Bézier and B-spline surfaces.

A Bézier surface is a parametric patch used to model smooth surfaces that can be scaled indefinitely and can be defined by a set of control points. These surfaces have better continuity, less points to represent curved surfaces, and easy manipulation with the help of control points. However, the number of control points in Bézier representation is directly related to the degree of the curve. Furthermore, a change in any control point affects the entire curve that complicates the designing process in case of region-specific modification. On the other hand, in case of B-spline curves, the control points impart local control over the curve unlike the global control in the Bézier curve. Therefore, for region-specific modification, only the corresponding segment of the curve gets altered.

B-Rep is suitable for constructing solid models of unusual shapes. It is relatively easy to be converted into a wireframe model due to face, edge, and vertex information. On the other hand, wireframe models are less suitable for 3-D visualization due to absence of face information. For solid modeling, both B-Rep and CSG are used. However, B-Rep describes the oriented surface of a solid as a data structure in terms of vertices, edges, and faces, whereas CSG uses a set of Boolean expression of primitive solid objects to define a simpler structure. Furthermore, along with the Boolean operations, B-Rep has additional operators, such as extrusion (or sweeping), chamfer, blending, drafting, shelling, and tweaking. However, B-Rep requires large storage, especially if curved object is approximated with polyhedral models.

D. Stellar Decomposition

Mesh data structures are usually efficient for simple surfaces which can be defined by small low dimensional meshes. However, these mesh structures do not perform well in case of larger and higher dimensional meshes. In such cases, flexibility is required to deal with complex meshes, including irregularly connected cell types with the help of exploiting locality within the mesh.

The stellar decomposition [30] is a modern data structure that supports efficient navigation of the topological connectivity of simplicial and canonical polytope (CP) complexes. CP complexes (as shown in Fig. 7) are a class of cell complexes based on quadrilaterals, polygons, and pyramids. The stellar decomposition is both scalable and flexible to support the generation of optimal local data structures at runtime. Fig. 8 refers to the stellar decomposition encoding of arrays of vertices and top CP cells that explicitly list the associated elements, vertices, and triangles for each region r.

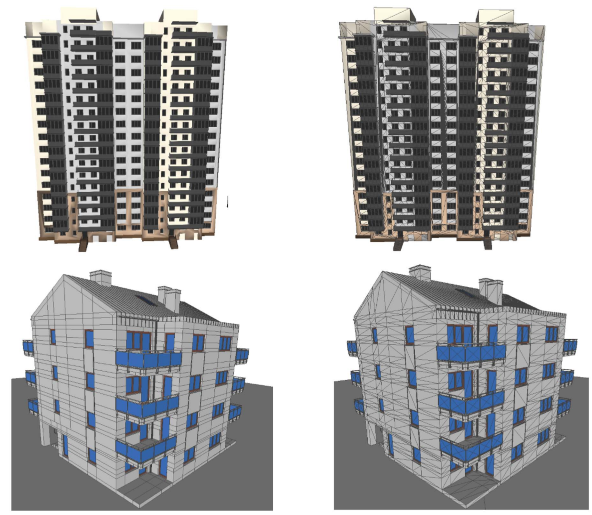
E. Assessment Mesh Versus Stellar Trees

A summary of different data structures mentioned in previous sections is shown in Fig. 2. In this section, we critically analyze the data structures based on the parameters, such as model accuracy and computational cost of 3-D reconstruction.

A mesh model constructed from point clouds using a surface reconstruction algorithm [31], [32] offers a decent representation. However, dense meshes generated from airborne images have shown greater geometric accuracy and completeness [33], whereas LiDAR scans provide less accurate representations due to sparsity, outliers, noise, and occlusions. Some recent advances in shape scanning and modeling [34], [35] utilize depth information encoded into mesh with 3-D textures, which results in the reconstruction of more realistic and geometrically accurate model. However, a European-style city with 40 400 distinct buildings [2] comprises 1.36 billion triangles with a storage consumption of 61 GB that makes the use of conventional meshes less suitable for web-based interactive applications. Thus, mesh simplification [17], [36] and polygonization [37] make a prominent candidate of meshes for lightweight urban reconstruction. However, there are significant challenges with mesh simplification technique that we will address in Section III.

On the other hand, Stellar tree based topological data structures have been used in mesh simplification [38] for local curvature estimation and mesh validation. Simplicial complexes that are not limited to triangle or tetrahedral meshes are complexes that are defined as collections of p-dimensional hypertetrahedra, and are known as p-simplices. We refer to the experimental analysis and comparison from [30], regarding storage comparison among Stellar tree encodings, such as EXPLICIT and COMPRESSED (most compact encoding). By adjusting the tuning parameter k_v, the EXPLICIT and VERTEX-COMPRESSED trees yield reduction in memory requirements with up to 20%–50%. For instance, NEPTUNE triangular dataset with storage requirement reduced from 32.0 to 26.2 MB for EXPLICIT trees, while COMPRESSED trees reduced from 5.76 to 1.24 MB.

While Stellar tree based data structures and simplicial complexes are widely used to discretize 3-D shapes, there is not enough study on them regarding urban representations. From our understanding, processing an urban model through stellar trees might not be the best choice as storage reductions offered in different encodings are limited. However, the efficient nature of stellar trees in processing the specific regions of a mesh by encoding topological relations might come in handy for lightweight urban reconstruction. Due to different levels of details required in lightweight reconstruction, such as geometric primitives based simple compact representation that preserves geometric structure, sharp features, and façade elements, we believe stellar trees can be used as local correspondents to define topological relations around feature areas of a mesh. Thus, we consider stellar trees as a useful representation for lightweight urban reconstruction.



(a) (b)

Fig. 9. (a) CAD models of sample buildings (downloaded from open-source 3-D modeling platform Sketchfab—https://sketchfab.com/features/free-3d-models). (b) Equivalent triangular mesh models.

F. Assessment-CAD Versus Mesh

CAD system based solid modeling techniques, such as CSG, B-Rep, and geometric primitives, have a vast background in modeling complex surfaces, objects, and buildings [26], [39], [40]. An earlier CSG-based reconstruction method [39] allows the generation of a model of the complicated buildings with sufficient details. However, the roof and wall textures in the images are mapped to the sample model to make the building models realistic, which increases the overall time consumption of reconstruction process. Another CSG B-Rep topological model mentioned in [40] assigns CSG-location, CSG-orientation, and CSG-subshape to a CSG-shape object such that Boolean operations can be performed on CSG B-Rep model. Note that this CSG B-Rep model offers a significantly lighter memory footprint of 336 KB against an equivalent triangular mesh model (3991 points and 7475 faces) with a memory footprint of 1986 KB. However, CSG B-Rep might not be suitable to represent real-world scenes having complex façade details, such as in European-style buildings. Considering the higher time consumption, we believe a CAD system based solid modeling approach might not be the best choice.

To better assess the representation of a CAD model in comparison to a mesh model, we take 3-D CAD models and their equivalent triangular mesh models (see Fig. 9). These CAD models have memory footprints of 14.4 and 2.94 MB in comparison to their equivalent triangular mesh models of 14.8 MB (116 166 vertices and 162 058 faces) and 3.46 MB (24 287 vertices and 35 423 faces) size. We observe that CAD-based representations are useful when model exhibits dominant planar regions. However, in case of complex architectural details, a triangular mesh model would be more suitable to better represent the LODs.

