A novel Building Section Skeleton for compact 3D reconstruction from point clouds: A study of high-density urban scenes

Yijie Wu, Fan Xue*, Maosu Li, Sou-Han Chen

Faculty of Architecture, The University of Hong Kong, Pokfulam, Hong Kong SAR, China

Abstract

Compact building models are demanded by global smart city applications, while high-definition urban 3D data is increasingly accessible by dint of the advanced reality capture technologies. Yet, existing building reconstruction methods encounter crucial bottlenecks against high-definition data of large scales and high-level complexity, particularly in high-density urban scenes. This paper proposes a Building Section Skeleton (BSS) to reflect architectural design principles about parallelism and symmetries. A BSS atom describes a pair of intrinsic parallel or symmetric points; a BSS segment clusters dense BSS atoms of a pair of symmetric surfaces; the polyhedra of all BSS segments further echo the architectural forms and reconstructability. To prove the concepts of BSS for automatic compact reconstruction, this paper presents a BSS method for building reconstruction that consists of one stage of BSS segments hypothesizing and another stage of BSS segments merging. Experiments and comparisons with four state-of-the-art methods have been conducted on 15 diverse scenes encompassing more than 60 buildings. Results confirmed that the BSS method achieves frontiers in compactness, robustness, geometric accuracy, and efficiency, simultaneously, especially for high-density urban scenes. On average, the BSS method reconstructed each scene into 623 triangles with a root-mean-square deviation (RMSD) of 0.82 m, completing the process in 110 seconds. First, the proposed BSS is an expressive 3D feature reflecting architectural designs in high-density cities, and can open new avenues to city modeling and other urban remote sensing and photogrammetry studies. Second, for practitioners in smart city development, the BSS method for building reconstruction offers an accurate and efficient approach to compact building and city modeling. The source code and tested scenes are available at https://github.com/eiiijiiiy/sobss.

Keywords: Building Section Skeleton, Compact building model, 3D point cloud, 3D reconstruction, High-density urban scenes

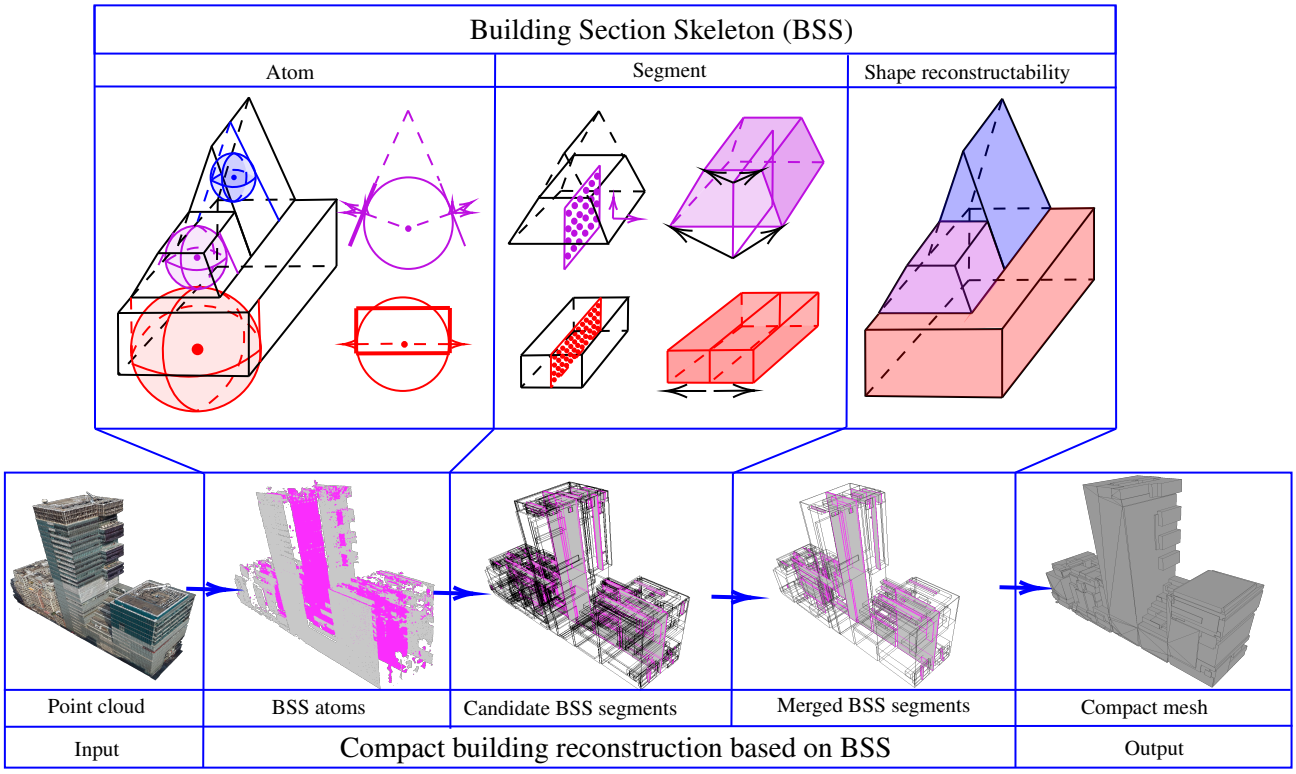
This is the author version of the paper:
Wu, Y., Xue, F., Li, M., & Chen, S.-H. (2024). A novel Building Section Skeleton for compact 3D reconstruction from point clouds: A study of high-density urban scenes, ***ISPRS Journal of Photogrammetry and Remote Sensing Volume 209***, Pages 85-100. doi: https://doi.org/10.1016/j.isprsjprs.2024.01.020

*Corresponding author, Tel: +852 3917 4174, Fax: +852 2559 9457, Email: xuef@hku.hk
Email addresses: yijiewu@connect.hku.hk (Yijie Wu), xuef@hku.hk (Fan Xue), maosulee@connect.hku.hk (Maosu Li), souhan@hku.hk (Sou-Han Chen)

Graphical Abstract

A novel Building Section Skeleton for compact 3D reconstruction from point clouds: A study of high-density urban scenes

Yijie Wu, Fan Xue, Maosu Li, Sou-Han Chen



Highlights

- Building Section Skeleton (BSS) is proposed with novel definitions of BSS atoms and segments.
- BSS revamps traditional shape skeletons to reflect architectural design principles about parallelism and symmetry.
- A BSS method consisting of two stages is developed for compact building reconstruction from urban point clouds.
- The BSS method of reconstruction was confirmed compact, robust, geometrically accurate, and efficient.

1. Introduction

The global smart and sustainable city development has given rise to a growing demand for 3D building and city models with higher levels of details (LoD) in both geometry and semantics (Batty, 2013; Acuto, 2016). 3D building and city models enable novel environmental assessments (Rodríguez et al., 2017; Li et al., 2022b), simulations (Malhotra et al., 2022; Stoter et al., 2020), and analyses (Xue et al., 2020; Labetski et al., 2022) in the fields such as urban planning, sustainable energy management, and urban resilience. Reality capture techniques, such as Light Detection And Ranging (LiDAR) and photogrammetry, are popular urban remote sensing tools in acquiring the 3D data of buildings and cities, by dint of their strong advantages in scanning speeds and densities (Colomina and Molina, 2014; Bisheng et al., 2017). However, the acquired high-density point clouds or meshes are usually of huge volume, have a variety of defects, and lack semantics (Berger et al., 2017). Therefore, building reconstruction that processes the unstructured point clouds into accurate and compact building and city models is critical to leverage the advanced data acquisitions for smart and sustainable city applications.

Automatic building reconstruction in the literature has evolved along with the advancement of reality capture (Haala and Kada, 2010; Xia et al., 2020). When sparse airborne point clouds were available at the urban scale in 2000s, researchers paid more attention to reconstructing simple prismatic building models approximately at LoD1 in CityGML (Zhang et al., 2006; Du et al., 2019). When slightly denser airborne data of building rooftops was available in 2010s, great efforts were taken to reconstruct major roof structures with the vertical extrusions of footprints as facades (Xiong et al., 2015; Li and Shan, 2022). In recent years, realistic 3D building surfaces, in the form of dense point clouds or meshes covering both building roofs and facades, become prevalent (Kölle et al., 2021). On top of the higher-quality input, researchers began to reconstruct buildings with more detailed roofs, facades, and projections (Li et al., 2016; Nan and Wonka, 2017).

The existing methods of detailed building reconstruction can generate compact building models efficiently for small and simple urban scenes but still face crucial bottlenecks of accuracy, compactness, and efficiency when applied in larger and more complex scenarios. Typically, these compact reconstruction methods coordinate their pipeline based on a single type of geometric primitives, which can be categorized into plane-based (Nan and Wonka, 2017; Bauchet and Lafarge, 2020), edge-based (Langlois et al., 2019; He et al., 2021), contour-based (Song et al., 2015, 2020), and triangle-based methods (Li and Nan, 2021; Salinas et al., 2015). These methods follow a similar pipeline containing two steps. The first step detects the basic geometric primitives and partitions them into candidates; and the other step selects and assembles the candidates into a compact mesh.

The plane-based methods primarily depend on local planar structures, i.e., face intersection and adjacency, to reconstruct watertight surfaces effectively (Nan and Wonka, 2017; Bauchet and Lafarge, 2020), though encountering limitations when handling missing and noisy planes. Meanwhile, global features, such as parallelism and symmetry, are employed as regularities to refine the positions and orientations of geometric primitives (Zhou and Neumann, 2012; Verdie et al., 2015). But other potential utility of parallelism and symmetry as global structures, encompassing the internal composition and arrangement of geometric primitives to form volumes, remains largely untapped.

This paper aims to integrate the global structures, with a focus on the intrinsic symmetry of buildings and shape skeletons, in compact building reconstruction. A shape skeleton is a versatile and compact representation (Li et al., 2015; Sun et al., 2015) that captures the intrinsic symmetry and structure of a shape, encodes both adjacent and non-adjacent components, and serves as intermediations for compact reconstruction (Tagliasacchi et al., 2016; Saha et al., 2016). However, native shape skeletons are mainly defined for the general shapes with curved surfaces and even tubular shapes (Blum, 1967; Livny et al., 2010). Consequently, the perceived symmetry of rectangles, especially for corners and edges incident by orthogonal planar components, is poorly reflected by the native skeletons (Giblin and Brassett, 1985).

This paper extends the definition of skeletons for reflecting buildings' global structures in terms of architectural symmetry and parallelism. A method of building reconstruction can then be further developed based on this skeleton. The global structures enable our method to reconstruct buildings from *low-rise, low-density* scenes, e.g., a bungalow, to *high-rise, high-density* scenes, e.g., a complex city block. The global structures of individual buildings and city blocks can also lead the method to the compactness and completeness in the results of reconstruction. The contributions of this paper can be summarized into two aspects:

- A novel Building Section Skeleton (BSS) theory. BSS reflects the general global structures of urban buildings using two new concepts at atom and segment levels. BSS atoms capture the global structures at the point level, while BSS segments group BSS atoms and compactly parameterize the symmetry, parallelism, and orthogonality at the plane level. As a result, the proposed BSS theory equips the established plane-based building reconstruction in literature with global structures, leading to higher robustness and compactness in complex urban scenes.

- A new BSS method for compact building reconstruction. The method consists of two stages. Stage 1 extracts BSS atoms from the input point clouds and proposes a dense candidate set of BSS segments. Stage 2 employs integer programming to merge the candidate BSS segments for accuracy and compactness. Meanwhile, our implementation of the BSS method has also been

released as open-source code to facilitate the research and practice of building reconstruction and smart city development.

2. Literature review

2.1. Building reconstruction from reality capture data

Based on the geometric primitives driving the pipeline, existing building reconstruction methods can be categorized into four categories: (i) plane-based, (ii) edge-based, (iii) contour-based, and (iv) triangle-based.

Plane-based methods. A typical pipeline of plane-based methods (Li et al., 2016; Nan and Wonka, 2017; Bouzas et al., 2020; Wang et al., 2020; Xie et al., 2021; Yang et al., 2022; Chen et al., 2022; Bauchet and Lafarge, 2020) takes the initial planes detected by RANSAC (Schnabel et al., 2007) or region growing (Gatzke and Grimm, 2006; Lafarge and Mallet, 2012) as inputs first. Then, the pipeline assembles the input planes through two key steps: (1) candidate plane or 3D cell partitioning, and (2) candidate selection. With a complete set of accurate initial planes, plane-based methods can further fairly hypothesize and select the candidates (Li et al., 2016; Nan and Wonka, 2017; Bauchet and Lafarge, 2020).

However, the quality and completeness of initial planes can be compromised, particularly when dealing with noisy or incomplete point clouds that disrupt the assumption of face adjacency. Users need to carefully tune the parameters in the plane detection to reduce the missing and noisy planes. This issue is particularly evident in the cases of pencil towers (tall buildings with small footprints), podiums, and low-rise buildings situated between high-rise buildings when reconstructing *high-rise, high-density* scenes. Their 3D point clouds could be noisy or sparse due to the severe occlusions. Besides, their main structures are smaller and more likely to be ignored in plane detection. To resolve this issue, one approach is to detect a greater number of initial planes to ensure the completeness and accuracy, at the expense of reconstruction time and model complexity. Another approach was presented in the work by Yu and Lafarge (2022), where an efficient exploration mechanism was designed based on an energy function to refine the input planes. However, a significant amount of additional computational time is still a downside of this refinement.

Edge-based methods. In edge-based methods (Langlois et al., 2019; He et al., 2021), 3D segments are first extracted from images. Planes or cuboids are then detected from the segments and optimized into the final models. However, edge-based reconstruction still struggles with its efficiency bottleneck.

Contour-based methods. Furthermore, contour-based methods (Song et al., 2015, 2020) extract horizontal contours from point clouds and then cluster and connect the contours based on their locations and 2D shapes. The contour clusters are then fitted with planar and non-planar primitives to assemble the whole building model. The methods were mainly designed for buildings with irregular shapes.

Triangle-based methods. These methods (Li and Nan, 2021; Salinas et al., 2015), usually termed mesh simplification, smooth the planar regions of meshes and collapse the triangular edges to reduce the primitive numbers of a model while preserving shape corners and edges. However, the computational complexity of the region refinement is the square of the number of segmented regions. Therefore, the computation may grow fast when applied the methods to reconstruct large and complex urban scenes. Moreover, there is no clear mechanism in the existing methods to recover or complete the missing parts of the input data.

Motivation to integrate global structures in building reconstruction. Although the face adjacency utilized by previous methods present effectiveness and generality in building reconstruction, methods relying on face adjacency only can still suffer from crucial bottlenecks when applied to more complex urban scenes. Global features, such as parallelism (Zhou and Neumann, 2012) and symmetry (Xue et al., 2019; Wu et al., 2021), are dominating regularities in architecture and turn to be useful and general assumptions. Parallel vertical planes serve as a fundamental form to organize space and distribute the structural forces of architecture (Ching, 2023). In high-density cities, buildings are always oriented to the streets or squares to maintain the harmony among their neighborhoods (TCC, 2013) or for thermal comfort (PlanD, 2022), which also populates the presence of parallel facades. Meanwhile, typical roof types, such as gable, hip, pyramid, and their combinations, are composed of inclined planes that are symmetric (Henn et al., 2013; Xiong et al., 2015; Zhang et al., 2021; Li et al., 2022a). In literature, parallelism and symmetry are mainly used as regularities to refine the positions and orientations of primitives (Verdie et al., 2015; Nan and Wonka, 2017). The extensive potential utility of parallelism and symmetry as global structures, specifically in terms of the internal composition and arrangement of geometric primitives to form volumes, has yet to be fully explored. Therefore, this paper aims to integrate global structures based on the intrinsic symmetry and parallelism into building reconstruction with the concept of shape skeleton.