Fig. 10 represents buildings and their simplified version as triangular meshes and boundary representation. We observe that

TABLE II
STATISTICS RELATED TO DIFFERENT 3-D REPRESENTATIONS

Data Structure	Topology	Geometry	Application
Polygonal Mesh	vertices, edges and faces	surfaces	Continuous surfaces
Constructive Solid Geometry	geometric primitives	surfaces, curves and points	CAD models
Boundary Representation	vertices, edges and faces	surfaces, curves and points	CAD models
Stellar Decomposition	vertices, edges and faces	surfaces	Continuous surfaces

TABLE III
COMPARISON OF VARIOUS DATA STRUCTURES

3D Models	Data Structure	Models	# Vertices	# Faces	Storage (in MB)
Empire State	Polygonal Mesh (a)	Original	432,514	861,642	63.9
		Simplified	72,695	141,903	19.2
	Boundary Representation (b)	Original	2477	1675	27.2
		Simplified	1900	1263	21.5
Lans	Polygonal Mesh (c)	Original	25,994	51,426	13.48
		Simplified	18,543	31,598	11.48
	Boundary Representation (d)	Original	3534	2367	36.4
		Simplified	3349	2254	35.1

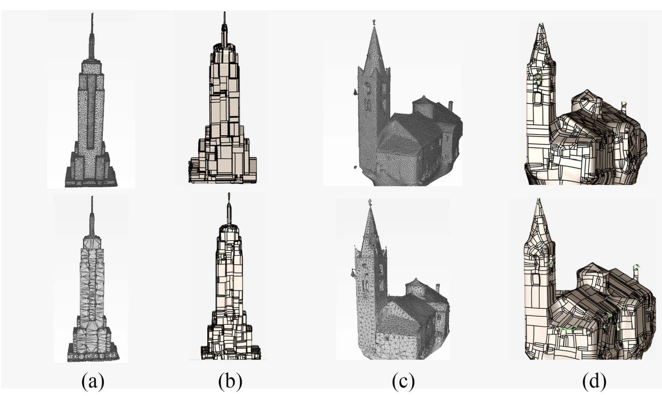


Fig. 10. 3-D models and their simplified version. (a) Empire state (triangular mesh). (b) Empire state (B-Rep). (c) Lans (triangular mesh). (d) Lans (B-Rep).

structural geometry can be preserved in B-Rep. However, this representation is not satisfactory to represent complex details, such as façade elements in comparison to triangular meshes. Table II lists the statistics such as number of vertices, number of faces, and total storage required for both original as well as simplified 3-D models. From the given data, it can be concluded that B-Rep offers lightweight modeling; however, such simpler representation is not enough to obtain complex detail. On the other hand, polygonal meshes consume more storage yet allows representations of precise information due to more number of faces.

G. Summary

To summarize, Table III presents a brief comparison on different potential data structures for lightweight reconstruction of urban scenes. Polygonal mesh and stellar decomposition can be classified into surface reconstruction where geometric element is surface, while CSG and B-Rep can be classified into solid modeling with geometric elements as surfaces, curves, and points. From topological perspective, polygonal mesh, B-Rep, and stellar decomposition rely on vertices, edges, and faces. However, CSG-based representations do not store such topological relation and, thus, only use collection of geometric primitives to represent shapes. This implies that CAD-based representations such as B-Rep and CSG are useful when model exhibits dominant planar regions. However, in case of complex architectural details, a triangular mesh model would be more suitable to better represent the LOD.

III. ALGORITHMS FOR LIGHTWEIGHT RECONSTRUCTION

In this section, we review various state-of-the-art lightweight reconstruction algorithms of geometric abstractions, procedural encoding, LOD modeling, deep learning based algorithms, and mesh decimation. A large volume of methods for urban scene reconstruction has been proposed; however, in this section, we mainly review the techniques relevant to lightweight reconstruction.

A. Geometric Abstraction

In many 3-D urban modeling works, 3-D laser scanning methods, such as LiDAR, are considered as the primary source of information. The data obtained through these methods include high geometric detail with an overwhelming amount of data which poses great challenges to visualization, therefore, encourage the need of lightweight 3-D models for interactive use. Due to the increasing complexity of geometric information, meaningful and concise abstractions become helpful for higher level interaction and modeling. More specifically, geometric abstraction allows for the reconstruction of 3-D models by using various primitives, which makes the resulting model lightweight and low-polygonal. Man-made environments, for instance, building-rich urban scenes are often dominated by planar surfaces such as walls. In such case, primitives-based

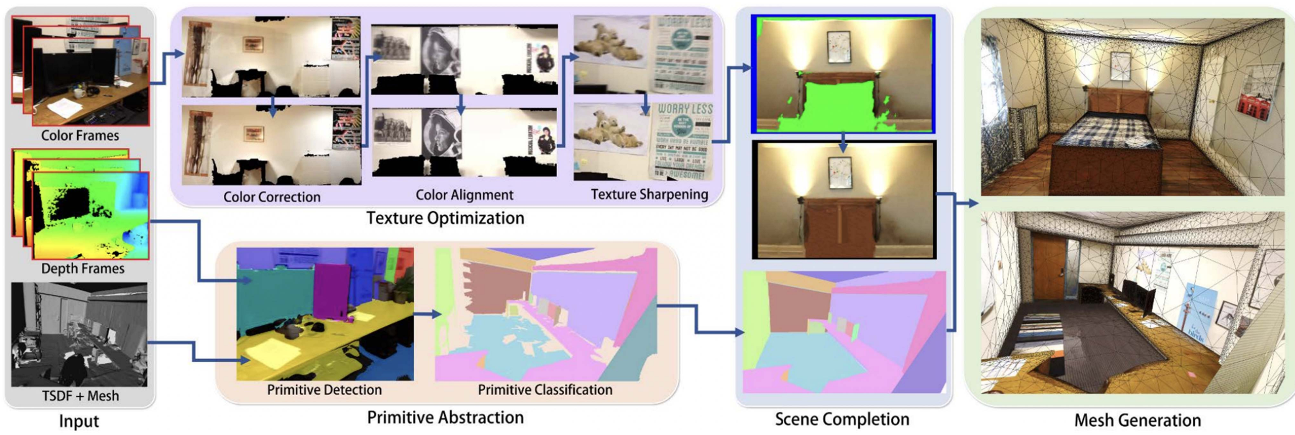


Fig. 11. Overall framework of 3DLite [3] to reconstruct low-polygonal meshes from RGB-D frames through primitive abstraction.

3-D geometric abstraction offers a decent representation with reduced memory footprint. RANSAC [41] is a popular preprocessing technique for geometric abstraction. It extracts shapes from point cloud data by randomly drawing minimal sets (minimum points that define a geometric primitive) and constructs respective primitive shapes [42]. Region growing is another popular technique for point cloud segmentation. It uses the normal of points in accordance with user-specified key parameters to categorize points that belong to the smooth surfaces. Hough transform [60] refers to an accumulator array where each cell of the array represents a set of parameters from discretized parameter space. This technique increases counter of cells for each point that belongs to a certain parameter space.

The 3DLite framework [3] (see Fig. 11) is proposed to reconstruct indoor 3-D environments using RGB-D sensors. This method computes a lightweight, low-polygonal primitive abstraction of the scanned RGB-D video frames. For each frame, 3DLite computes the initial camera poses and a truncated signed distance field representation of the scene to extract the initial mesh. Primitives are detected using the generated meshes and depth frames and the classified primitives are later used for scene completion. This method produces high-resolution, sharp surface textures through texture optimization technique that maps the scene geometry with sharp colors from the RGB data. It presents visually compelling 3-D reconstructed results. 3DLite performs geometry completion and texture completion tasks with a significant reduction in data size. It addresses the issues of oversmoothing and low color quality and provides a production ready solution for 3-D content creation community. However, due to the plane-based abstraction approach, sometimes this model does not perform efficiently with the nonplanar objects such as chairs.

In another lightweight modeling work [45], the authors described an algorithm that generates models and outperforms existing approximation techniques by preserving the sharpness of the raw data. This method proposes the modeling of buildings based on crosssectional contours using extrusion and tapering operations and further uses 2-D image processing techniques to perform 3-D reconstruction of urban buildings directly from point cloud data.

Geometric primitives along with model contours substantially contribute to lightweight reconstruction techniques. In [46], the authors proposed a framework for curved building reconstruction by assembling and deforming geometric primitives. First, the input LiDAR-based point cloud gets converted into contours that comprises the identification of individual buildings. Based on the identification of geometric primitives from the building contours, the initial models are obtained for reconstructing the building curves. Further, a warping field is used to refine the obtained models with respect to the several highly curved buildings.

Another primitive-based method by Xie et al. [59] have adopted primitive-based modeling and presented a method for the efficient reconstruction of building models from photogrammetric point clouds by combining rule-based and hypothesis-based methods. Initially, the planar primitives and respective boundaries extracted from the point cloud are regularized to obtain abstracted building contours. Then, a two-stage reconstruction method is applied to generate 3-D building models. In the first stage, to recover the topological relationship between different primitives, the regularity and adjacency of the building contours are used to construct an initial reconstruction model. In the next stage, an integer linear optimization problem is solved to remove and reconstruct topologies related to ambiguous areas. This method shows significant reduction in the number of faces in the reconstructed model against the state-of-the-art methods, hence it provides a concise 3-D representation to the urban scenes.

Zhang et al. [44] presented an automatic reconstruction of 3-D building façade models from photogrammetric mesh models. First, the mesh model is divided into components based on contour line. Local contour trees are exploited to find the segmented contour graphs by analyzing the topological relationship between the contours. Then, through an iterative process, whole model is segmented into diverse components from bottom to top. Next, the mesh model components are approximated via minimum circumscribed cuboids in an iterative manner. Finally, to ensure the accuracy of the reconstructed façade model, the parameters of the cuboid model are adjusted by means of a least square process.

Fig. 12. Example of building façade from showing the regularity and symmetry in the urban facade structure that is utilized to create procedural grammar [47].

B. Procedural Encoding of Urban Objects

Procedural modeling refers to the creation of 3-D models from sets of rules. These rules are generated using the regularity and similarity in the urban structures. Fig. 12 shows an example of an urban building which exhibits a number of windows positioned in a regular manner. The most common procedural modeling techniques are the Lindenmayer system (L-system) and shape grammar which apply sets of rules for producing objects as volumetric shapes. The resulting model comprises a combination of volumetric shapes as a lightweight representation. L-system is a parallel rewriting system that was introduced by Lindenmayer [61] and was designed to generate plants by computers using symbolic expressions. The two major components of L-systems are the rewriting system and the representing system. The rewriting system receives one axiom along with one or more production rules as input, and it expands the input iteratively. The representing system acts like turtle graphics to draw vector graphics by following the expanded commands from the rewriting system. The original system works in two-dimensional space; however, Lindenmayer expanded the system to three dimensions to generate more realistic plants. Since then many works are based on improvised versions of L-systems driven by their successful results.

An earlier work [47] proposed a procedural approach based on L-systems to model cities. This model takes various image maps as input such as land–water boundaries and population density to generate a system of highway and streets that divide the land into lots and creates the appropriate geometry for the buildings on the respective allotments. Further, by applying another L-system, the buildings are generated as a string representation of Boolean operations on simple solid shapes. Finally, a parser interprets all the results for the visualization software which processes polygonal geometry and texture maps. The use of shape grammar is also emphasized in this work which defines rules directly on shapes. Here, shape grammars are used to generate 2-D patterns and interactive design applications based on 3-D designs generated with the help of grammars. However, in this system, each style texture must be defined manually by visually determining the regularities and measuring façade element sizes. Once a shader is defined, the texture can scale to any width or height.

In the context of grammar-based designing, L-systems have shown impressive results in plant modeling [62], [63], [64]. However, buildings have stricter spatial constraints and their structure usually does not reflect a growth process. Therefore, L-systems cannot be easily adapted to the modeling of buildings.

Procedural modeling has been followed for several years as an efficient strategy to generate three-dimensional models of buildings. However, the key challenge of procedural technique is achieving the desired user intent. Therefore, it is difficult to obtain a procedural model that would generate a specific geometry. A significant change of parameters can create uncalculated changes during the repeated applications of the procedural rules and can quickly bring large modifications to the generated geometry. These changes might be difficult to understand and can cause situations out of control. Therefore, since more than 20 years, the emphasis on another way to obtain a procedural model is being given in many papers which is commonly known as reverse engineering on an existing geometry or IPM.

A recent paper on IPM [65] learns L-system representations of pixel images with branching structures. This method comprises a deep learning model to discover atomic structures such as line segments or branches. The orientation and scaling of these structures are determined and the detected structures are formed into a tree. The repeating parts are encoded into a small grammar by using greedy optimization where the output is an L-system that represents the input image as a simple text and a set of terminal symbols. This fully automatic model generates a compact set of textual rewriting rules to describe the input. However, authors have not provided any comparative study to show the level of compactness of this method.

C. LOD Modeling

LOD modeling is the process of generating a lightweight yet detailed representations of 3-D object models by preserving geometric features such as boundaries. This reduces the computation cost on the computer allowing algorithms to produce lightweight models of urban scenes. For city-scale reconstruction, aerial images captured at high altitudes are often used. Although the images are of high resolution and quality, the output MVS surface meshes still suffer from problems such as occlusion, shadow, and weak/repetitive texture. Compared with LiDAR point clouds, MVS meshes contain more noise. In addition, raw MVS surfaces usually contain many geometric and topological defects, such as self-intersections, noise in consistency, nonmanifold edges, dynamic parts, and diffuse color.

Recent research works have been proposed to output reconstruction models of low complexity and strong regularity in the form of semantic LODs. When working on city-scale data, rendering such dense and large scenes can be challenging due to its high memory requirement. The method presented in [48] automatically reconstructs multiple coherent LODs from surface meshes generated by MVS systems. The main steps of the algorithm, including classification, abstraction, and reconstruction, are depicted in Fig. 13. The proposed method partitions the scene into four classes by applying a geometric feature preserving Markov random field (MRF). This method aims to alleviate the scalability and robustness through a greedy process of abstraction, filtering, and simplification. First, the

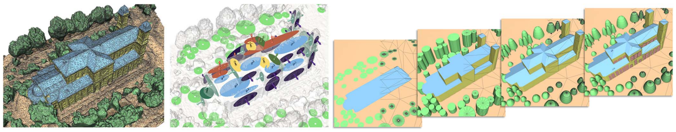


Fig. 13. Example of LOD generation from Urban Scenes from [48] showing three major steps of LOD generation including classification, abstraction, and reconstruction.

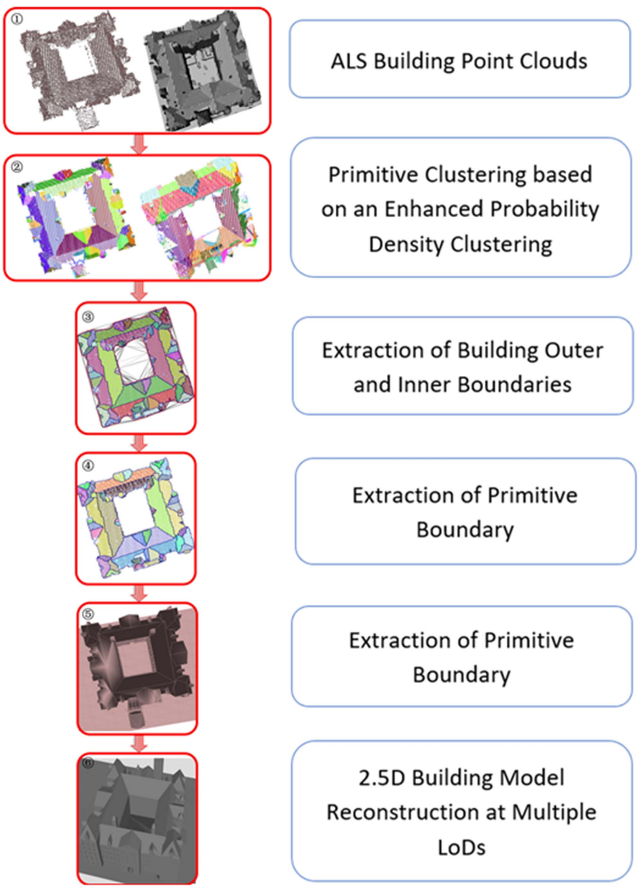


Fig. 14. Example of five LOD modeling. Topologically aware building rooftop reconstruction includes three main steps: rooftop primitive clustering, representation of primitive boundary, and reconstruction of 2.5-D building models at multiple LODs [49].

canonical geometric relationships are hierarchically organized to regularize different planes. The simplified data involve characteristic icons and planar proxies, which are used as input to the reconstruction process to generate watertight models of buildings. Reconstruction step involves a min-cut formulation on a set of 3-D planar arrangements, which is applied to provide robustness to input mesh defects.

Similarly, the method proposed in [48] offers a robust and scalable solution for 3D urban reconstruction. However, it cannot capture the geometries precisely. Chen et al. [49] present a topologically aware 2.5-D building reconstruction methodology from ALS point clouds. The proposed 3-D reconstruction method generates building models at five levels of details using primitive clustering and boundary representation, as shown in Fig. 14. In primitive clustering, an enhanced probability density clustering algorithm is proposed to cluster the rooftop primitives considering the topological consistency among primitives. In boundary representation, a novel Voronoi subgraph-based algorithm is employed to seamlessly trace the primitive boundaries. The aim of the proposed approach in [49] is to maintain the topological consistency to produce watertight and compact 2.5-D polyhedral rooftop models. Additionally, the proposed approach generates hybrid key points from primitive boundaries to reconstruct lightweight and regular geometric models while reducing the crack defects among adjacent primitives.