2.2. *Skeletonization of shapes*

A shape skeleton, referring to the shape's intrinsic medial axes and surfaces, jointly describes the geometry, topology, and symmetry (Tagliasacchi et al., 2016; Saha et al., 2016). In computer graphics, skeletons with a range of desirable properties have been proven especially useful for many classic

tasks, including reconstruction, segmentation, matching, and retrieval (Tagliasacchi et al., 2016; Saha et al., 2016). Even in the context of urban reconstruction, skeletons have also served as effective intermediate representations for tree modeling and synthesis (Bucksch and Lindenbergh, 2008; Livny et al., 2010; Bremer et al., 2013; Xia et al., 2020).

Definitions of shape skeletons. Shape skeletons have been defined at the atom level from four well-established perspectives, deriving various skeletonization methods (Tagliasacchi et al., 2016). A skeleton can be defined as the centers and radii of all the *maximal inscribed balls* (MIBs) of a shape (Blum, 1967). Corresponding skeletonization was implemented by searching or predicting these MIBs (Ma et al., 2012; Jalba et al., 2012; Rebain et al., 2019). Another definition is *Maxwell set* - for each point of a skeleton, there are at least two closest points on the shape boundary (Mather, 1983). This inspires methods (Dey and Zhao, 2002; Tagliasacchi et al., 2012; Li et al., 2015; Sun et al., 2015; Yan et al., 2018; Dou et al., 2022) to take vertices of the Voronoi cells generated from the sampled points of shape boundaries as initial skeleton points (Amenta and Bern, 1998). Skeletons have also been defined as symmetry sets, in which each point is bi-tangent to the shape boundary (Giblin and Brassett, 1985). Symmetry-based skeletonization was mainly implemented for tubular shapes by estimating and connecting their rotational symmetry axes as skeletons (Tagliasacchi et al., 2009; Livny et al., 2010). Another alternative definition is the so-called *grassfire analogy*, which shrinks the shape boundary towards the interior of the shape to form the skeleton. The grassfire analogy definition led to the skeletonization via topology-preserving distance transform or shape thinning (Arcelli et al., 2010).

Skeleton simplification and compact surface reconstruction. Nevertheless, skeletonizing noisy data based on the exact definitions can result in redundant and unstable skeletons (Attali et al., 2009; Sun et al., 2015). Therefore, skeleton simplification is required, in particular for compact building reconstruction focused on in this paper. Exact reconstruction from non-simplified skeletons through reverse distance transform or MIB union produces too many voxels or mesh patches despite high accuracy (Arcelli et al., 2010; Ma et al., 2012; Rebain et al., 2019; Jalba et al., 2012, 2015). While exact reconstruction on reduced skeletons such as curved skeletons sacrifices too much accuracy (Arcelli et al., 2010). So far, to the best of our knowledge, only approximate reconstruction from approximate skeletons simultaneously guarantees accuracy and compactness. The core idea is to represent the subsets of a skeleton, so-called skeleton segments, as parametric curves or surface sheets and approximate shapes with geometric primitives estimated from the parameters of skeleton segments. The primitives used in literature include general cylinders (Tagliasacchi et al., 2009; Livny et al., 2010), cones, and slabs (Li et al., 2015; Sun et al., 2015; Lin et al., 2021; Dou et al., 2022). Furthermore, the

approximation errors can in return guide skeleton simplification (Li et al., 2015; Sun et al., 2015; Lin et al., 2021; Dou et al., 2022).

Problems to apply native skeletons in building reconstruction. The existing definitions and skeletonization of general shapes cannot reflect the shape structures of buildings effectively for accurate and compact reconstruction. The theoretical issue underlying is that native skeleton definitions 'poorly reflect the perceived symmetry of rectangular shapes' (Giblin and Brassett, 1985), while cuboid is the most dominating geometric primitive of buildings. Skeletonization based on these definitions produces inclined curves or surfaces of skeletons for the corners or edges of rectangular/cuboid shapes. Except for pruning these parts of skeletons or approximating the corners and edges by curved surfaces, compact representation of skeletons and corresponding shape approximation regarding the rectangular/cuboid shapes are still lacking. Therefore, this paper aims to extend the definition of skeletons to reflect the intrinsic global symmetry, parallelism, and orthogonality of buildings to facilitate building reconstruction.

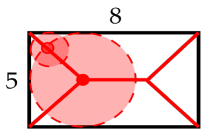
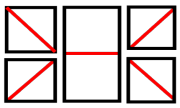
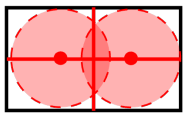
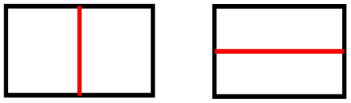
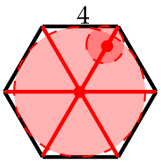
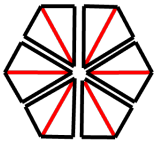
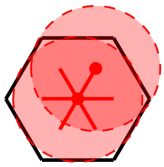
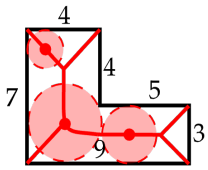
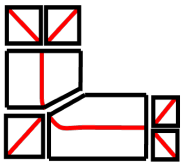
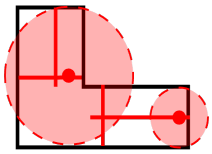
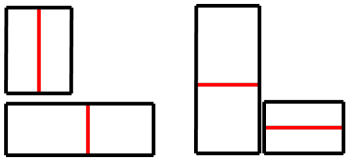
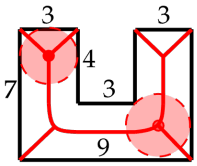
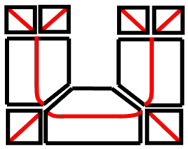
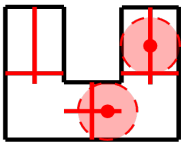
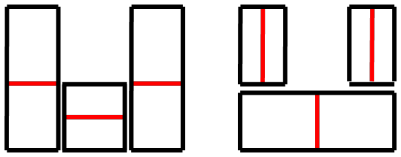
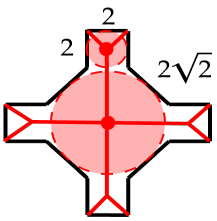
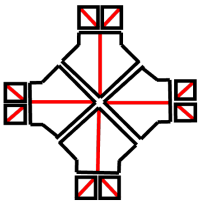
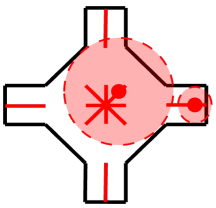
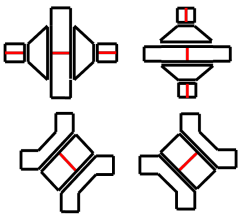
3. The proposed Building Section Skeleton (BSS)

This section presents a novel BSS that revamps traditional shape skeletons to capture parallelism and symmetry in architectural designs commonly seen in urban areas. BSS first introduces a definition at the atom level in Sect. 3.1. The set of closely related BSS atoms is further clustered into BSS segments in Sect. 3.2 to form a compact skeleton representation, of which the corresponding polyhedra and reconstructability drive the task of compact 3D reconstruction in Sect. 3.3.

3.1. Atoms of BSS

The definition of BSS atoms follows the logic of the classical definition by Blum (1967). According to Blum (1967), a skeleton can be defined as the centers and radii of all the shape's *maximal inscribed balls* (MIBs). Each ball, with its center and radius, is an atom of the skeleton. The first column in Table 1 presents the corresponding skeletons and some MIBs. The corners and turns of 2D shapes are reflected by the inclined or curved skeleton segments, which are inconsistent with the perceived symmetry of rectangular shapes (Giblin and Brassett, 1985). However, such corners and turns are everywhere in the buildings and urban blocks with rich rectangular, cuboid, and inclined shapes. Meanwhile, the continuously changing radii along the skeleton segment are expensive to record in order to recover or reconstruct buildings and urban blocks compactly. To resolve these issues, this paper removes the inscription requirement of the ball and defines the BSS atoms to locate at the symmetric planes of buildings, i.e., the straight symmetric axes of 2D footprints in Table 1.

Table 1: Comparison between the MIB-based skeleton and Building Section Skeleton (BSS) of building footprints.

MIB-based skeleton		BSS	
Full skeleton	Segment	Full skeleton	Segment
			
$l \approx 17.14$		$l = 13$	
			
$l = 24$		$l = 12$	
			
$l \approx 21.169$		$l = 16$	
			
$l \approx 24.423$		$l = 20$	
			
$l \approx 27.314$		$l \approx 17.657$	

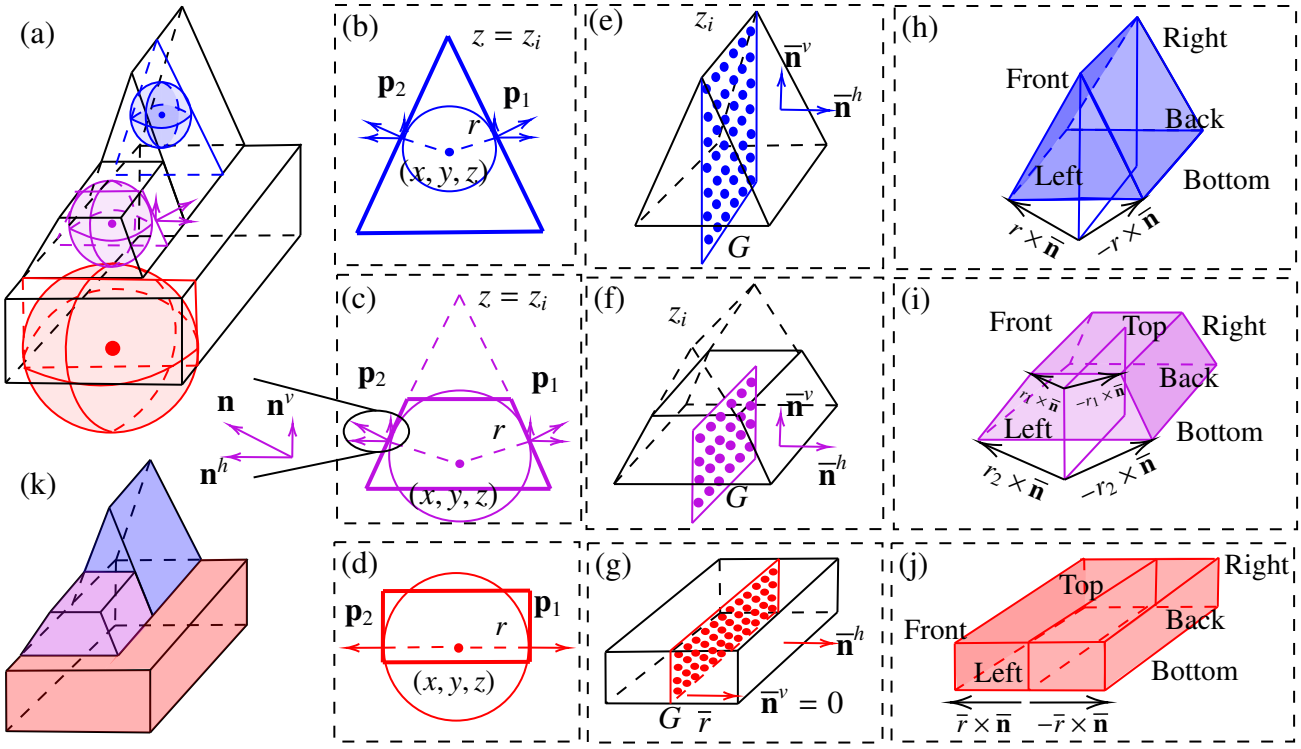


Figure 1: Illustration of 3D BSS. (a) Input shape; (b)-(d) The section graphs of BSS atoms in (a); (e)-(g) The BSS segments of the cuboid, triangular prism, and trapezoidal prism in (a); (h)-(j) Corresponding polyhedra of the BSS segments in (e)-(g); and (k) The shape segmentation and reconstruction based on BSS segments.

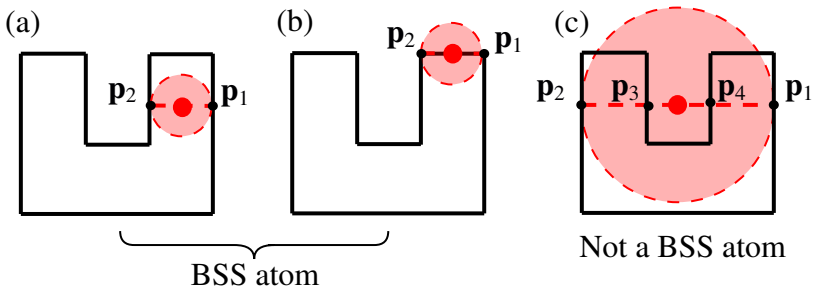


Figure 2: Illustration of how BSS atoms preserve local shape convexity by Cond. 3 of Def. 3.1. (a) and (b) satisfy Cond. 3 while (c) doesn't.

Definition 3.1. *Given a shape O bounded by $\dot{O}$, a BSS atom $\mathbf{a} = (x, y, z, r)$ associates to a ball B bounded by $\mathcal{B}$. B is centered at (x, y, z) with a radius of r. $\exists \mathbf{p}_1, \mathbf{p}_2 \in \dot{O} \cap \mathcal{B}, \mathbf{p}_1 \neq \mathbf{p}_2$, satisfy the following conditions:*

1. $n_{\mathcal{B}}(\mathbf{p}_1) = n_{\dot{O}}(\mathbf{p}_1)$, $n_{\mathcal{B}}(\mathbf{p}_2) = n_{\dot{O}}(\mathbf{p}_2)$;
2. $n_{\mathcal{B}}(\mathbf{p}_1)^v = n_{\mathcal{B}}(\mathbf{p}_2)^v \geq 0$, $n_{\mathcal{B}}(\mathbf{p}_1)^h = -n_{\mathcal{B}}(\mathbf{p}_2)^h$; *and*
3. *Let* $l = \{\mathbf{p}' | \mathbf{p}' = \mathbf{p}_1 + \lambda \overrightarrow{\mathbf{p}_1 \mathbf{p}_2}, 0 \leq \lambda \leq 1\}$. $\dot{O} \cap l = \{\mathbf{p}_1, \mathbf{p}_2\}$ *or* l.