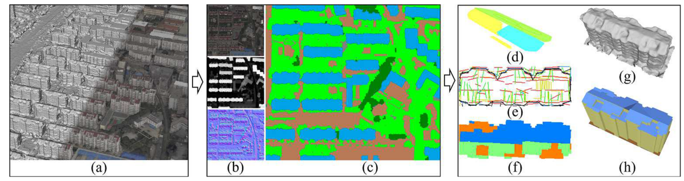


Fig. 15. Large urban scene modeling pipeline of input MVS mesh (step a) from [50] describing various intermediate steps including semantic segmentation [steps (b)–(c)] and building modeling [steps (d)–(h)].

One of the major challenges in the previous research is the computation cost. To curb the computation burden, Zhu et al. [50] cut the input mesh into multiple memory-manageable blocks where each block is processed in parallel. This method provides a solution for robustness by combining the geometric and appearance cues. Taking MVS systems meshes of the large urban scenes as input, the method outputs simplified models at different levels of details while preserving semantics. As shown in Fig. 15, the proposed approach consists of two major steps: segmentation and building modeling. The scene is first segmented into four classes with an MRF combining height and image features. Segmentation involves grid sampling while considering geometry as well as appearance information from the orthographs. Therefore, this method simplifies the 3-D modeling into a 2-D shape labeling, which makes the modeling relatively fast.

Han et al. [51] explain a modeling framework for aerial images and textured 3-D models based on large-scale urban scenes that generate compact polygonal models with semantics at a different LOD. As shown in Fig. 16, the method comprises different stages such as scene segmentation, roof contour extraction, and building modeling. First, a deep neural network segments the scene into three classes, such as ground, vegetation, and building. Then, the 2-D line segment-based roof contours are detected that divide the ground into polygon cells. A roof plane gets assigned to each polygon cell with the help of MRF optimization technique. Finally, building models with different LODs are obtained by extruding cells to different planes.

D. Deep Learning Based Algorithms

Over the past decade, the ample accessibility of large training datasets has facilitated the researchers with the possibility of working on data-driven techniques to produce large 3-D urban models using data priors. These learning-based models are capable of handling big datasets of urban scenes having lots of different objects, such as buildings, roads, vegetation, and

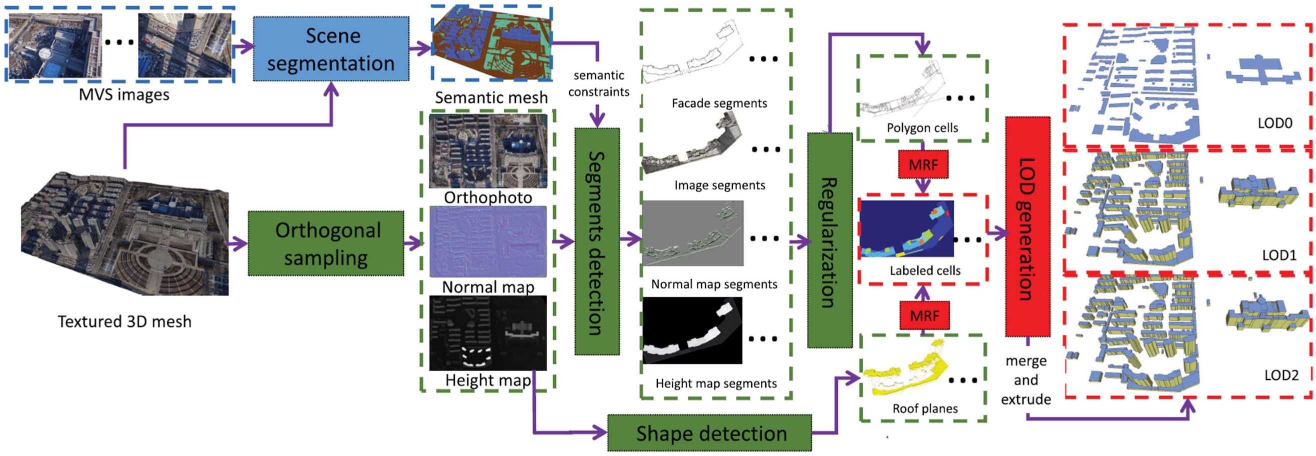


Fig. 16. Urban scene LOD vectorized modeling of photogrammetry meshes including three major stages: scene segmentation, roof contour extraction, and building modeling [51].

parking area. Many deep learning based methods [52], [53], [66], [67], [68] have address the urban reconstruction from multiview images or satellite imagery, while point cloud-based urban reconstruction [54], [55] and lightweight reconstruction approaches are considered by fewer researchers. As follows, a number of deep learning urban reconstruction approaches with the aim of lightweight reconstruction are reviewed.

CNN model discussed in [66] examines the utility of lightweight baseline architectures for scene and appearance changes in satellite images due to different seasons. This article studies the necessity of authentic selection of image pair in multiview 3-D reconstruction. This method presents a compact model requiring lesser number of network parameters due to weights shared within concatenated layers which helps in the training of the CNN model. Therefore, such lightweight CNNs can be used for semantic segmentation for 3-D reconstruction. Another multiview reconstruction network RED-Net introduced in [52] achieves high efficiency and resolution in large-scale reconstruction with lesser memory requirements. Similarly, a very recent web-based interactive platform VGI3D [53] works with VGI images to generate lightweight 3-D building models using CNN. This platform realizes fast solutions with lower time and labor costs. Furthermore, CNN helps in detecting building façades of different and complex architectural style buildings.

Similar to CNNs, generative adversarial network (GAN) is another deep learning approach that is gaining popularity in the image-based 3-D reconstruction community. FrankenGAN [67] uses a lightweight reconstruction process to obtain realistically detailed mass models. In this work, the memory requirement of GAN architecture is compensated by employing individual texturing of windows and super-resolution GAN. In this method, authors claim to provide a compact representation for large-scale 3-D city models. However, they have not provided any quantitative analysis to show the model compactness. Furthermore, this representation lacks sufficient detailing to guarantee seamless textures at boundary areas. An improvement to FrankenGAN is proposed in [68], which has replaced BicycleGAN with StarGAN architecture to generate higher quality textures.

Point clouds carry 3-D spatial information for efficient 3-D reconstruction of objects or scenes. In a recent deep learning based framework [43], authors present a method to reconstruct compact, watertight, polygonal 3-D building models from point clouds with the help of a learnable implicit field using a deep neural network. In this method, the implicit field extracts a smooth surface model of the object by directly learning from the point cloud and MRF extracts the compact surface of the building through combinatorial optimization. Another neural network proposed in [54] and [55] uses 2.5-D dual-contouring method to produce lightweight 3-D models from ALS point clouds of large residential areas. Both of these neural networks do not require large training data for network parameter learning for point cloud labeling, which results in the significant reduction in the parameter tuning cost. Overall, both approaches are suitable for 3-D reconstruction of irregular, complex roof components as well as small structures. However, the deep reinforcement learning framework proposed in [54] sometimes struggles in learning discriminative features from smaller point clouds. Furthermore, the rectified linear unit neural network proposed in [55] is limited by its high computation time.

Bauchet and Lafarge [69] reconstruct urban environments as lightweight polyhedral meshes. While most of the buildings can be reconstructed accurately, the proposed scheme needs improvement in dealing with free-form shapes and small structures of buildings. In addition, this method processes large data volumes in short time, and hence, this approach is sufficiently fast and scalable. Moreover, the authors have not provided any quantitative analysis to show the lightweight aspect of the proposed work.

E. Mesh Decimation

Mesh decimation deals with the reduction in complexity of mesh structures by reducing faces. Therefore, this technique contributes in enhancing the memory efficiency of the urban reconstruction method. One of an earlier work [56] proposes a user-assisted mesh simplification method that converts CAD

TABLE IV
LIGHTWEIGHT URBAN SCENE 3-D RECONSTRUCTION ALGORITHMS

Sl.No	Algorithm	Paper	Input Type	Output Representation	Strengths	Limitations
1	Geometric Abstraction	[3]	Image	Mesh	Sharp and high quality textures. Geometry and texture completion.	It is unable to generate entirely incomplete geometry, sometimes fails to geometrically distinguish the small and non-planar objects.
2	Deep Learning based method	[41]	Point Cloud	Mesh	Compact, watertight and high-quality reconstruction. Computationally efficient.	It may fail in case of incomplete or errorneous primitives.
3	Geometric Abstraction	[42]	Mesh	Mesh	Effective façade reconstruction in case of occlusion, outliers, and other artefacts.	High processing power consumption in case of multiple-view images.
4	Geometric Abstraction	[43]	Point Cloud	Procedural model	High resolution reconstruction and effective with occluded and noisy data.	Larger computation time and ineffective at intersection of two structures.
5	Mesh Decimation	[37]	Mesh	Mesh	Accurate, computationally efficient, manifold, and watertight reconstruction. Effective in defective meshes.	Limited to individual building reconstruction. Requirement of recovery of building primitives and adjacency relationships.
6	Geometric Abstraction	[44]	Point Cloud	Primitive-based Polyhedral Model	Accurate reconstruction of highly curved buildings. Enhanced storage data management and processing.	Full autonomous workflow is difficult in case of mixed types of objects, curved, and polyhedrons.
7	Procedural Modeling	[45]	Image Maps	Polygon + Procedural	High scene detailing and complexity with detailed textures. Suitable for large variety of building types.	Requirement of building ground-plan and manual inspection of façade elemental regularity and size.
8	LOD Modeling	[46]	Mesh	Mesh	Watertight, robust, scalable, and meaningful LODs generation on complex buildings and large-scale urban scenes.	Reconstruction error due to limited classes, classification error for irregular non-flat ground buildings, and insufficient for freeform structures.
9	LOD Modeling	[47]	Point Cloud	Meshes	Watertight and accurate building model with flexible rooftop modeling, capable of generating five LODs in real time.	Insufficient for low quality point clouds of complex rooftops. Trade-off between algorithm flexibility and photorealism of models.
10	LOD Modeling	[48]	Mesh	Meshes	Robust and Fast reconstruction of LOD models of large urban scene from noisy and occluded MVS meshes.	Slightly higher error for LOD2 model for maintaining the balance among model complexity, accuracy, and regularity.
11	LOD Modeling	[49]	Meshes	Mesh	Efficient, robust, and accurate generation of LOD0-LOD2 vectorized building models without requiring global prior. Preserves sharp features.	Low detailing in reconstructed models.
12	Deep Learning based method	[50]	Images	Point Cloud	Generalizable, accurate, and complete, large-scale aerial MVS reconstruction with low GPU memory requirement.	Model efficiency depends on high number of NN layers and parameters.
13	Deep Learning based method	[51]	Images and User Sketching	Point Cloud	Interactive platform. Suitable for real time applications and complex scenes. Efficient for low quality images with limited views. Fast reconstruction.	User interaction is required to obtain various building elements in images.
14	Deep Learning based method	[52]	Point Cloud	Point Set	Improved generalization. High accuracy. Suitable for complex roofs from incomplete/noisy point clouds. Preserves small structures.	Unable to sustain the semantic information.
15	Deep Learning based method	[53]	Point Cloud	Point Cloud	Efficient reconstruction from noisy and occluded data. Suitable for complex building roofs with small structures.	Loss of semantic information. Low computing efficiency for regular roof.
16	Mesh Decimation	[54]	Mesh	Mesh	Simplified and accurate meshes with different LODs and no holes. Capable of model simplification to single LOD with boundary preservation.	User assistance required and memory depends on the model complexity.
17	Mesh Decimation	[55]	Mesh	Mesh	Fast computation while preserving the spectral properties of input surface.	Complex cost evaluation criteria and edge flips result in missing the whole part.
18	Mesh Decimation	[56]	Mesh	Mesh	LOD1-LOD3 reconstruction while maintaining the topological consistency.	Limited efficiency for large-scale city scenes and manual assignment of parameter a is required.
19	Geometric Abstraction	[57]	Point Cloud	Primitive-based model	Compact reconstruction with ability to handle noisy point clouds.	Incorrect reconstruction in case of sparse point clouds, severe occlusion and holes.
19	Mesh Decimation	[17]	Mesh	Mesh	Compact and generalized reconstruction. Suitable for both planar and non-planar regions.	High computational complexity for large free-form surfaces. No guarantee for perfect alignment of refined vertices with abstracted planes.

models to triangle meshes and performs the simplification of each subobject independently at different LODs. In this way, the user can desire a total number of triangles in the simplified model while some parts of the model are maintained or simplified to a definite percentage of simplification. Furthermore, different levels of detail for different subobjects avoid the appearance of holes and preserve the boundaries between the subobjects.

In one of the recent work [37], authors propose a novel approach for the polygonization of MVS meshes of buildings that results in compact and topologically valid models. For polygonization, the planar components of the input mesh along with their topology in the 3-D space get detected using region growing technique. Further, an initial set of candidate's faces is generated to approximate the meshes. The optimization process constructs the simplified surface models considering sharp features through a building scaffold and faces through 2-D arrangements. Another mesh simplification method proposed in [17] reduces the complexity and storage size of the urban models while preserving the sharp contours of building models. This method comprises filtering and simplification of 3-D building mesh models to preserve piecewise planar structures.

Lescoat et al. [57] propose a method to simplify a mesh using edge collapses while targeting to preserve the input eigenvectors

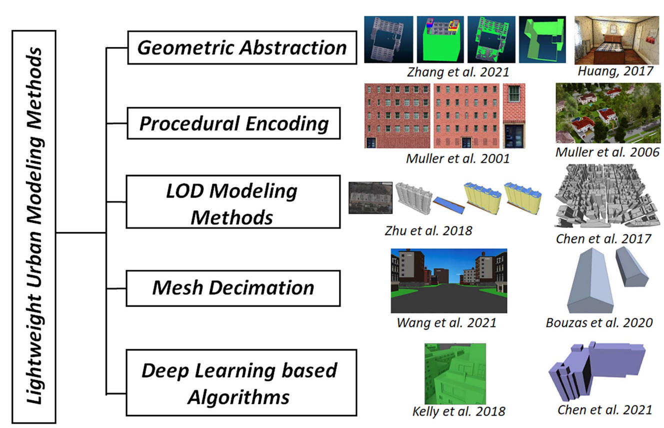


Fig. 17. Summary of lightweight algorithms for reconstruction of 3-D urban scenes.

and eigenvalues through functional maps (a linear mapping between function spaces). This method supports the preservation of spectral properties and offers the similar storage size of simplified methods based on [36] but with a higher quality Laplacian. In [58], a topology-preserving mesh simplification method is proposed for 3-D building models. First, the method classifies building into different segments such that each segment represents different components. Then, the vertices of the model get divided into different categories such as boundary vertices, hole vertices, and other regular vertices. Later, the cost related to edges is determined. For instance, for a boundary edge, the angle between the edge and component (E-C angle) is introduced to estimate an error metric to skip the edge collapses at adjacent areas of building components. An improved quadratic error metric is explained further to address the unexpected error in the case of hole vertices.

Existing lightweight 3-D reconstruction systems (as shown in Fig. 17) have been designed under independent conditions with different prominence; therefore, it is not feasible to compare them straight. However, Table IV lists these urban scene lightweight reconstruction algorithms discussed in this section and their strengths and limitations along with the corresponding input and output representation.

F. Summary and Remarks

We summarize our review on various lightweight reconstruction algorithms based on their applicability, advantages, and disadvantages in the context of urban scene reconstruction including man-made environments.

Geometric abstraction-based methods are reliable for understanding and analyzing urban scene layouts as well as indoor 3-D environments. Major part of an urban scene comprises buildings with dominant planar regions; therefore, primitive-based methods appear suitable for the reconstruction of 3-D urban models. Geometric abstraction-based techniques have been employed in multiple tasks such as façade parsing and segmentation of ALS point clouds into various urban object categories, such as buildings, ground-objects, and vegetation. Later, these segmented point clouds can be substituted with CAD-based primitives in order to address lightweight aspect. However, these methods are limited to simpler approximations with primitive types, such as plane, cylinder, and cone. Therefore, they cannot be applied to complex objects/scenes in real-world examples.