In Cond. 1, $n_{\mathcal{B}}(\mathbf{p}_1)$ and $n_{\dot{O}}(\mathbf{p}_1)$ represent the normals of $\mathbf{p}_1$ tangent at the boundary of B and O, respectively. Note that the 3D normals in this paper are of unit lengths. In Cond. 2, $n_{\mathcal{B}}(\mathbf{p}_1)^v$ and $n_{\mathcal{B}}(\mathbf{p}_1)^h$ are the vertical and horizontal components of $n_{\mathcal{B}}(\mathbf{p}_1)$, respectively. We also constrain the $n_{\mathcal{B}}(\mathbf{p}_1)^v$ to be non-negative, since negative $n_{\mathcal{B}}(\mathbf{p}_1)^v$ is rare in common building shapes. Note that the vertical component is a real number, while the horizontal one is a 2D vector. In Cond. 3, l is a segment connecting $\mathbf{p}_1$ and $\mathbf{p}_2$. This condition is defined to maintain the local convexity of shapes. The first two conditions are illustrated in Fig. 1(a)-(d). Positive and negative examples of Cond. 3 are presented in Fig. 2.

As shown in the last two columns of Table 1, BSS is more intuitive and simpler than MIB-based skeletons for building footprints. BSS can also resolve the issues of complex polylines and curves in MIB-based segments. Furthermore, BSS is much more efficient in terms of length, i.e., l in Table 1. For example, the MIB-based skeletons of building footprints are considerable, i.e., 22-100%, longer than the proposed BSS ones. This shows that fewer atoms are required for BSS than MIB-based skeleton when we use the skeleton atoms at a given interval to represent or store a skeleton, indicating that BSS is a more compact skeletal representation for buildings.

3.2. Segments of BSS

Generally, a BSS segment is a set of spatially connected BSS atoms located at the vertical symmetric planes of a pair of planar segments of buildings (Fig. 1(e) - (g)). On the 2D footprints, a BSS segment is the straight symmetric axis of a pair of parallel segments on the building footprints, as shown in the last two columns of Table 1. Table 1 also compares the segments of MIB-based skeleton and BSS. The segments of MIB-based skeletons induce fragmented segmentation of building footprints, while the segments of BSS partition the footprints into larger parts that reflect the perceived global structures of buildings.

In the 3D cases, let one of the plane normals of a pair of symmetric planar segments be $\overline{\mathbf{n}}$, and its horizontal and vertical component be $\overline{\mathbf{n}}^h$ and $\overline{\mathbf{n}}^v$. The ball of a BSS atom locating on their symmetric plane is tangent to the two planes with normals of $(\overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v)$ and $(-\overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v)$. The formal definition of a BSS segment can be written as:

Definition 3.2. *Given a pair of planar segments of the shape boundary, their symmetric plane is vertical. The normals of the two planar segments are* $(\overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v)$ *and* $(-\overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v)$*. Assume any point on either planar segments can be paired with one point on the other plane to form a BSS atom. The BSS segment of the planar segments is a set of BSS atoms* $\mathcal{S}' = \{\mathbf{a} = (x, y, z, r)\}$.

1. *The atoms are spatially connected; and*
2. *The balls of the atoms are tangent at the given planar segments with normals* $(\overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v)$ *and* $(-\overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v)$.

Based on Def. 3.2, when the two planar segments are vertical, the BSS atoms should share a same radius $\overline{r}$ which is half of the distance between the two planes. However, when the symmetric planar segments are inclined, the radii of BSS atoms decrease as the z-coordinates of the ball centers increase. We suggest a new parameter z_i shared by the BSS atoms in this case. That is the z-coordinate of the line where the two planes intersect, named *intersection elevation* in the following. Given a BSS atom (x, y, z, r) and $\mathbf{n}^v$, it could be easily calculated as $z + r/\mathbf{n}^v$. Visual illustrations of BSS segments are shown in Fig. 1(e)-(g). A lemma about the shared parameters of BSS segments can be induced as:

Lemma 3.1. *Given a BSS segment of a pair of planar segments,*

1. *If the two planar segments are vertical, i.e.,* $\overline{\mathbf{n}}^v = 0$*, the atoms of the BSS segment share a same radius* $\overline{r} = \overline{d}/2$ *where* $\overline{d}$ *is the distance between the two planar segments;*
2. *Otherwise, i.e.,* $\overline{\mathbf{n}}^v \neq 0$*, the atoms share a same intersection elevation* $\overline{z_i} = \overline{z} + \overline{r}/\overline{\mathbf{n}}^v$ *where* $\overline{z}$ *and* $\overline{r}$ *are the average z-coordinate and radius of the BSS atoms.*

If we use G to denote the polygon bounding the BSS atoms of a BSS segment, a BSS segment can be further parameterized as $b = (G, \overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v, \overline{r})$ for vertical symmetric planes or $b = (G, \overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v, \overline{z_i})$ for non-vertical ones. As shown in Fig. 1(j), for a BSS segment of vertical planes, its G can be extruded along $\overline{\mathbf{n}}$ and $-\overline{\mathbf{n}}$ respectively by $\overline{r}$ to construct a cuboid whose front and back faces are the corresponding vertical planes of the shape boundary. Similarly, for a BSS segment of non-vertical planes (Fig. 1(h) and (i)), the vertices of G can be back-projected along $\overline{\mathbf{n}}$ and $-\overline{\mathbf{n}}$ to recover the front and back faces of its polyhedron. The other faces can be constructed by connecting the vertices of the front and back faces. Therefore, the polyhedron of $b = (G, \overline{\mathbf{n}}^h, \overline{\mathbf{n}}^v, \overline{z_i})$ is a triangular or trapezoidal prism. The polyhedron of a BSS segment will be denoted as P in the following.

3.3. Building reconstructibility of BSS

By reflecting the intrinsic symmetry, parallelism, and orthogonality of buildings, BSS segments present promising potentials to motivate building reconstruction by equipping planar segments with

global structures. According to Def. 3.2 and Lemma 3.1, a BSS segment captures a pair of planar segments, corresponds to a polyhedron covered by the shape O, and approximates some parts of the shape boundary (Fig. 1(k)).

BSS is applicable to the surface reconstruction of buildings that are dominated by parallel vertical facades and symmetric inclined roofs discussed in Sect. 2.1. BSS equips the global structures to planes based on parallelism and symmetry, which will enhance the robustness of building reconstruction along with the effective face adjacency. Specifically, each BSS segment captures the two parallel or symmetric planar segments that could be non-adjacent. A BSS segment also induces a polyhedron occupying the interior of buildings, which can be used to validate and guarantee the completeness of building reconstruction. Hence, the inclusion of global structures in the BSS method can serve as a complement to face adjacency, thereby enhancing the robustness of building reconstruction and offering a novel perspective for the advancement of compact building reconstruction.

4. A BSS method for compact building reconstruction

4.1. Overview

The BSS segment and its approximation of shape surfaces drive our method design for large-scale compact building reconstruction, as shown in Fig. 3. The input is an oriented point cloud which can be of different scales and complexity, i.e., covering one single building or a high-density street block, and the output is a compact polygon mesh, which is composed of watertight 2D manifolds.

Our method adopts the hypothesizing strategy commonly used in the plane-based building reconstruction (Li et al., 2016; Nan and Wonka, 2017) and frames the pipeline in Fig. 3 into two stages. Stage 1 skeletonizes the point cloud into BSS atoms according to Def. 3.1. Then, the atoms are projected onto 2D vertical planes and coarsely segmented into candidate BSS segments. The corresponding polyhedra of the candidate BSS segments fit the given point cloud densely and tightly. To further reconstruct a compact mesh in Stage 2, an integer programming model is formulated and solved to merge the candidate BSS segments while maintaining a minimal geometric deviation. The final mesh is then generated as 2D manifolds by performing union operations on the volumes of BSS segments.

4.2. Stage 1: Generating BSS segment candidates

4.2.1. Skeletonization

BSS atoms are extracted by pairing non-horizontal points with normals. To streamline the proof of concept, the BSS method assumes that an input has only one pair of orthogonal horizontal orientations, and extracts the BSS atoms of one single primary orientation. Because orthogonal surfaces are

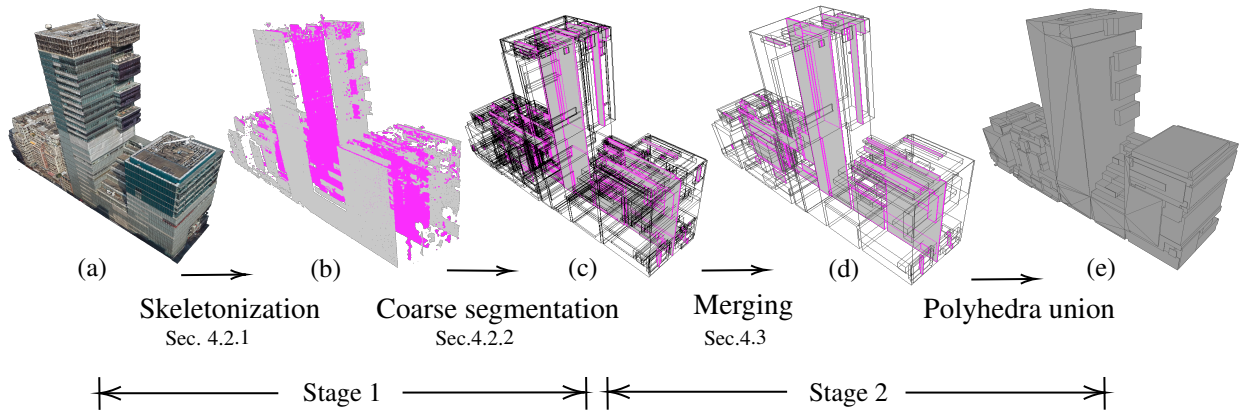


Figure 3: Overview of the building reconstruction based on BSS. (a) Input point cloud; (b) BSS atoms of the primary orientation; (c) and (d) Candidate and merged BSS segments with their corresponding polyhedra; and (e) Output compact mesh.

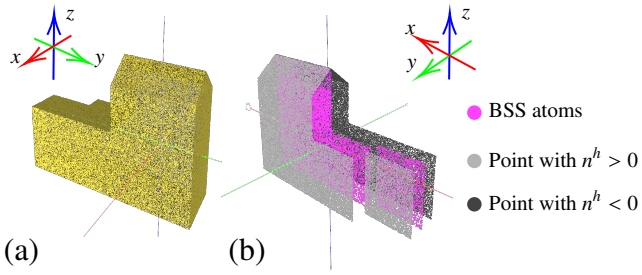


Figure 4: Skeletonization along the primary orientation. (a) Input point cloud; (b) Preserved points of the primary orientation (in light and dark gray) and the skeletonized BSS atoms (in magenta), where the primary orientation is aligned with the positive direction of y-axis.

covered with duplication by the induced shape boundary of orthogonal BSS segments, as illustrated in Rows 1, 3, and 4 of Table 1. Horizontal points are filtered out from the input point cloud based on the given threshold on the vertical angle of normals. Next, the primary orientation is voted as the most frequent horizontal orientations among the normals of the remaining points. Note that all the orientations are converted within the range $[-\pi/2, \pi/2)$ and the opposite orientations are equivalent in the voting. The method then preserves the points facing and opposite to the primary orientation only for the following skeletonization. The remaining points are then rotated to align the primary orientation to the positive direction of y-axis (Fig. 4(b)) and voxelized. After rotation, the horizontal components of normals $n^h \in [-1, 1]$ are real numbers rather than 2D vectors, which are also attached to each occupied voxel. For each occupied voxel with n^h, our method searches along a ray pointing to the opposite horizontal direction of n^h to find the nearest voxel with $-n^h$ and computes the locations and radii according to Def. 3.1.

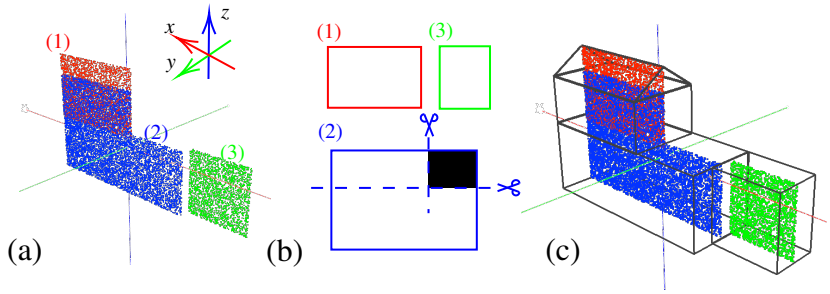


Figure 5: Illustration of coarse BSS segmentation taking the BSS atoms in Fig. 4 as input. (a) Three atom groups after binning, with labels (1) to (3); (b) Projected binary images of atom groups, and the binary image with label (2) is clipped into two rectangles; (c) Corresponding polyhedra of four BSS segment candidates.

4.2.2. Coarse BSS segmentation

BSS segment candidates are proposed by grouping the spatially connected BSS atoms with close parameters. Note that the shapes of BSS segments are not limited to rectangles in Def. 3.2. However, for the sake of simplicity, this paper assumes the BSS atom groups can be decomposed into a set of rectangles. As illustrated in Fig. 5 (a), we bin the BSS atoms based on their y-coordinates, normals' horizontal components n^h, radii r, and intersection elevations z_i with certain intervals, resulting in atom groups that resemble '3D sheets'. Atoms on each sheet are projected onto a vertical plane to generate a binary image as shown in Fig. 5 (b), followed by morphological operations to fill in the holes and smooth out the zigzag boundaries. Rectangles are segmented from the binary image by iteratively clipping along the rows or columns with maximal average gradients. For each rectangle, we then take the average n^h and r (for vertical planes) or n^h and z_i (for inclined planes) to form a BSS segment candidate and its corresponding polyhedron (Fig. 5 (c)). This step involves one major parameter, the distance interval of segmentation δ_{seg}, to control the LoD of reconstruction and one minor parameter, the angular interval θ_{seg}, to separate the BSS atoms with different inclined slopes.

4.3. Stage 2: Merging BSS segments

Stage 2 formulates an integer programming (IP) model to find out the optimum pairs of BSS segments and their polyhedra to merge. Given a set of BSS segment candidates $\mathcal{B} = \{b_i | 1 \leq i \leq N\}$ of size N, the Boolean variables of all the possible pairs are $\mathcal{M} = \{m_{ij} | 1 \leq i, j \leq N, i \neq j\}$, where m_{ij} denotes whether b_i is merged into b_j (i.e., $m_{ij} = 1$) or not (i.e., $m_{ij} = 0$). Inspired by Li et al. (2016); Nan and Wonka (2017), the objective function is composed of (1) a fitting term D to minimize the deviations and (2) a complexity term C to reduce the number of BSS segments after merging. The

formulation is:

$$\arg\min_{\mathcal{M}} \; D + \lambda_{com} C \tag{1}$$

$$s.t. \forall i, \sum_{1 \le j \le N, j \ne i} m_{ij} + m_{ji}/N \le 1, \tag{2}$$

where λ_{com} is the key weighting parameter that balances the two terms D and C. The constraint requires that each BSS segment b_i should be merged into at most one other BSS segment so that $\sum_{1 \le j \le N, j \ne i} m_{ij} \le 1$. Meanwhile, if b_i is merged to b_j, i.e. $\sum_{1 \le j \le N, j \ne i} m_{ij} = 1$, then other BSS segments cannot be merged into b_i so that $\sum_{1 \le j \le N, j \ne i} m_{ij}/N = 0$. Otherwise, i.e., $\sum_{1 \le j \le N, j \ne i} m_{ij} = 0$, there could be some other BSS segments merged into b_i, which means $0 \le \sum_{1 \le j \le N, j \ne i} m_{ij}/N < 1$. For smoothing out excess geometric details and noise, the removal of BSS segments can be integrated into the IP model by adding a null BSS segment as $b_\emptyset$ into $\mathcal{B}$. A BSS segment b_i is deleted if it is merged into $b_\emptyset$.

4.3.1. Formulation of the fitting and complexity terms

The fitting term decomposes the geometric deviations into the residual volume $\Delta V(i, j)$ caused by each merging operation:

$$D = \sum_{i=1}^{N} \sum_{j=1, j \ne i}^{N} m_{ij} \cdot |\Delta V(i, j)|/v. \tag{3}$$

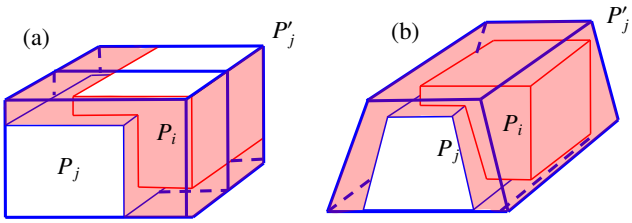


Figure 6: Illustration of merging BSS segments. (a) Updating cuboid P_j to merge P_i; (b) Updating trapezoidal prism P_j to merge P_i. The space of $\Delta V(i, j)$ is filled in red.

For each merging operation $m_{ij} = 1$, the polygon G and parameters of b_j should be updated so that its polyhedron can be enlarged exactly to cover all the vertices of the original polyhedra P_i and P_j, as shown in Fig. 6(a) and (b). However, the enlarged P'_j also causes some geometric deviations which are quantified as the residual volume $\Delta V(i, j)$ (filled in red in Fig. 6). To compute the $\Delta V(i, j)$, a union operation is performed on all the corresponding polyhedra of the candidate BSS segments, resulting in overall polyhedra denoted as P_{all}. The $\Delta V(i, j)$ is then computed as the volume of $P'_j - P_{all}$. To normalize the fitting term, the total residual volume is then divided by the volume of P_{all}, i.e., v in Eqn. (3).

One key dimensional parameter in this step is the threshold of truncation δ_{trun}. Only BSS segments close to b_i can be merged into b_i. An extremely large value is assigned to $\Delta V(i, j)$ if the distance

between P_i and P_j is greater than δ_{trun}. Besides, as only a fraction of P_{all} is involved in the difference operation of $P'_j - P_{all}$, we also approximate this polyhedral difference as $P'_j - n(P_j)$, where $n(P_j)$ only consists of P_j itself and other polyhedra whose distances to P_j are less than the threshold of truncation δ_{trun}.

The complexity term counts the amount of remaining BSS segments after merging and encourages more merging operations. The formulation is as follows:

$$\sum_{1\le i\le N} keep(i)/N \tag{4}$$

$$keep(i) = \begin{cases} 1, & \sum_{1\le j\le N, j\neq i} m_{ij} = 0 \\ 0, & otherwise, \end{cases} \tag{5}$$

where $keep(i) = 1$ indicates the i-th BSS segment candidate b_i is not merged. Note that in an optimal solution $\mathcal{M}$ of Eqn. (1), there can be multiple candidates to merged with one target candidate b_j, i.e., $\sum_{1\le i\le N, i\neq j} m_{ij} > 1$. The post-merging polyhedron P'_j should exactly cover P_j and all the other candidates' polyhedra.

5. Experiments

5.1. Test data

We tested the BSS method on 15 point clouds from a variety of datasets collected by ground-borne LiDAR scanning and air-borne photogrammetry. Five scenes exhibit *low-rise, low-density* buildings with inclined roof structures, as shown in Samples (1) to (5) of Fig. 7. The first one is in-house data while the remaining four are from two open datasets, Hessigheim3D (Kölle et al., 2021) and Open Heritage 3D (CHEI, 2023). The other 10 point clouds, Samples (6) to (15) in Fig. 7, are *high-rise, high-density* and Manhattan-world scenes of Hong Kong from the public 3D visualization maps released by HKSAR government (LandsD, 2022). In total, there are more than 60 buildings in the 15 test samples. Despite the original data being of different forms, i.e., triangular mesh and point clouds, and in different densities, we resampled all the scenes into point clouds with a point distance at 0.1 m. The size of resampled point clouds and the number of buildings, i.e., #pt and #b in Fig. 7, reflect the reconstruction complexity of each sample.

5.2. Implementation details

Our BSS method of reconstruction was implemented in C++ and Python. The coarse BSS segmentation (Sect. 4.2.2) is implemented in Python. The other steps were written in C++. We solved the IP model of BSS segment merging (Sect. 4.3) by *Gurobi* (ver. 10.0), and performed the union and

difference operations on polyhedra through the 3D boolean operations in Package Nef_Polyhedra of CGAL (ver. 5.5). Besides, paralleling computing was integrated in the skeletonization and merging. All experiments were conducted on a workstation with an Intel Core i7 CPU (2.9 GHz, 8 cores) and 128 GB of RAM.

The three key parameters of the BSS method were set up according to the two groups of scenes. For *high-rise, high-density* scenes, we set δ_{seg}=2 m, δ_{trun}=1 m, and λ_{com}=1. For *low-rise, low-density* scenes, the parameters were δ_{seg}=2 m, δ_{trun}=0.25 m, and λ_{com}=0.125, where the values δ_{trun} and λ_{com} were smaller for reflecting the smaller dimensions and lower complexity, respectively. The remaining two parameters were consistent for all experiments. The voxel size of the skeletonization was 0.25 m, which should not be less than δ_{seg}; the angular interval θ_{seg} was set to $\pi/9$, while modifying this parameter within the range of $[\pi/18, \pi/6]$ did not noticeably affect the reconstruction.

5.3. Results

Fig. 7 shows the results of the BSS method on all the 15 samples. Overall, the BSS method demonstrated its effectiveness in reconstructing both simple and complex scenes, capturing essential structures with relatively small deviations, and providing compact meshes within reasonable processing time. The results can be further elaborated in terms of three complexity levels as follows.

For the *low-rise, low-density* buildings shown in Fig. 7(1) to (5), the reconstruction of the BSS method captured the essential structures, including the symmetric roofs and parallel facades, of all the five scenes into tens of faces with a small RMSD from the input point clouds to the reconstructed meshes ranging from 0.4 to 0.95 m. The BSS method only took around 2-8 seconds to reconstruct these simple *low-rise, low-density* scenes.

Meanwhile, the BSS method also reconstructed high-rise buildings with complex structures faithfully, as shown in Fig. 7(6), (7), (8), (12), and (15). Although there are only one or two buildings in each scene, the structures of buildings are complex, which could be decomposed into multiple components, including projections on the roofs and facades. The BSS method robustly reconstruct all the main components of each building, along with the major projections, achieving an RMSD from 0.6 to 1.5 m within 0.5 to 2.5 minutes. Meanwhile, the results of these buildings were compact, using 100 to 820 triangles to approximate the entire scene.

Furthermore, in the case of *high-rise, high-density* scenes with a few to a dozen buildings, as depicted in Fig. 7(9), (10), (11), (13), and (14), the BSS method consistently outlined all the buildings and their significant projections, even when confronted with highly complex and noisy input data. The reconstruction of these scenarios compressed the entire scenes from millions of points into several hundreds to over a thousand faces, simultaneously achieving an RMSD ranging from 0.9 to 1.1 m,

with a computational time ranging from approximately 1.5 to 7 minutes.

5.4. Comparisons

We compared the BSS method with other four baseline building reconstruction methods, i.e., DualCont (Zhou and Neumann, 2010), ManBox (Li et al., 2016), PolyFit (Nan and Wonka, 2017), and KSR (Bauchet and Lafarge, 2020). The overall evaluation on the 15 test samples was plotted in Fig. 8. Detailed metrics (Table A.1) and visualization of the comparisons (Figs. A.1 to A.6) are attached in Appendix. Note that both the number of triangles and planar regions are reported to assess the compactness. The BSS method achieved the fair balance between accuracy and compactness and reconstructed all the 15 samples. DualCont was also validated robust as achieving 100% success rate and a small overall deviation. But the numbers of faces reconstructed by DualCont were the second largest. In contrast, KSR and PolyFit presented an effective balance between accuracy and compactness. Their scatters were the two closest to the BSS method's. Besides, as confined to Manhattan-World assumption, ManBox were excluded in the test of five *low-rise, low-density* scenes with inclined roof planes. Despite targeting at compact reconstruction, ManBox were unstable in reconstructing highly complex scenes, resulting in the most faces and largest deviations in our experiments.

Fig. 9 highlights four samples in the comparison to showcase the reconstruction details of the five methods for understanding the overall evaluation summarized above. As shown in the second column of Fig. 9, DualCont reconstructed all the four samples into very dense triangles. Though accurate, the surfaces looks rough. ManBox eliminated this roughness by approximating buildings with planar primitives, as shown in the third column of Fig. 9. Yet, its accuracy and generality were both declined as it cannot reconstruct inclined roof planes and missed some major structures in Samples (14) and (15). PolyFit and KSR achieved much better compactness and accuracy yet still encountered some reconstruction issues as their input planes detected by RANSAC could be incomplete or noisy. Fig. A.7 shows the inputs of PolyFit and KSR on these four samples. The reconstruction defects caused by the missing plane issue are highlighted by the red circles in Samples (3) and (11). Besides, PolyFit was sensitive to the missing or noisy planes near the groud. Fig. 9 shows the missing bottom parts in its results of Samples (14) and (15) to illustrate this issue. Note that the missing planes can be reduced by increasing the minimum support point number and distance tolerance in RANSAC plane detection. However, the number of input planes will subsequently grow significantly, resulting in much longer processing time and undermining compactness in the reconstructed meshes.

In contrast, the BSS method presented superior robustness against missing and noisy planes. Because the BSS method only relies on a pair of parallel or symmetric planes to reconstruct one volume

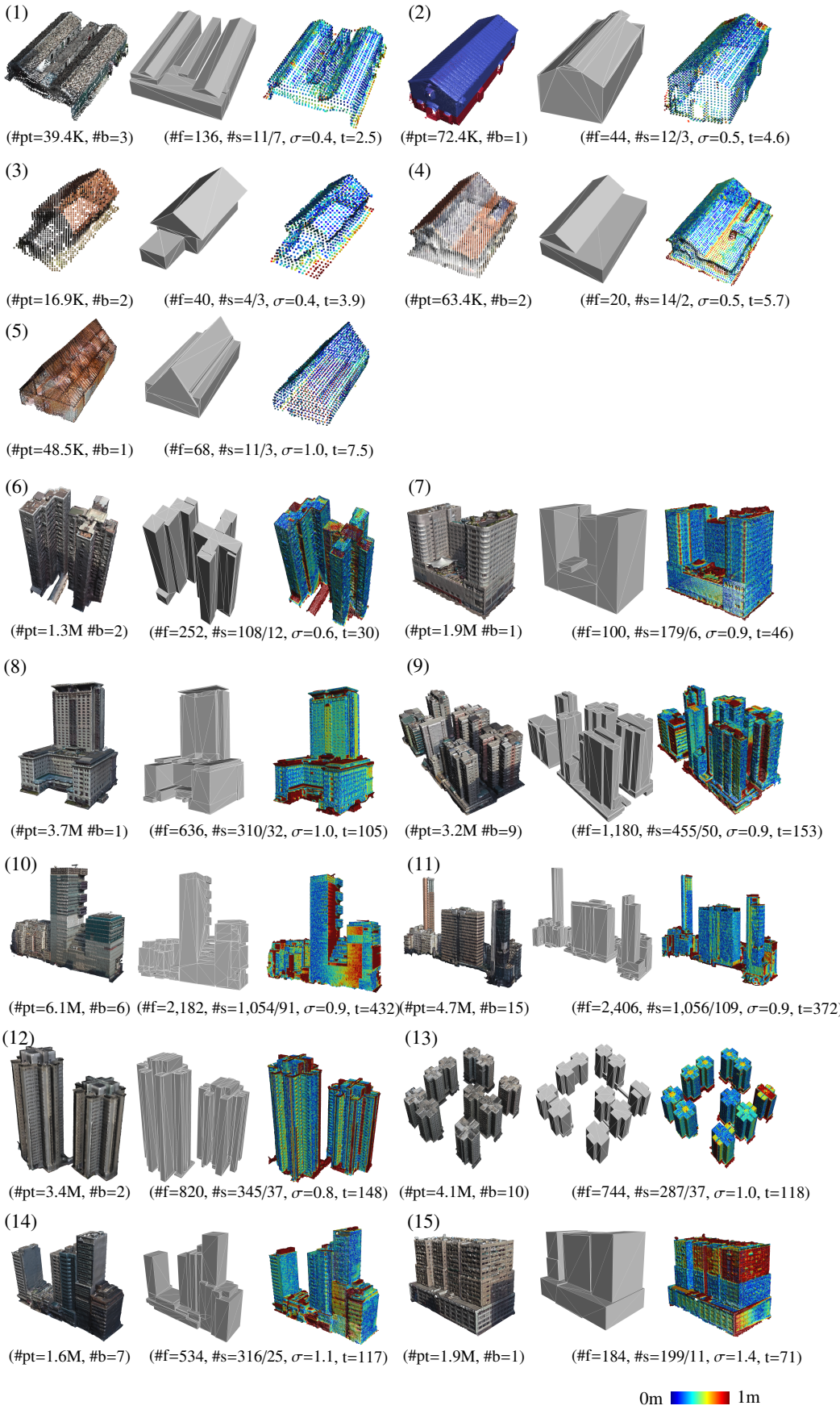


Figure 7: Results of the BSS methods on the 15 test samples containing more than 60 buildings in total. Left to right: input point cloud, output mesh, and geometric deviation from the input to the output. #pt, #b, #f, #s, σ, and t refer to the number of input points, the number of input buildings, the number of output faces (triangles), the number of output BSS segments before / after merging, RMSD (m), and processing time (sec.), respectively.

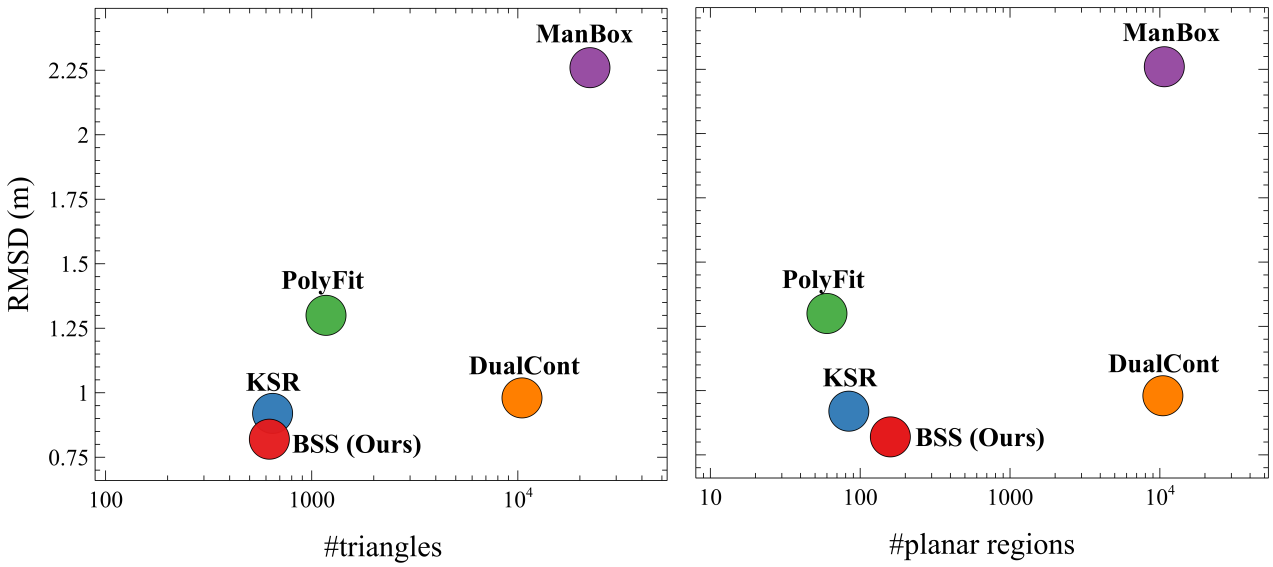


Figure 8: Comparisons between the BSS and baseline methods in terms of accuracy and compactness.