Many existing 3-D modeling works have shown the efficacy of procedural modeling for 3-D reconstruction of urban scene which comprises buildings, vegetation, and roads. These methods can produce high visual quality and low-cost models by using production rules iteratively. Therefore, procedural encodings can be applied to entertainment-related applications, such as 3-D gaming, 3-D movies/animations, and VR/AR applications. Procedural modeling uses a set of rules to define the geometry of the shape, and hence, they can be useful in encoding large city scenes which comprise a large number of urban objects. However, procedural rule defining is a labor-intensive task. On the other hand, IPM automatically derives meaningful split grammars from the sample layouts and uses them to generate 3-D models. However, generation of user-desired 3-D models is still a difficult task and it requires huge expertise. Furthermore, real-world examples, such as European-style buildings, comprise complex architectural variations, which requires a large number of procedural rules. Therefore, procedural encoding-based methods do not appear to be a good choice for representing man-made physical city scenes.

LOD modeling methods have shown prominent results for generating lightweight yet detailed representations. These methods have been used in various works that take aerial images or/and textured 3-D models as input to generate compact polygonal models. To automatically reconstruct city scenes with different LODs, the method pipeline with subtasks such as scene segmentation and feature extraction appears appropriate for efficient representation. Scene segmentation helps in categorizing the urban objects into buildings, vegetation, and ground while feature exaction helps in obtaining rooftop contours. Further, some methods incorporate mesh simplification to reduce polygon count in order to create lightweight models. However, similar to procedural modeling, these methods also require user assistance, and hence, they do not favor the reconstruction of the large-scale city scenes.

Recently, deep learning-based methods for 3-D urban reconstruction have been introduced with the availability of large training datasets of different urban objects, such as buildings, roads, and vegetation. Due to the unorganized nature of point clouds, training a neural network directly from the point cloud is not an easy task. In a recent work [43], a deep learning framework is presented for reconstructing compact and watertight polygonal building models from point clouds in which learnable implicit fields are used to characterize 3-D surface extracted through MRF. Although this method generates valid models with accurate structural geometry, missing LOD makes such models unsuitable for an interactive urban scene application. However, neural networks can be optimized to learn faster and better, not only to approximate 3-D shapes but also to render complex details from the point clouds.

There are many existing works on mesh decimation for reconstructing urban scenes as polygonal meshes from dense LiDAR point clouds, which offers detailed topological information of the 3-D shape. Although point clouds comprise noise, outliers, and other defects, techniques such as mesh filtering and denoising can help in smoothing the noise and uneven shape density produced due to approximation of meshes. By reducing the face count, complexity of mesh structures can be reduced, thereby, making meshes as one of the prominent candidates for lightweight reconstruction. Various models based on mesh simplification and polygonization offer lightweight representation of the geometry; however, these models lack the required LOD due to their compact nature. For instance, façade level details and complex architectural patterns are often omitted or ignored in various methods. Furthermore, various works often discuss about the detailed simplified mesh models by incorporating various schemes, such as use of local correspondents, custom edge collapse operations with feature, and local geometric error metrics. However, in the context of reconstructing detailed urban scene, these techniques are not adequate, and hence, they require further processing to improve the detailing in the reconstructed 3-D model.

IV. Future Directions

The research on lightweight 3-D urban reconstruction is still in infancy due to insufficient work targeting on the memory and rendering requirement of the reconstructed models.

Deep learning method is another promising approach in the context of lightweight urban reconstruction. Advances in deep learning methods for image processing and point cloud processing have shown outstanding results in the development of large-scale 3-D models. Neural network based modeling can easily leverage the high dimensionality in the data for enhancing the quality of the reconstruction. However, large-scale neural network based reconstructions are still restricted due to demand for large GPU memory for gradient-based optimization in backpropagation. Further, sparsity in the input dataset also acts as a bottleneck in using deep learning models for large-scale reconstruction. Therefore, some of the future directions can be listed as follows in Sections IV-A–IV-E.

A. Semantic Guided Mesh Decimation

Use of mesh modeling algorithms for 3-D reconstruction is very common in many applications. However, polygonal meshes contain geometric defects such as nonuniform density and noise along with thousands to millions of faces which enhance the storage memory and create complexity for visualization and data-transfer of real-world applications. Therefore, filtering and simplifying 3-D building mesh models by reducing the number of faces while preserving the original structure with meaningful levels of detail is recommended as a future work. The method proposed in [17] is a good example of a mesh filtering technique which yields piecewise planar regions and extracts crease contours to process single buildings in the scene one by one to preserve planar structures and sharp features. Filtering is further followed by a hierarchical mesh decimation through a series of edge collapse operations to reduce the face count. Another way to approach mesh decimation technique is the semantic-guided decimation. In this method, first each individual building is sementically segmented into various primitives (plane, free-form patches, and other analytical primitives), and then, different levels of mesh decimation are implemented for different primitives, such as loose decimation for complex surface and heavy decimation for regular shapes formed by some planes or analytical primitives. This method helps in avoiding the simplification using a fixed parameter to maintain the geometric accuracy and adds the semantic information for models. Thus, mesh decimation is an evolving research area that can be beneficial for lightweight reconstruction of mesh-based 3-D models.

B. Reconstruction of Freeform Architectures

With the rising interest in the architectural research, the study of architectural ornaments has attracted a lot of attention over the past decade. Different architectural ages have different ornamentations, such as geometric and plant-based ornaments in roman age, calligraphic or vegetal decorations in Islamic age, and cylindrical columns with semicircular arches in renaissance age. Fine details and design patterns along with the curved surfaces play an important role in these architectures. The 3-D reconstruction of these ornate buildings can aid in the study of the evolution of these architectural ornaments from antiquity to the contemporary. However, very few existing works have addressed the urban reconstruction of buildings with curved elements, which is very common in the architecturally rich building structures. Furthermore, fine detailing requires more storage space for the reconstructed models. Hence, large-scale reconstruction of urban scenes with these types of buildings requires more research in context of model storage capacity.

C. NeRF-Based Synthesis

Scene synthesis using neural radiance fields is an emerging area that can address urban reconstruction even from the sparse input dataset. Neural radiance field [70] is a neural-based technique to reproduce a single input scene. It is a fully connected network which maps 5-D input of a scene, i.e., spatial location (x,y,z) and viewing direction (θ, ϕ) to 4-D output (view-dependent RGB color and volume density). Classical volume rendering technique is used to differentially render new 2-D image views. The error minimization through gradient descent results in the prediction of a coherent model of the scene using assignment of high volume densities and accurate colors to the true scene-content locations. Although this method uses volumetric representations to define complex geometry and appearance, it overcomes the high storage cost constraint of discretized voxel grids through hierarchical volume sampling. Therefore, it requires lesser storage memory in comparison to the input image. Block NeRF [71] has used neural radiance fields for reconstruction of a large neighborhood area from 2.8 M images. It requires memory to store only network parameters to define large building models. This method is flexible and scalable to large-scale modeling. However, for large scenes, rendering time is significantly high and network requires more than one computation devices. Furthermore, retraining is required in case of expansion or modification of the environment. Thus, further research on the reduction of rendering time or exploiting the building similarities to reduce the retraining of the network can yield the opportunity of utilizing the NeRF technique for efficient lightweight 3-D urban reconstruction.

D. Integration of Traditional Methods With Deep Learning

One way to deal with the high storage requirements of ML-based 3-D reconstruction methods could be the use of primitive extraction technique for mesh or point cloud segmentation. Geometric primitives can be efficient in determining various urban objects or even different building elements with low storage cost; however, such reconstructed models are coarser in nature. On the other hand, regularity and similarity in the urban building layouts is advantageous for enhancing the memory efficiency in large-scale urban models. These features are utilized by grammar-based reconstruction methods, such as procedural and IPM. Such techniques require smaller memory to store the grammars of the buildings due to the repetitive nature in the layout of various urban objects of the large-scale scenes. Therefore, the integration of traditional technique, such as procedural modeling [72] or IPM, with the deep learning could be another promising direction for lightweight urban reconstruction.

E. Lightweight Reconstruction With Details

One of the major limitations of existing lightweight reconstruction techniques is the lack of details in the reconstructed models. On the other hand, fine details in the 3-D urban models are vital part of applications, such as city planning and simulation of different environmental conditions, study of urban heat island effect, and rising pollutant concentrations. These applications require digital twin of the cities having thousands of urban objects with detailed structural elements of the buildings. Therefore, another future direction is the preservation of details in the lightweight models to enhance the overall quality of the reconstructed model along with the memory efficiency. The 3-D template matching and assembly [73] has shown impressive results in detail enhancement in the reconstructed urban model. Therefore, one possible solution to this problem could be through matching of 3-D templates available in the input template library using GANs, which can help in synthesizing more variants of representative urban elements created using a user interface.