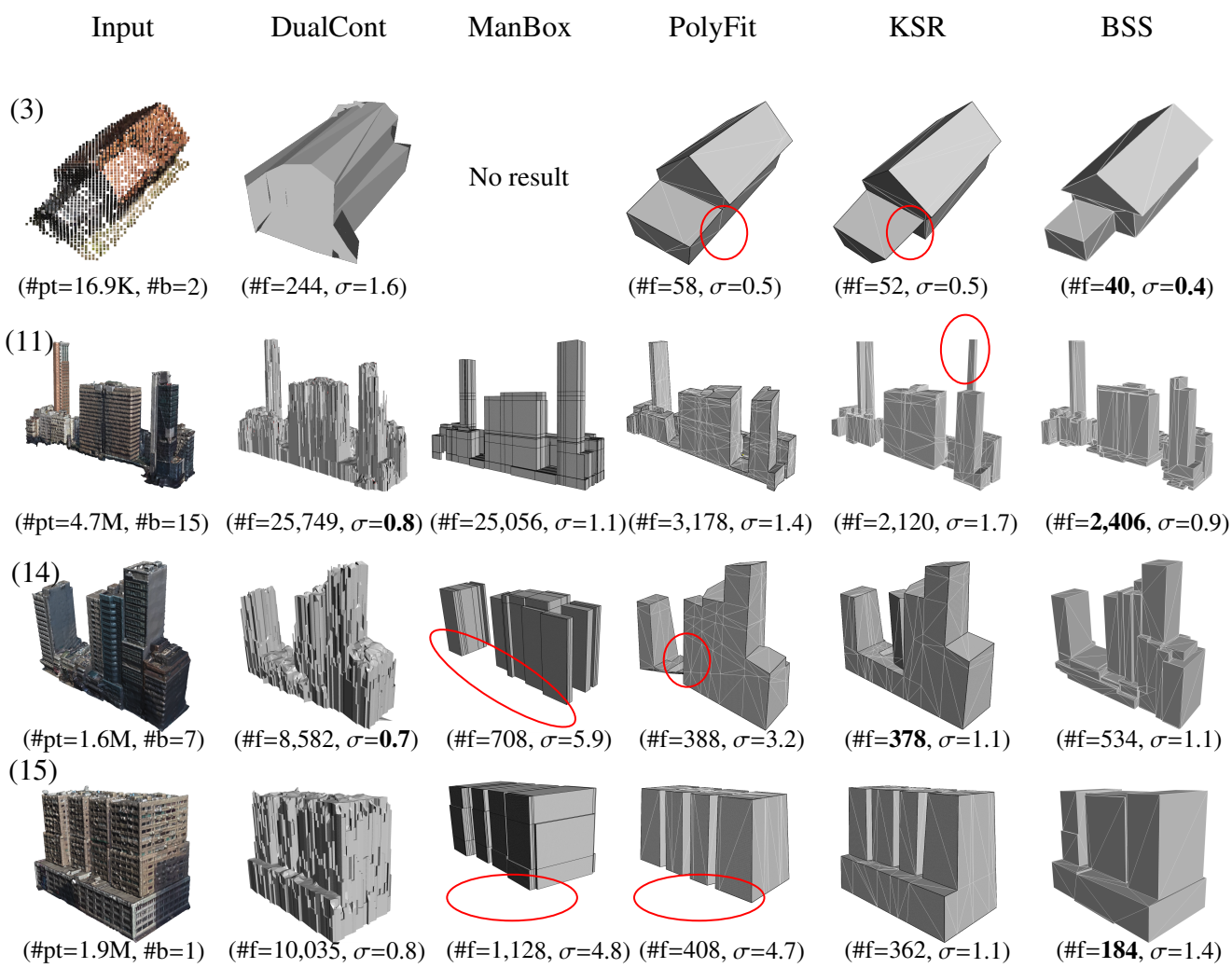


Figure 9: Comparison between the BSS and four baseline methods on Samples (3), (11), (14), and (15). #pt, #b, #f, and σ refer to the number of input points, the number of input buildings, the number of output faces (triangles), and RMSD (m), respectively. Detailed input planes to PolyFit and KSR are also shown in Fig. A.7.

of building interior. Missing planes that are not orienting to the primary orientation would not affect the reconstruction. Besides, different from the well-established mechanism of candidate selection by ManBox, PolyFit, and KSR, the polyhedra of BSS segment candidates already outline the building interior and the following merging is designed to reduce the number of BSS segments without removing the volumes supposed to be in the building interior. To sum up, the comparisons demonstrated the enhanced robustness contributed by the global structures integrated in the BSS method of compact reconstruction, despite room for further improve discussed in Sect. 6.2.

5.5. *Parameter analysis*

Fig. 10 shows the sensitivity tests of BSS parameters defined in Sect. 4.2.2 and Sect. 4.3 against the three performance metrics. The three parameters largely regulate BSS method's reconstructions. The first two parameters δ_{seg} and δ_{trun} were correlated and tested together, while the λ_{com} is relatively independent. We tuned the parameters in an exponential order around the given parameter settings.

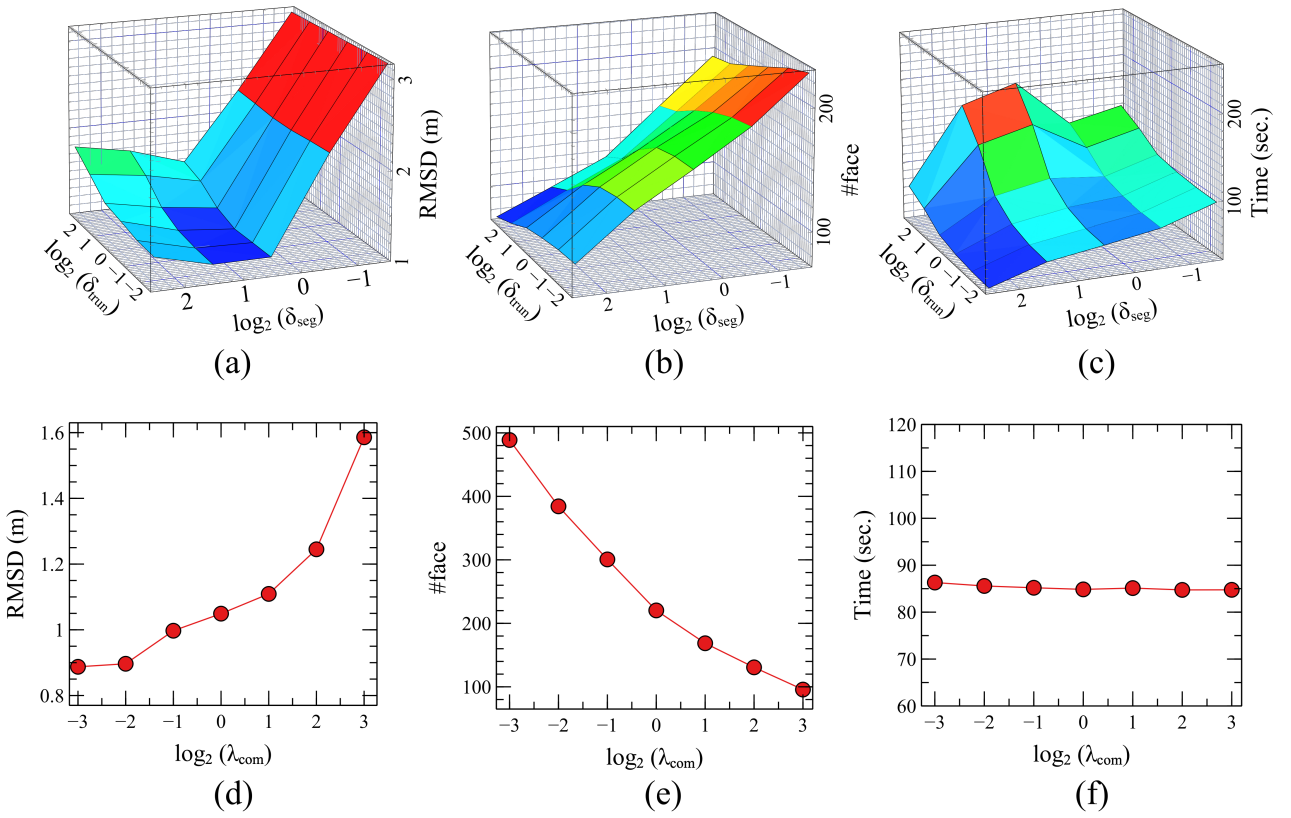


Figure 10: Parameter analysis against the segmentation interval δ_{seg} of coarse BSS segmentation as well as the threshold of truncation δ_{trun} and λ_{com} of BSS segment merging. λ_{com} was set as 1 for the experiments of (a) to (c), while δ_{seg} and δ_{trun} were set as 2 m and 0.5 m for (d) to (f), respectively.

Segmentation interval δ_{seg} in coarse BSS segmentation. The δ_{seg} was tuned as 0.5, 1, 2, and 4 m. Note that δ_{seg} should be greater than the voxel size 0.25 m. A δ_{seg} larger than 4 m will result in over-smoothing. As shown in the first row of Fig. 10, with a larger δ_{seg}, the number of faces is smaller

while the RMSD increases. Meanwhile, the reconstruction time grows if the coarse segmentation produces more candidate BSS segments. We observed that there were sufficient candidate set of BSS segments with a lower RMSD when δ_{seg}=2 m, as shown in Fig. 11. The test results revealed that candidate BSS segments captured with δ_{seg}=2 m achieved appropriate coverage for the test building scenes. δ_{seg} can be set to smaller values for denser and cleaner point clouds. For example, the result of σ_n=0 m with a density of 100 pts/m^2 in Fig. 13 is reconstructed with δ_{seg}=0.5 m.

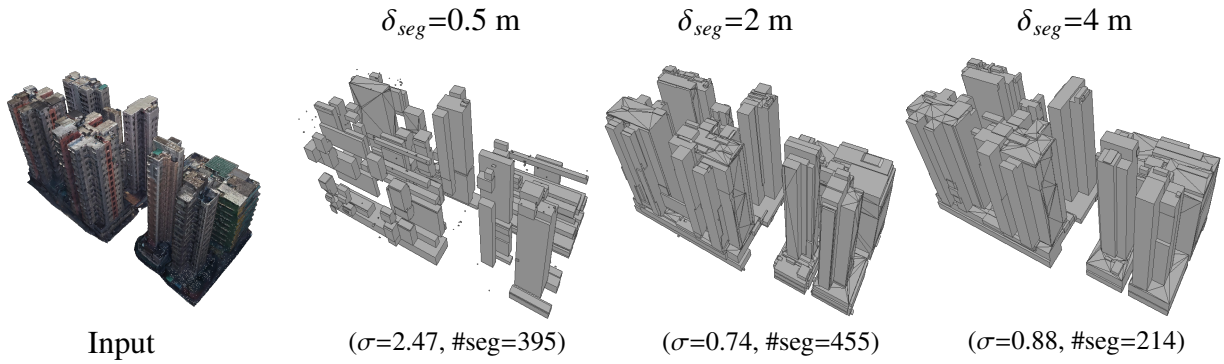


Figure 11: Corresponding polyhedra produced by coarse BSS segmentation with different segmentation intervals (δ_{seg}). σ refers to the RMSD (m) from input point clouds to reconstructed surfaces after polyhedra union. #seg is the number of candidate BSS segments.

Threshold of truncation δ_{trun} in merging BSS segments. The δ_{trun} was tuned as 0.25, 0.5, 1, 2, and 4 m. As shown in Fig. 10(c), a larger δ_{trun} led to longer processing time, which was especially long when the coarse segmentation also produced more candidate BSS segments. Meanwhile, a larger δ_{trun} can also slightly reduce the number of reconstructed faces.

Weighting factor λ_{com} in merging BSS segments. We tested the factor λ_{com} of Eqn. (1) as 0.125, 0.25, 0.5, 1, 2, 4, and 8. It impacted accuracy and compactness mainly, as shown in Fig. 10(d) to (f) where δ_{seg}=2 m and δ_{trun}=0.5 m. Corresponding examples are shown in Fig. 12. With a smaller λ_{com}, the BSS segment merging encourages a higher fitting accuracy and tends to merge fewer candidate BSS segments and keep more faces. With a larger λ_{com}, the merging smoothed more details and preserved only large polyhedra representing the main building structures of the input, as shown in the last column in Fig. 12. Besides, the time cost did not change significantly as shown in Fig. 10(f).

5.6. Robustness against noise and density

Fig. 13 shows the robustness of the BSS method against increasing noise levels on synthetic point clouds. As the level of Gaussian noise (standard deviation, *std*) increases, the geometric deviations between the reconstructed meshes and the input point clouds also increase. This effect is particularly pronounced on the faces that are orthogonal to the primary orientations, as shown in σ_n=0.1 and

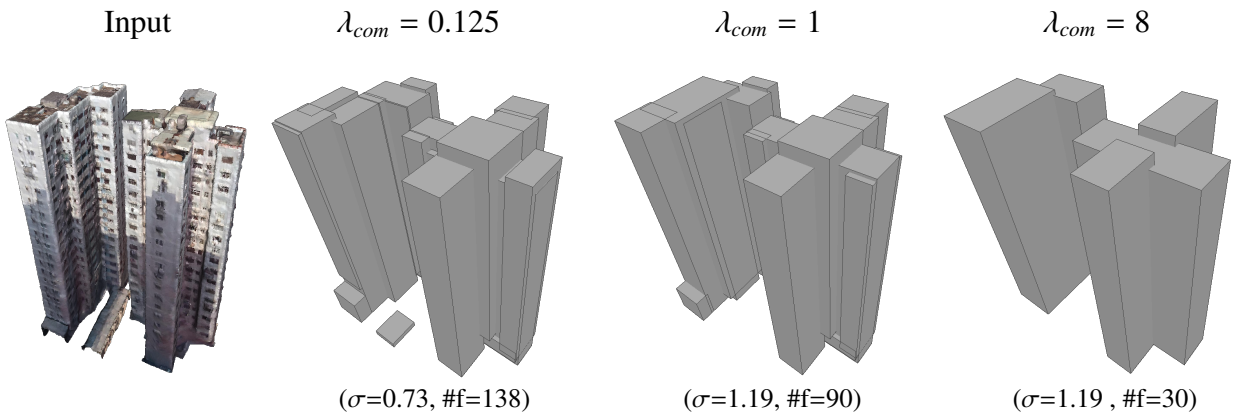


Figure 12: Reconstructed meshes with different complexity factors λ_{com} of merging BSS segments. σ refers to the RMSD (m) from input point clouds to reconstructed meshes, while #f is the number of faces.