V. Conclusion

With the global emergence of 5G network, there is a strong need of real-time 3-D reconstruction techniques for next-generation applications, such as augmented virtuality or live gaming. For maintaining quality of experience, 3-D reconstruction scheme should be suitable and manageable with 5G infrastructure. The 3-D reconstruction is popular for spatial analysis in the complex urban areas, and it still has enormous development potential. However, generating 3-D models of urban scenes involves high computational cost and storage for efficient processing the geometric details of objects. Therefore, adoption of lightweight techniques for 3-D urban reconstruction is a necessity, especially for the applications such as smart cities, urban planning and surveillance, virtual tourism, and e-sports. This article provides a state-of-the-art review of data structures and algorithms to lightweight reconstruction of urban scenes along with some potential future directions for the generation of economical models with decent level of geometric structure and façade element detail.

References

[1] B. Sheng, F. Zhao, X. Yin, C. Zhang, H. Wang, and P. Huang, "A lightweight surface reconstruction method for online 3D scanning point cloud data oriented toward 3D printing," *Math. Probl. Eng.*, vol. 2018, 2018, Art. no. 4673849.

[2] Z. Kuang, B. Chan, Y. Yu, and W. Wang, "A compact random-access representation for urban modeling and rendering," *ACM Trans. Graph.*, vol. 32, no. 6, pp. 1–12, 2013.

[3] J. Huang, A. Dai, Leonidas J. Guibas, and M. Nießner, "3DLite: Towards commodity 3D scanning for content creation," *ACM Trans. Graph.*, vol. 36, no. 6, 2017, Art. no. 203.

[4] L. Wang and W. Hua, "Survey and practice of 3D city modeling," in *Proc. Int. Conf. Technol. E-Learn. Digit. Entertainment*, 2006, pp. 818–828.

[5] C. Wang, C. Wen, Y. Dai, S. Yu, and M. Liu, "Urban 3D modeling with mobile laser scanning: A review," *Virtual Reality Intell. Hardware*, vol. 2, no. 3, pp. 175–212, 2020.

[6] Y. Ying, M. N. Koeva, M. Kuffer, and J. A. Zevenbergen, "Urban 3D modelling methods: A state-of-the-art review. the international archives of photogrammetry," *Remote Sens. Spatial Inf. Sci.*, vol. 43, pp. 699–7062020.

[7] 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, pp. 146–177, 2013.

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

[9] M. Berger et al., "A Survey of surface reconstruction from point clouds," *Comput. Graph. Forum*, vol. 36, pp. 301–329, 2017.

[10] Y. Xu and U. Stilla, "Toward building and civil infrastructure reconstruction from point clouds: A review on data and key techniques," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2857–2885, 2021.

[11] S. Xia, D. Chen, R. Wang, J. Li, and X. Zhang, "Geometric primitives in LiDAR point clouds: A review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 685–707, 2020.

[12] R. Wang, J. Peethambaran, and D. Chen, "LiDAR point clouds to 3-D urban models: A review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 2, pp. 606–627, Feb. 2018.

[13] P. Tang, D. Huber, B. Akinci, R. Lipman, and A. Lytle, "Automatic reconstruction of as-built building information models from laser-scanned point clouds: A review of related techniques," *Automat. Constr.*, vol. 19, no. 7, pp. 829–843, 2010.

[14] F. Lafarge, "Some new research directions to explore in urban reconstruction," in *Proc. IEEE Joint Urban Remote Sensing Event*, 2015, pp. 1–4.

[15] F. Biljecki, J. Stoter, H. Ledoux, S. Zlatanova, and A. Çöltekin, "Applications of 3D city models: State of the art review," *ISPRS Int. J. Geo-Inf.*, vol. 4, no. 4, pp. 2842–2889, 2015.

[16] J. E. Stoter et al., "State of the art in 3D city modelling: Six challenges facing 3D data as a platform," *GIM Int., Worldwide Mag. Geomatics*, vol. 34, 2020, Art. no. 10.

[17] M. Li and L. Nan, "Feature-preserving 3D mesh simplification for urban buildings," *ISPRS J. Photogrammetry Remote Sens.*, vol. 173, pp. 135–150, 2021.

[18] G. Bruce Baumgart, "Winged edge polyhedron representation," Tech. Rep., Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, 1972.

[19] M. Mäntylä, *An Introduction to Solid Modeling*. Rockville, MD, USA: Computer Science Press, 1987, pp. 439–447.

[20] S. Marschner, "Triangle meshes," [Online]. Available : https://http://www.cs.cornell.edu/courses/cs4620/2013fa/lectures/07trimesh.pdf

[21] B. Ameri and D. Fritsch, "Automatic 3D building reconstruction using plane-roof structures," *ASPRS*, 2000, Art. no. 47.

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

[23] Wikimedia Commons, "File:csg tree.png – wikimedia commons, the free media repository," 2020. Accessed: Dec. 30, 2022. [Online]. Available: https://commons.wikimedia.org/w/index.php?title=File:Csg_tree.png&oldid=485443649

[24] A. Kaiser, A. JoseY. Zepeda, and T. Boubekeur, "A survey of simple geometric primitives detection methods for captured 3D data," *Comput. Graph. Forum*, vol. 38, pp. 167–196, 2019.

[25] 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.

[26] 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.*, 2006, vol. 2, pp. 2213–2220.

[27] Q. Wu, K. Xu, and J. Wang, "Constructing 3D CSG models from 3D raw point clouds," *Comput. Graph. Forum*, vol. 37, pp. 221–232, 2018.

[28] Wikipedia contributors, "Boundary representation—Wikipedia, the free encyclopedia," 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Boundary_representation&oldid=1087182572. Accessed: May 23, 2022.

[29] Christoph M. Hoffmann, *Geometric and Solid Modeling*. San Mateo, CA, USA: Morgan Kaufmann, 1989.

[30] R. Fellegara, K. Weiss, and L. De Floriani, "The stellar decomposition: A compact representation for simplicial complexes and beyond," *Comput. Graph.*, vol. 99, pp. 322–343, 2021.

[31] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Proc. 4th Eurograph. Symp. Geom. Process.*, 2006, vol. 7, pp. 61–70.

[32] Gaël Guennebaud and M. Gross, "Algebraic point set surfaces," *ACM Trans. Graph.*, vol. 26, pp. 23–es, 2007.

[33] M. Rouhani, F. Lafarge, and P. Alliez, "Semantic segmentation of 3D textured meshes for urban scene analysis," *ISPRS J. Photogrammetry Remote Sens.*, vol. 123, pp. 124–139, 2017.

[34] J. Chibane and G. Pons-Moll, "Implicit feature networks for texture completion from partial data," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 717–725.

[35] A. Saint, A. Kacem, K. Cherenkova, and D. Aouada, "3DBooSTeR: Body shape and texture recovery," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 726–740.

[36] M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," in *Proc. 24th Annu. Conf. Comput. Graph. Interactive Techn.*, 1997, pp. 209–216.

[37] V. Bouzas, H. Ledoux, and L. Nan, "Structure-aware building mesh polygonization," *ISPRS J. Photogrammetry Remote Sens.*, vol. 167, pp. 432–442, 2020.

[38] R. Fellegara, F. Iuricich, L. De Floriani, and U. Fugacci, "Efficient homology-preserving simplification of high-dimensional simplicial shapes," *Comput. Graph. Forum*, vol. 39, pp. 244–259, 2020.

[39] E. Stamboulouglou and J. Shan, *Building Modeling and Visualization for Urban Environment*, 2002.

[40] H. Ming, D. Yanzhu, Z. Jianguang, and Z. Yong, "A. topological enabled three-dimensional model based on constructive solid geometry and boundary representation," *Cluster Comput.*, vol. 19, no. 4, pp. 2027–2037, 2016.

[41] A. Martin Fischler and Robert 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.

[42] R. Schnabel, R. Wahl, and R. Klein, "Efficient Ransac for point-cloud shape detection," *Comput. Graph. Forum*, vol. 26, pp. 214–226, 2007.

[43] Z. Chen, S. Khademi, H. Ledoux, and L. Nan, "Reconstructing compact building models from point clouds using deep implicit fields," *ISPRS J. Photogrammetry Remote Sens.*, vol. 194, no. 6, pp. 58–73, 2022.