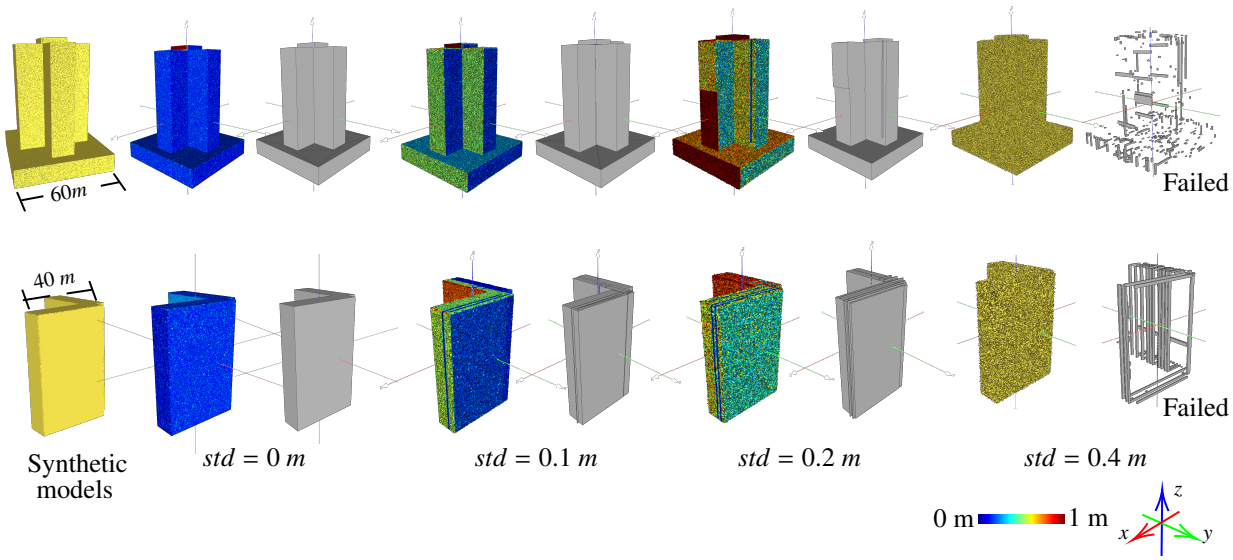


Figure 13: Results of synthetic point clouds with increasing levels of Gaussian noises from *std*=0 m to *std*=0.4 m. Primary orientations are facing to the positive directions of y-axes (shown in green).

0.2 m of Fig. 13. When σ_n = 0.4 m, the skeletonized BSS atoms were noisy and the BSS segment candidates failed to adequately cover the entire interior of buildings.

Fig. 14 displays the reconstruction of the BSS method on two real-world noisy samples where there are unevenly distributed noises. The BSS atoms of small structures or around missing parts could not form reasonable BSS segments, resulting in the final meshes disregarding or overestimating such small structures and incomplete scans.

Fig 15 showcases the reconstruction of the BSS method against decreasing density levels from 50 to 0.1 pts/m^2. With a density no less than 1 pts/m^2, the BSS method can faithfully reconstruct the building. When the density decreases to 0.5 pts/m^2, there could be missing parts and larger geometric deviations in the reconstructed mesh. And if the density is as low as 0.1 pts/m^2, the reconstruction failed due to the insufficient BSS atoms, making it impossible to propose reasonable BSS segment

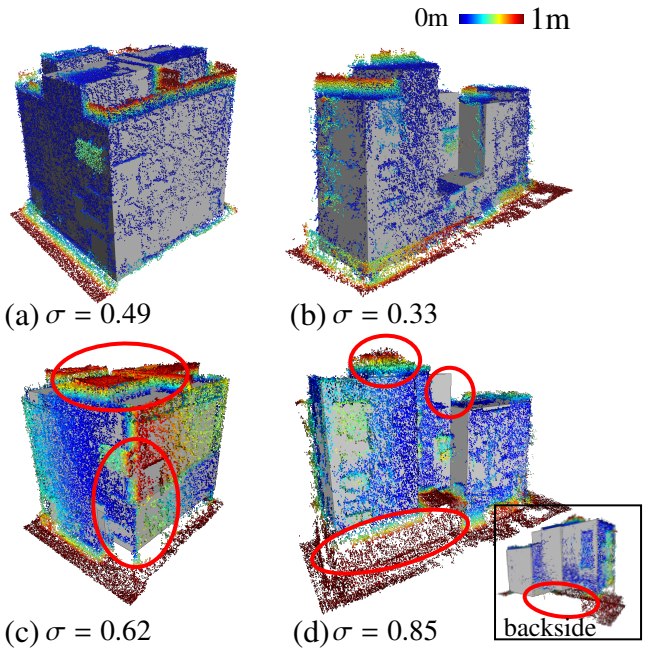


Figure 14: Missing parts and overestimation in the reconstruction of two noisy samples from Li et al. (2016) and Nan and Wonka (2017). (a) and (b) are results of PolyFit; (c) and (d) are results of BSS. σ refers to RMSD (m).

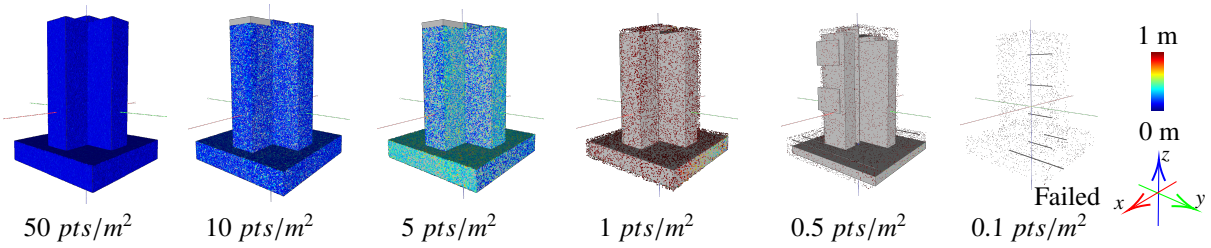


Figure 15: Results of synthetic point clouds with increasing densities from 50 to 0.1 pts/m^2. Primary orientations are facing to the positive directions of y-axes (shown in green).

candidates.

Moreover, in Fig. 13 and Fig. 15, the axes of coordinate systems are drawn to demonstrate the robustness of the primary orientation estimation in Sect. 4.2.1. Even at a high noise level (σ_n = 0.4 m) or low point cloud density (0.1 pts/m^2), the primary orientations were still estimated consistently.

5.7. Performance Analysis

Fig. 16(a) shows the total computational time and the duration of three steps, i.e., skeletonization, coarse BSS segmentation, and BSS segment merging. On average, the three steps took 12%, 6%, and 82% of the total time, respectively. The processing time for both skeletonization and coarse segmentation increases linearly with the number of input points. The time for merging presents a quadratic growth and so does the total time. Despite the quadratic growth, the BSS method can stably reconstruct compact building models from millions of input points in several minutes in our test environment. Fig. 16(b) shows that the BSS method and KSR took a comparable amount of time

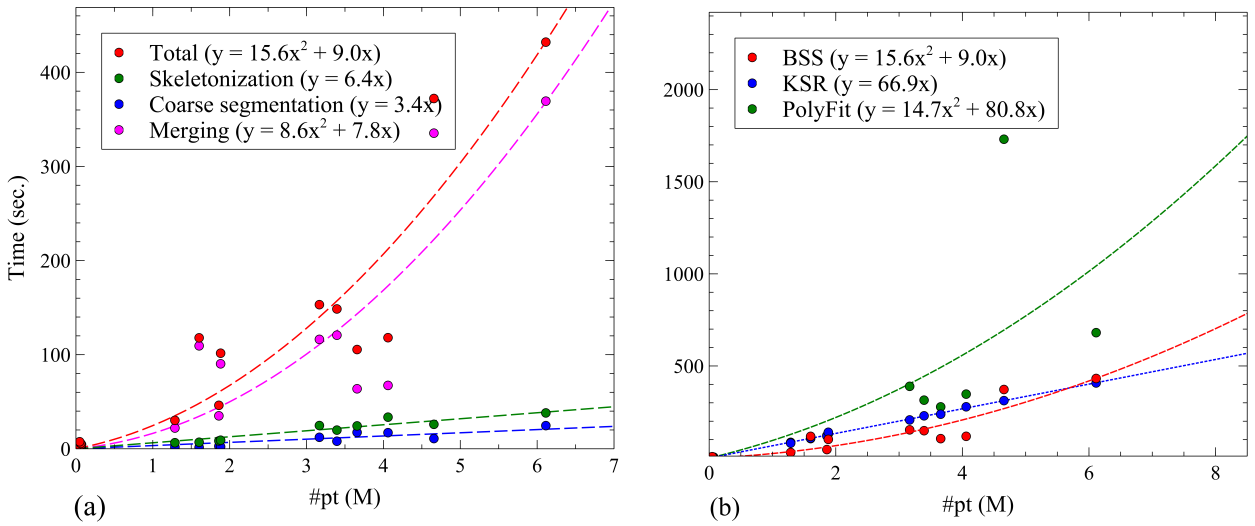


Figure 16: Processing time of the BSS and baseline methods, where #pt (M) refers to the number of input points in million. (a) Time costs of the BSS method's overall procedure and three crucial steps, skeletonization, coarse segmentation, and merging; (b) Comparison between the PolyFit, KSR, and BSS methods.

to reconstruct the 15 test samples. Yet, the projection indicates a linear growth of KSR which exhibits higher scalability for inputs containing a larger number of points. Note that both KSR and PolyFit require planes detected by RANSAC as inputs, whereas the BSS method takes oriented point clouds only as inputs. Therefore, the BSS method relies on its own to detect the planar BSS atom groups, which contributes to the processing time.

6. Discussion

6.1. Implications

The BSS defined in this paper is an expressive 3D feature reflecting architectural designs in high-density cities. BSS atoms accurately represent the 3D point pairs, including the pairs of noisy points. The BSS segment and associated polyhedra cancel the noisy BSS atoms and collectively mitigate the random errors along the normal direction in the aggregation from atom level to segment level. Furthermore, BSS graphs or feature vectors can open new avenues to volumetric city modeling and other urban remote sensing and photogrammetry studies, such as unsupervised urban pattern clustering, relational query using BSS map of a city block (Xue et al., 2021a), and semantic enrichment of city models using BSS-enhanced podiums or balcony (Xue et al., 2021b).

For practitioners in smart city development, the BSS method of reconstruction offers a robust, accurate, and efficient approach to compact building and city modeling. To demonstrate the ability of reconstructing larger-scale and more general urban scenes, we tested the BSS method on two areas of UrbanBIS (Yang et al., 2023), i.e., *Qingdao* (2.31 km^2) and *Yingrenshi* (0.08 km^2). As shown in

Fig. 17, the BSS method reconstructed most of the building points, i.e., 242.0M/269.6M points (89.8 %) of *Qingdao* and 14.6M/15.0M (97.3 %) points of *Yingrenshi*, into only 57.9K and 2.9K faces with average RMSD as 1.39 m and 1.12 m, respectively. The BSS-generated compact building models can further facilitate smart and sustainable city applications. Examples include automatic window view content and depth assessment (Li et al., 2022b), BSS building volumes for urban material stocks (Yuan et al., 2023), and digital urban governance (Deng et al., 2021).

6.2. Limitations

Nevertheless, given the assumption that buildings are composed of parallel vertical or symmetric inclined planes, the BSS method cannot reconstruct buildings with highly curved surfaces (the closeup of Fig. 17(f)) and shapes that lack both parallel vertical and symmetric inclined planes (Fig. 18(a)). Meanwhile, the following limitations of the proposed BSS theory and the reconstruction method should be addressed in future studies:

1. Single-orientation BSS segments. By assuming the orthogonality of the input data, we only extract the BSS atoms facing to the primary orientation of inputs. Therefore, the current reconstruction based on BSS segments is constrained to buildings composed of cuboid shapes and equally slanted roofs. In our future work, we will explore the extraction of multiple-orientation BSS atoms and segments, as demonstrated in Rows 2 and 5 of Table 1. This will allow generalization to symmetric buildings with non-orthogonal surfaces, which could further eliminate the zigzag shapes in the reconstruction of slightly curved buildings (the closeup in Fig. 17(e)). Meanwhile, multi-orientation BSS segments will enhance the robustness in reconstructing buildings with parallel vertical faces that deviate from the single primary orientation (Fig. 18(b)) and buildings whose primary-oriented faces are missing (the closeup in Fig. 17(b)).
2. No explicit relations between BSS segments and their polyhedra. There are different types of topological relations between BSS segments and their polyhedra, such as overlapping, covering, and touching. These relations reveal the arrangement of BSS segments and imply rich geometric constraints of the polygons and parameters of different BSS segments, which include the face adjacency prevalent in the previous building reconstruction. In our future work, we would like to formally define the arrangement of BSS segments and utilize the topology between BSS segments to facilitate building reconstruction. This will eliminate the deviation heterogeneity of faces with different orientations (Fig. 13) and fix the unwanted gaps between BSS segments (Fig. 19) that caused a larger number of planar regions as reported in Fig. 8(b) and Table A.1.

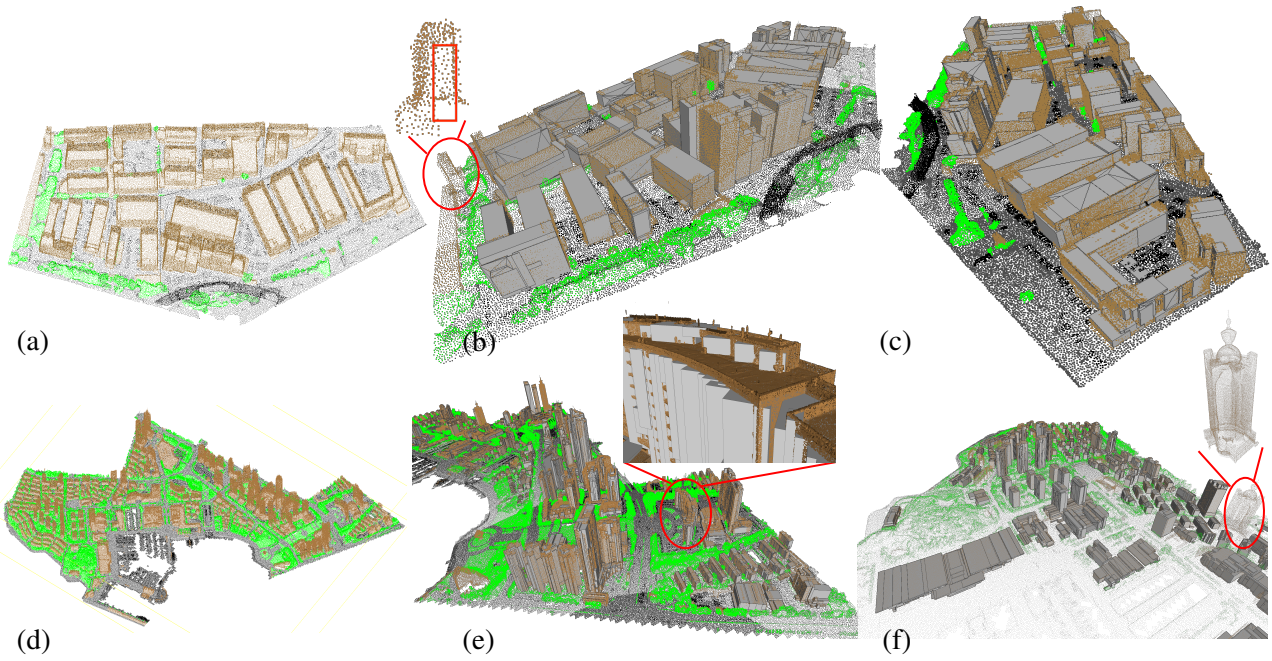


Figure 17: Large-scale reconstruction of *Yingrenshi* (0.08 km^2) and *Qingdao* (2.31 km^2) in UrbanBIS dataset. (a) and (d) Input point clouds of *Yingrenshi* and *Qingdao*, respectively. The points of buildings are colorized as brown. (b), (c), (e), and (f) are the side views of reconstructed results overlapped with the input point clouds. The closeup of (b), (e), and (f) highlights the limitations of the BSS methods in handling incompleteness, moderate curved surfaces, and significant curved surfaces, respectively.