[44] Y. Zhang, C. Zhang, S. Chen, and X. Chen, "Automatic reconstruction of building Façade model from photogrammetric mesh model," *Remote Sens.*, vol. 13, no. 19, 2021, Art. no. 3801.

[45] W. Li, G. Wolberg, and S. Zokai, "Lightweight modeling of urban buildings from range data," in *Proc. Int. Conf. 3D Imag., Model., Process., Visual. Transmiss.*, 2011, pp. 124–131.

[46] J. Song, S. Xia, J. Wang, and D. Chen, "Curved buildings reconstruction from airborne LiDAR data by matching and deforming geometric primitives," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1660–1674, Feb. 2020.

[47] Y. I. H. Parish and P. Müller, "Procedural modeling of cities," in *Proc. 28th Annu. Conf. Comput. Graph. Interactive Techn.*, 2001, pp. 301–308.

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

[49] 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, Dec. 2017.

[50] L. Zhu, S. Shen, X. Gao, and Z. Hu, "Large scale urban scene modeling from MVS meshes," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 614–629.

[51] J. Han et al., "Urban scene LOD vectorized modeling from photogrammetry meshes," *IEEE Trans. Image Process.*, vol. 30, pp. 7458–7471, 2021.

[52] J. Liu and S. Ji, "A. novel recurrent encoder-decoder structure for large-scale multi-view stereo reconstruction from an open aerial dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6050–6059.

[53] H. Fan, G. Kong, and C. Zhang, "An interactive platform for low-cost building modeling from VGI data using convolutional neural network," *Big Earth Data*, vol. 5, no. 1, pp. 49–65, 2021.

[54] L. Zhang and L. Zhang, "Deep learning-based classification and reconstruction of residential scenes from large-scale point clouds," *IEEE Trans. Geosci. Remote Sens.*, 56, no. 4, pp. 1887–1897, Apr. 2018.

[55] L. Zhang, Z. Li, A. Li, and F. Liu, "Large-scale urban point cloud labeling and reconstruction," *ISPRS J. Photogrammetry Remote Sens.*, vol. 138, pp. 86–100, Apr. 2018.

[56] C. González, J. Gumbau, M. Chover, F. Ramos, and R. Quirós, "User-assisted simplification method for triangle meshes preserving boundaries," *Comput.-Aided Des.*, vol. 41, no. 12, pp. 1095–1106, 2009.

[57] T. Lescoat et al., "Spectral mesh simplification," *Comput. Graph. Forum*, vol. 39, pp. 315–324, 2020.

[58] B. Wang, G. Wu, Q. Zhao, Y. Li, Y. Gao, and J. She, "A topology-preserving simplification method for building models," *ISPRS Int. J. Geo- Inf.*, vol. 10, no. 6, 2021, Art. no. 422.

[59] L. Xie et al., "Combined rule-based and hypothesis-based method for building model reconstruction from photogrammetric point clouds," *Remote Sens.*, vol. 13, no. 6, 2021, Art. no. 1107.

[60] D. Borrmann, J. Elseberg, K. Lingemann, and A. Nüchter, "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, 2011.

[61] A. Lindenmayer et al. *The Algorithmic Beauty of Plants*, vol. 1. Berlin, Germany: Springer, 1990.

[62] R. Měch and P. Prusinkiewicz, "Visual models of plants interacting with their environment," in *Proc. 23rd Annu. Conf. Comput. Graph. Interactive Techn.*, 1996, pp. 397–410.

[63] P. Prusinkiewicz and A. Lindenmayer, *The Algorithmic Beauty of Plants*. Berlin, Germany: Springer, 2012.

[64] P. Prusinkiewicz and M. James, "Synthetic topiary," in *Proc. Special Int. Group Graph. Interactive Techn.*, 1994, pp. 351–358.

[65] J. Guo et al., "Inverse procedural modeling of branching structures by inferring L-Systems," *ACM Trans. Graph.*, vol. 39, no. 5, pp. 1–13, 2020.

[66] M. Bosch, K. Foster, G. Christie, S. Wang, G. D. Hager, and M. Brown, "Semantic stereo for incidental satellite images," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2019, pp. 1524–1532.

[67] T. Kelly, P. Guerrero, A. Steed, P. Wonka, and N. J. Mitra, "FrankenGAN: Guided detail synthesis for building mass-models using style-synchronized GANs," *ACM Trans. Graph.*, vol. 37, no. 6, pp. 1–14, 2018.

[68] Z. Du, H. Shen, X. Li, and M. Wang, "3D building fabrication with geometry and texture coordination via hybrid GAN," *J. Ambient Intell. Humanized Comput.*, vol. 13, pp. 5177–5188, 2022.

[69] J.-P. Bauchet and F. Lafarge, "City reconstruction from airborne LiDAR: A computational geometry approach," in *Proc. 14th Conf. 3D GeoInfo*, 2019.

[70] B. Mildenhall et al., "NeRF: Representing scenes as neural radiance fields for view synthesis," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 405–421.

[71] M. Tancik et al., "Block-NeRF: Scalable large scene neural view synthesis," 2022, in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8248–8258.

[72] H. Zeng, J. Wu, and Y. Furukawa, "Neural procedural reconstruction for residential buildings," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 737–753.

[73] L. Nan, C. Jiang, B. Ghanem, and P. Wonka, "Template assembly for detailed urban reconstruction," in *Computer Graphics Forum*, vol. 34, pp. 217–228, 2015.

Vivek Kamra received the bachelor's degree in computer science from I.K. Gujral Punjab Technical University, Kapurthala, India, 2018. He is currently working toward the master's degree with the Department of Mathematics and Computing Science, Saint Mary's University, Halifax, NS, Canada.

His research interests include 3-D computer vision, point cloud processing and related applications including LiDAR-based urban scene modeling.

Prachi Kudeshia received the B.Tech. degree in electronics and communication engineering from CSJM University Kanpur, India, in 2013, and the M.Tech. degree in mobile communication from The LNMIIT Jaipur, India, in 2016. She is currently working toward the Ph.D. degree with the Department of Mathematics and Computing Science, Saint Mary's University, Halifax, NS, Canada.

Her research interests include 3-D computer vision, 3-D point cloud processing, graph representation learning, geometric learning, and 3-D urban reconstruction.

Somaye ArabiNaree received the master's and Ph.D. degrees in computer science from the Iran University of Science and Technology, Tehran, Iran, in 2008 and 2014, respectively.

She is an Assistant Professor of computer science with the Faculty of Mathematical Sciences and Computer, Kharazmi University, Tehran, Iran. She is currently collaborating as a Visiting Researcher with the Department of Mathematics and Computing Science, Saint Mary's University, Halifax, NS, Canada. Her main research interests include machine learning, data analysis, image processing, numerical linear algebra, matrix methods in data mining and pattern recognition, recommender systems, and other topics related to data science.

Dong Chen received the bachelor's degree in computer science from the Qingdao University of Science and Technology, Qingdao, China, in 2005, the master's degree in cartography and geographical information engineering from the Xi'an University of Science and Technology, Xi'an, China, in 2009, and the Ph.D. degree in geographical information sciences from Beijing Normal University, Beijing, China, in 2013.

He is currently an Associate Professor with Nanjing Forestry University, Nanjing, China. He is also 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.

Yasushi Akiyama received the master's degree in mathematics and computing science from Saint Mary's University, Halifax, NS, Canada, in 2005, and the Ph.D. degree in computer science from Dalhousie University, Halifax, NS, Canada, in 2015.

He is currently an Assistant Professor with the Department of Mathematics and Computing Science, Saint Mary's University, Halifax. His research interests include human–computer interaction, child–computer interaction, digital creative tools, and multimodal interaction technologies.

Jiju Peethambaran (Member, IEEE) received the bachelor's degree in information technology from the University of Calicut, Malappuram, India, in 2005, the master's degree in computer science from the National Institute of Technology, Karnataka, India, in 2009, and the Ph.D. degree in computational geometry from IIT Madras, Chennai, India, in 2015.

He is currently an Assistant Professor with the Department of Mathematics and Computing Science, Saint Mary's University, Halifax, NS, Canada. His research interests include 3-D computer vision, computer graphics, geometric deep learning, and related applications including LiDAR-based urban modeling.