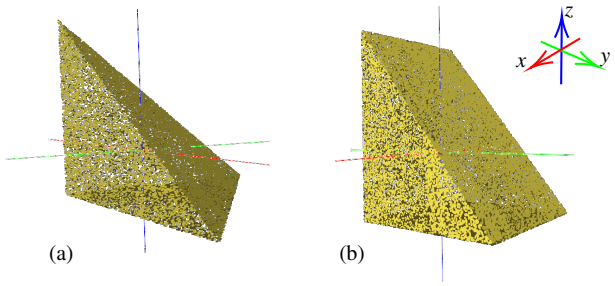


Figure 18: Shapes that BSS method failed to reconstruct.

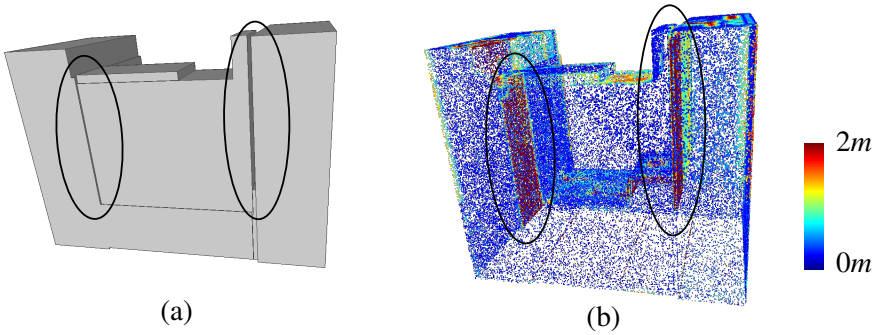


Figure 19: Examples of the gaps between reconstructed surfaces.

7. Conclusion

This paper defines a Building Section Skeleton (BSS) by revamping traditional shape skeletons to describe common architectural designs of buildings using urban point clouds. Based on BSS, a compact building reconstruction method is proposed as a two-stage pipeline. The first stage skeletonizes the input point cloud and proposes a set of candidate BSS segments; The second stage merges the candidate BSS segments via an integer programming model to minimize the geometric deviations and complexity of reconstruction. Experimental results and comparisons conducted on both *low-rise, low-density* and *high-rise, high-density* scenes confirmed that the BSS method for building reconstruction is compact, accurate, and efficient.

The proposed BSS is novel and expressive for processing urban point clouds, and the BSS method of reconstruction has great potential for studies and applications in the fields of city modeling and other urban remote sensing and photogrammetry. Nevertheless, we also identified some limitations of the current BSS theory and BSS method of reconstruction. Recommended future research directions include (1) developing multiple-orientation BSS method for reconstructing more general buildings and (2) defining explicit relations between BSS segments for higher reconstruction quality and inducing BSS graphs.

Acknowledgments

The work described in this paper was supported by the Hong Kong Research Grants Council [grant numbers 27200520 and T22-504/21-R] and the Natural Science Foundation of Guangdong, China [grant number 2023A1515010757]. Besides, we would like to express gratitude to the authors of PolyFit (Nan and Wonka, 2017) and MeshPlg (Bouzas et al., 2020), in particular Prof. Liangliang Nan at TU Delft, for helping us replicate their works in our experiments.

References

Acuto, M., 2016. Give cities a seat at the top table. Nature 537, 611–613. doi:10.1038/537611a.

Amenta, N., Bern, M., 1998. Surface reconstruction by Voronoi filtering, in: Proceedings of the Fourteenth Annual Symposium on Computational geometry, pp. 39–48. doi:10.1145/276884.276889.

Arcelli, C., Di Baja, G.S., Serino, L., 2010. Distance-driven skeletonization in voxel images. IEEE Transactions on Pattern Analysis and Machine Intelligence 33, 709–720. doi:10.1109/TPAMI.2010.140.

Attali, D., Boissonnat, J.D., Edelsbrunner, H., 2009. Stability and computation of medial axes: a state-of-the-art report, in: Mathematical foundations of scientific visualization, computer graphics, and massive data exploration. Springer, pp. 109–125. doi:10.1007/b106657_6.

Batty, M., 2013. The New Science of Cities. MIT press.

Bauchet, J.P., Lafarge, F., 2020. Kinetic shape reconstruction. ACM Transactions on Graphics 39. doi:10.1145/3376918.

Berger, M., Tagliasacchi, A., Seversky, L.M., Alliez, P., Guennebaud, G., Levine, J.A., Sharf, A., Silva, C.T., 2017. A survey of surface reconstruction from point clouds. Computer Graphics Forum 36, 301–329. doi:10.1111/cgf.12802.

Bisheng, Y., Fuxun, L., Huang, R., 2017. Progress, challenges and perspectives of 3D lidar point cloud processing. Acta Geodaetica et Cartographica Sinica 46, 1509–1516. In Chinese, doi:10.11947/j.AGCS.2017.20170351.

Blum, H., 1967. A transformation for extracting new descriptions of shape, in: Models for the perception of speech and visual form. MIT press, pp. 362–380.

Bouzas, V., Ledoux, H., Nan, L., 2020. Structure-aware building mesh polygonization. ISPRS Journal of Photogrammetry and Remote Sensing 167, 432–442. doi:10.1016/j.isprsjprs.2020.07.010.

Bremer, M., Rutzinger, M., Wichmann, V., 2013. Derivation of tree skeletons and error assessment using LiDAR point cloud data of varying quality. ISPRS Journal of Photogrammetry and Remote Sensing 80, 39–50. doi:10.1016/j.isprsjprs.2013.03.003.

Bucksch, A., Lindenbergh, R., 2008. CAMPINO—A skeletonization method for point cloud processing. ISPRS journal of photogrammetry and remote sensing 63, 115–127. doi:10.1016/j.isprsjprs.2007.10.004.

CHEI, Q.I., 2023. Open Heritage 3D. https://openheritage3d.org/. Accessed: 2023-5-31.

Chen, Z., Ledoux, H., Khademi, S., Nan, L., 2022. Reconstructing compact building models from point clouds using deep implicit fields. ISPRS Journal of Photogrammetry and Remote Sensing 194, 58–73. doi:10.1016/j.isprsjprs.2022.09.017.

Ching, F.D., 2023. Architecture: Form, space, and order (5th ed.). John Wiley & Sons.

Colomina, I., Molina, P., 2014. Unmanned aerial systems for photogrammetry and remote sensing: A review. ISPRS Journal of photogrammetry and remote sensing 92, 79–97. doi:10.1016/j.isprsjprs.2014.02.013.

Deng, T., Zhang, K., Shen, Z.J.M., 2021. A systematic review of a digital twin city: A new pattern of urban governance toward smart cities. Journal of Management Science and Engineering 6, 125–134. doi:10.1016/j.jmse.2021.03.003.

Dey, T.K., Zhao, W., 2002. Approximate medial axis as a voronoi subcomplex, in: Proceedings of the Seventh ACM Symposium on Solid Modeling and Applications, pp. 356–366. doi:10.1145/566282.566333.

Dou, Z., Lin, C., Xu, R., Yang, L., Xin, S., Komura, T., Wang, W., 2022. Coverage Axis: Inner Point Selection for 3D Shape Skeletonization. Computer Graphics Forum 41, 419–432. doi:10.1111/cgf.14484.

Du, J., Chen, D., Wang, R., Peethambaran, J., Mathiopoulos, P.T., Xie, L., Yun, T., 2019. A novel framework for 2.5-D building contouring from large-scale residential scenes. IEEE Transactions on Geoscience and Remote Sensing 57, 4121–4145. doi:10.1109/TGRS.2019.2901539.

Gatzke, T.D., Grimm, C.M., 2006. Estimating curvature on triangular meshes. International Journal of Shape Modeling 12, 1–28. doi:10.1142/S0218654306000810.

Giblin, P.J., Brassett, S., 1985. Local symmetry of plane curves. The American Mathematical Monthly 92, 689–707. doi:10.1080/00029890.1985.11971717.

Haala, N., Kada, M., 2010. An update on automatic 3D building reconstruction. ISPRS Journal of Photogrammetry and Remote Sensing 65, 570–580. doi:10.1016/j.isprsjprs.2010.09.006.

He, Z., Wang, Y., Cheng, Z., 2021. Manhattan-world urban building reconstruction by fitting cubes. Computer Graphics Forum 40, 289–300. doi:10.1111/cgf.14421.

Henn, A., Gröger, G., Stroh, V., Plümer, L., 2013. Model driven reconstruction of roofs from sparse LiDAR point clouds. ISPRS Journal of Photogrammetry and Remote Sensing 76, 17–29. doi:10.1016/j.isprsjprs.2012.11.004.

Jalba, A.C., Kustra, J., Telea, A.C., 2012. Surface and curve skeletonization of large 3D models on the GPU. IEEE Transactions on Pattern Analysis and Machine Intelligence 35, 1495–1508. doi:10.1109/TPAMI.2012.212.

Jalba, A.C., Sobiecki, A., Telea, A.C., 2015. An unified multiscale framework for planar, surface, and curve skeletonization. IEEE Transactions on Pattern Analysis and Machine Intelligence 38, 30–45. doi:10.1109/TPAMI.2015.2414420.

Kölle, M., Laupheimer, D., Schmohl, S., Haala, N., Rottensteiner, F., Wegner, J.D., Ledoux, H., 2021. The Hessigheim 3D (H3D) benchmark on semantic segmentation of high-resolution 3D point clouds and textured meshes from UAV LiDAR and Multi-View-Stereo. ISPRS Open Journal of Photogrammetry and Remote Sensing 1, 11. doi:10.1016/j.ophoto.2021.100001.

Labetski, A., Vitalis, S., Biljecki, F., Arroyo Ohori, K., Stoter, J., 2022. 3D building metrics for urban morphology. International Journal of Geographical Information Science , 1–32doi:10.1080/13658816.2022.2103818.

Lafarge, F., Mallet, C., 2012. Creating large-scale city models from 3D-point clouds: a robust approach with hybrid representation. International Journal of Computer Vision 99, 69–85. doi:10.1007/s11263-012-0517-8.

LandsD, H., 2022. 3D Visualisation Map. https://www.landsd.gov.hk/en/survey-mapping/mapping/3d-mapping.html. Accessed: 2022-11-01.

Langlois, P.A., Boulch, A., Marlet, R., 2019. Surface reconstruction from 3D line segments, in: 2019 International Conference on 3D Vision (3DV), IEEE. pp. 553–563. doi:10.1109/3DV.2019.00067.

Li, L., Song, N., Sun, F., Liu, X., Wang, R., Yao, J., Cao, S., 2022a. Point2Roof: End-to-end 3D building roof modeling from airborne LiDAR point clouds. ISPRS Journal of Photogrammetry and Remote Sensing 193, 17–28. doi:10.1016/j.isprsjprs.2022.08.027.

Li, M., Nan, L., 2021. Feature-preserving 3D mesh simplification for urban buildings. ISPRS Journal of Photogrammetry and Remote Sensing 173, 135–150. doi:10.1016/j.isprsjprs.2021.01.006.

Li, M., Wonka, P., Nan, L., 2016. Manhattan-world urban reconstruction from point clouds, in: European Conference on Computer Vision, Springer. pp. 54–69. doi: 10.1007/978-3-319-46493-0_4.

Li, M., Xue, F., Wu, Y., Yeh, A.G.O., 2022b. A room with a view: Automatic assessment of window views for high-rise high-density areas using City Information Models and deep transfer learning. Landscape and Urban Planning 226, 104505. doi:10.1016/j.landurbplan.2022.104505.

Li, P., Wang, B., Sun, F., Guo, X., Zhang, C., Wang, W., 2015. Q-mat: Computing medial axis transform by quadratic error minimization. ACM Transactions on Graphics 35, 1–16. doi:10.1145/2753755.

Li, Z., Shan, J., 2022. RANSAC-based multi primitive building reconstruction from 3D point clouds. ISPRS Journal of Photogrammetry and Remote Sensing 185, 247–260. doi:10.1016/j.isprsjprs.2021.12.012.

Lin, C., Li, C., Liu, Y., Chen, N., Choi, Y.K., Wang, W., 2021. Point2skeleton: Learning skeletal representations from point clouds, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4277–4286. doi:10.1109/CVPR46437.2021.00426.

Livny, Y., Yan, F., Olson, M., Chen, B., Zhang, H., El-Sana, J., 2010. Automatic reconstruction of tree skeletal structures from point clouds, in: ACM SIGGRAPH Asia 2010 papers, pp. 1–8. doi:10.1145/1866158.1866177.

Ma, J., Bae, S.W., Choi, S., 2012. 3D medial axis point approximation using nearest neighbors and the normal field. The Visual Computer 28, 7–19. doi:10.1007/s00371-011-0594-7.

Malhotra, A., Shamovich, M., Frisch, J., van Treeck, C., 2022. Urban energy simulations using open CityGML models: A comparative analysis. Energy and Buildings 255, 111658. doi:10.1016/j.enbuild.2021.111658.

Mather, J.N., 1983. Distance from a submanifold in Euclidean-space, in: Proceedings of symposia in pure mathematics, AMS. pp. 199–216.

Nan, L., Wonka, P., 2017. Polyfit: Polygonal surface reconstruction from point clouds, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 2353–2361. doi:10.1109/ICCV.2017.258.

PlanD, H., 2022. Urban Design Guidelines. https://www.pland.gov.hk/pland_en/tech_doc/hkpsg/full/pdf/ch11.pdf. Accessed: 2023-10-06.

Rebain, D., Angles, B., Valentin, J., Vining, N., Peethambaran, J., Izadi, S., Tagliasacchi, A., 2019. LSMAT least squares medial axis transform. Computer Graphics Forum 38, 5–18. doi:10.1111/cgf.13599.

Rodríguez, L.R., Duminil, E., Ramos, J.S., Eicker, U., 2017. Assessment of the photovoltaic potential at urban level based on 3D city models: A case study and new methodological approach. Solar Energy 146, 264–275. doi:10.1016/j.solener.2017.02.043.

Saha, P.K., Borgefors, G., di Baja, G.S., 2016. A survey on skeletonization algorithms and their applications. Pattern Recognition Letters 76, 3–12. doi:10.1016/j.patrec.2015.04.006.

Salinas, D., Lafarge, F., Alliez, P., 2015. Structure-aware mesh decimation. Computer Graphics Forum 34, 211–227. doi:10.1111/cgf.12531.

Schnabel, R., Wahl, R., Klein, R., 2007. Efficient RANSAC for point-cloud shape detection, in: Computer Graphics Forum, Wiley Online Library. pp. 214–226. doi:10.1111/j.1467-8659.2007.01016.x.

Song, J., Wu, J., Jiang, Y., 2015. Extraction and reconstruction of curved surface buildings by contour clustering using airborne LiDAR data. Optik 126, 513–521. doi:10.1016/j.ijleo.2015.01.011.

Song, J., Xia, S., Wang, J., Chen, D., 2020. Curved buildings reconstruction from airborne LiDAR data by matching and deforming geometric primitives. IEEE Transactions on Geoscience and Remote Sensing 59, 1660–1674. doi:10.1109/TGRS.2020.2995732.

Stoter, J., Peters, R., Commandeur, T., Dukai, B., Kumar, K., Ledoux, H., 2020. Automated reconstruction of 3D input data for noise simulation. Computers, Environment and Urban Systems 80, 101424. doi:10.1016/j.compenvurbsys.2019.101424.

Sun, F., Choi, Y.K., Yu, Y., Wang, W., 2015. Medial meshes–a compact and accurate representation of medial axis transform. IEEE transactions on visualization and computer graphics 22, 1278–1290. doi:10.1109/TVCG.2015.2448080.

Tagliasacchi, A., Alhashim, I., Olson, M., Zhang, H., 2012. Mean curvature skeletons, in: Computer Graphics Forum, Wiley Online Library. pp. 1735–1744. doi:10.1111/j.1467-8659.2012.03178.x.

Tagliasacchi, A., Delame, T., Spagnuolo, M., Amenta, N., Telea, A., 2016. 3D skeletons: A state-of-the-art report. Computer Graphics Forum 35, 573–597. doi:10.1111/cgf.12865.

Tagliasacchi, A., Zhang, H., Cohen-Or, D., 2009. Curve skeleton extraction from incomplete point cloud, in: ACM SIGGRAPH 2009 papers, pp. 1–9. doi:10.1145/1576246.1531377.

TCC, C., 2013. Tall Building Design Guidelines. https://www.toronto.ca/wp-content/uploads/2018/01/96ea-cityplanning-tall-buildings-may2013-final-AODA.pdf. Accessed: 2023-10-06.

Verdie, Y., Lafarge, F., Alliez, P., 2015. Lod generation for urban scenes. ACM Transactions on Graphics 34. doi:10.1145/2732527.

Wang, S., Cai, G., Cheng, M., Junior, J.M., Huang, S., Wang, Z., Su, S., Li, J., 2020. Robust 3D reconstruction of building surfaces from point clouds based on structural and closed constraints. ISPRS Journal of Photogrammetry and Remote Sensing 170, 29–44. doi:10.1016/j.isprsjprs.2020.09.004.

Wu, Y., Shang, J., Xue, F., 2021. RegARD: Symmetry-based coarse registration of smartphone's colorful point clouds with cad drawings for low-cost digital twin buildings. Remote Sensing 13, 1882. doi:10.3390/rs13101882.

Xia, S., Chen, D., Wang, R., Li, J., Zhang, X., 2020. Geometric primitives in LiDAR point clouds: A review. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13, 685–707. doi:10.1109/JSTARS.2020.2969119.

Xie, L., Hu, H., Zhu, Q., Li, X., Tang, S., Li, Y., Guo, R., Zhang, Y., Wang, W., 2021. Combined rule-based and hypothesis-based method for building model reconstruction from photogrammetric point clouds. Remote Sensing 13, 1107. doi:10.3390/rs13061107.

Xiong, B., Jancosek, M., Elberink, S.O., Vosselman, G., 2015. Flexible building primitives for 3D building modeling. ISPRS Journal of Photogrammetry and Remote Sensing 101, 275–290. doi:10.1016/j.isprsjprs.2015.01.002.

Xue, F., Li, X., Lu, W., Webster, C.J., Chen, Z., Lin, L., 2021a. Big data-driven pedestrian analytics: Unsupervised clustering and relational query based on Tencent Street View photographs. ISPRS International Journal of Geo-Information 10, 561. doi:10.3390/ijgi10080561.

Xue, F., Lu, W., Chen, Z., Webster, C.J., 2020. From LiDAR point cloud towards digital twin city: Clustering city objects based on Gestalt principles. ISPRS Journal of Photogrammetry and Remote Sensing 167, 418–431. doi:10.1016/j.isprsjprs.2020.07.020.

Xue, F., Lu, W., Webster, C.J., Chen, K., 2019. A derivative-free optimization-based approach for detecting architectural symmetries from 3D point clouds. ISPRS Journal of Photogrammetry and Remote Sensing 148, 32–40. doi:10.1016/j.isprsjprs.2018.12.005.

Xue, F., Wu, L., Lu, W., 2021b. Semantic enrichment of building and city information models: A ten-year review. Advanced Engineering Informatics 47, 101245. doi:10.1016/j.aei.2020.101245.

Yan, Y., Letscher, D., Ju, T., 2018. Voxel cores: Efficient, robust, and provably good approximation of 3D medial axes. ACM Transactions on Graphics 37, 1–13. doi:10.1145/3197517.3201396.

Yang, G., Xue, F., Zhang, Q., Xie, K., Fu, C.W., Huang, H., 2023. Urbanbis: a large-scale benchmark for fine-grained urban building instance segmentation, in: ACM SIGGRAPH 2023 Conference Proceedings, pp. 1–11. doi:10.1145/3588432.3591508.

Yang, S., Cai, G., Du, J., Chen, P., Su, J., Wu, Y., Wang, Z., Li, J., 2022. Connectivity-aware graph: A planar topology for 3D building surface reconstruction. ISPRS Journal of Photogrammetry and Remote Sensing 191, 302–314. doi:10.1016/j.isprsjprs.2022.07.024.

Yu, M., Lafarge, F., 2022. Finding good configurations of planar primitives in unorganized point clouds, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6367–6376. doi:10.1109/CVPR52688.2022.00626.

Yuan, L., Lu, W., Xue, F., Li, M., 2023. Building feature-based machine learning regression to quantify urban material stocks: A Hong Kong study. Journal of Industrial Ecology , in pressdoi:10.1111/jiec.13348.

Zhang, K., Yan, J., Chen, S.C., 2006. Automatic construction of building footprints from airborne LIDAR data. IEEE Transactions on Geoscience and Remote Sensing 44, 2523–2533. doi:10.1109/TGRS.2006.874137.

Zhang, W., Li, Z., Shan, J., 2021. Optimal model fitting for building reconstruction from point clouds. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14, 9636–9650. doi:10.1109/JSTARS.2021.3110429.

Zhou, Q.Y., Neumann, U., 2010. 2.5 d dual contouring: A robust approach to creating building models from aerial lidar point clouds, in: Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part III 11, Springer. pp. 115–128.

Zhou, Q.Y., Neumann, U., 2012. 2.5D building modeling by discovering global regularities, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE. pp. 326–333. doi:10.1109/CVPR.2012.6247692.

Appendix A. Supplementary table and figures of experiments

Table A.1: Comparison of DualCont, ManBox, PolyFit, KSR, and the BSS method in terms of compactness, accuracy, and processing time (best values in bold).

	#triangles/#planar regions[#]					RMSD (m)					Time (sec.)				
	DualCont	ManBox	PolyFit	KSR	Ours	DualCont	ManBox	PolyFit	KSR	Ours	DualCont	ManBox	PolyFit	KSR	Ours
1	513 / 114	−⋆	80 / **35**	**68** / 49	136 / 38	1.47	−⋆	0.47	**0.37**	0.40	**0.05**	−⋆	2.26	3.00	2.46
2	830 / 589	−⋆	**16** / **7**	20 / 8	44 / 15	0.67	−⋆	0.387	**0.386**	0.45	**0.09**	−⋆	4.73	4.69	4.62
3	244 / 49	−⋆	58 / **10**	52 / 15	**40** / 14	1.64	−⋆	0.454	0.51	**0.449**	**0.05**	−⋆	1.03	1.63	3.89
4	843 / 131	−⋆	36 / **8**	36 / **8**	**20** / 11	1.03	−⋆	**0.425**	0.426	0.51	**0.1**	−⋆	3.45	4.22	5.71
5	379 / 300	−⋆	**16** / **7**	**16** / **7**	68 / 22	0.94	−⋆	**0.9305**	0.931	0.95	**0.05**	−⋆	3.08	3.33	7.48
6	3,846 / 2038	1,836 / 648	446 / **45**	266 / 48	**252** / 71	0.91	1.05	1.08	0.91	**0.61**	57	**4**	85	81	30
7	8,763 / 5370	684 / 239	488 / **22**	302 / 34	**100** / 29	0.76	1.02	**0.80**	0.86	0.87	79	**5**	122	120	46
8	18,719 / 10349	12,156 / 3506	1,528 / **65**	690 / 102	**636** / 176	1.19	2.25	**1.00**	1.11	1.04	265	**16**	278	238	105
9	19,396 / 11758	52,248 / 15073	2,810 / **146**	1,308 / 169	**1,180** / 300	**0.81**	1.62	1.30	1.56	0.91	219	**20**	389	207	153
10	26,746 / 15307	44,184 / 12608	5,044 / **100**	**1,936** / 189	2,182 / 500	1.08	0.98	1.52	1.20	**0.94**	1,263	**43**	681	408	432
11	25,749 / 15877	25,056 / 7564	3,178 / **111**	**2,120** / 188	2,406 / 587	**0.79**	1.08	1.36	1.69	0.90	504	**28**	1,731	312	372
12	10,126 / 5437	−†	1,940 / **125**	1,204 / 161	**820** / 222	0.85	−†	0.91	**0.67**	0.82	522	−†	314	229	**148**
13	22,358 / 11630	63,408 / 16667	1,160 / **152**	934 / 200	**744** / 209	1.10	1.64	0.96	**0.94**	0.96	300	**54**	347	278	118
14	8,582 / 5168	708 / 270	388 / **32**	**378** / 39	534 / 140	**0.73**	5.93	3.21	1.09	1.05	55	**5**	108	108	117
15	10,035 / 5746	1,128 / 410	408 / **32**	362 / 39	**184** / 54	**0.75**	4.78	4.74	1.14	1.41	62	**9**	139	140	101
Avg.	10,475 / 5991	22,379 / 6332	1,173 / **60**	646 / 84	**623** / 159	0.98	2.26	1.30	0.92	**0.82**	222	**20**	281	143	110

⋆: Manhattan-World assumption not met, no experiments conducted; †: RMSD > 10m.

[#]: Planar regions are extracted by region growing with a distance threshold of 0.01 m, an angle threshold of 5°, and a minimum region size of 1 m^2.

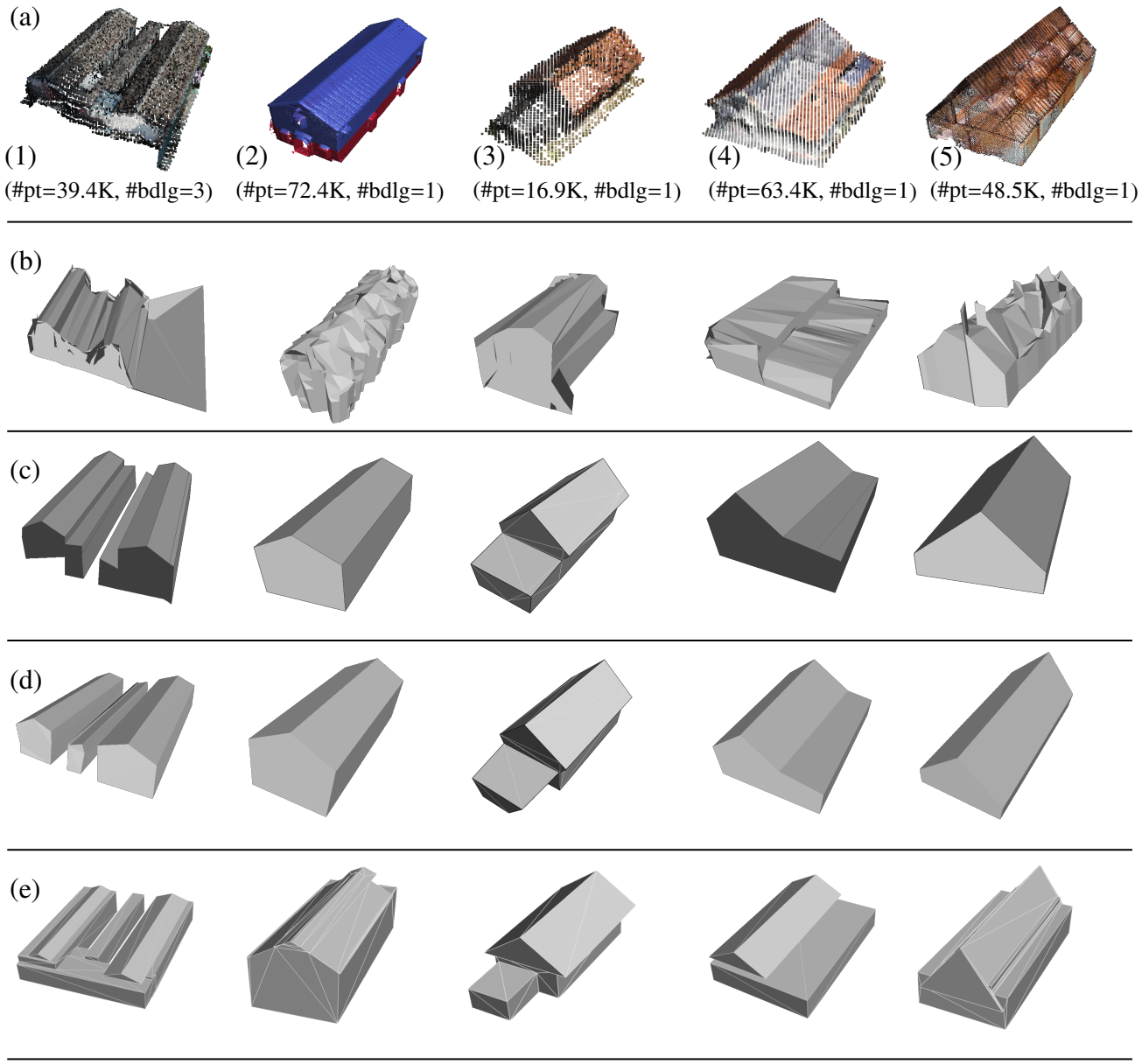


Figure A.1: Results of Samples (1) to (5). The corresponding distance deviations are plotted in Fig. A.4. (a) Inputs (#pt and #bdlg refer to the number of input points and buildings, respectively); (b) Results of DualCont; (c) Results of PolyFit; (d) Results of KSR; and (e) Results of our BSS method.

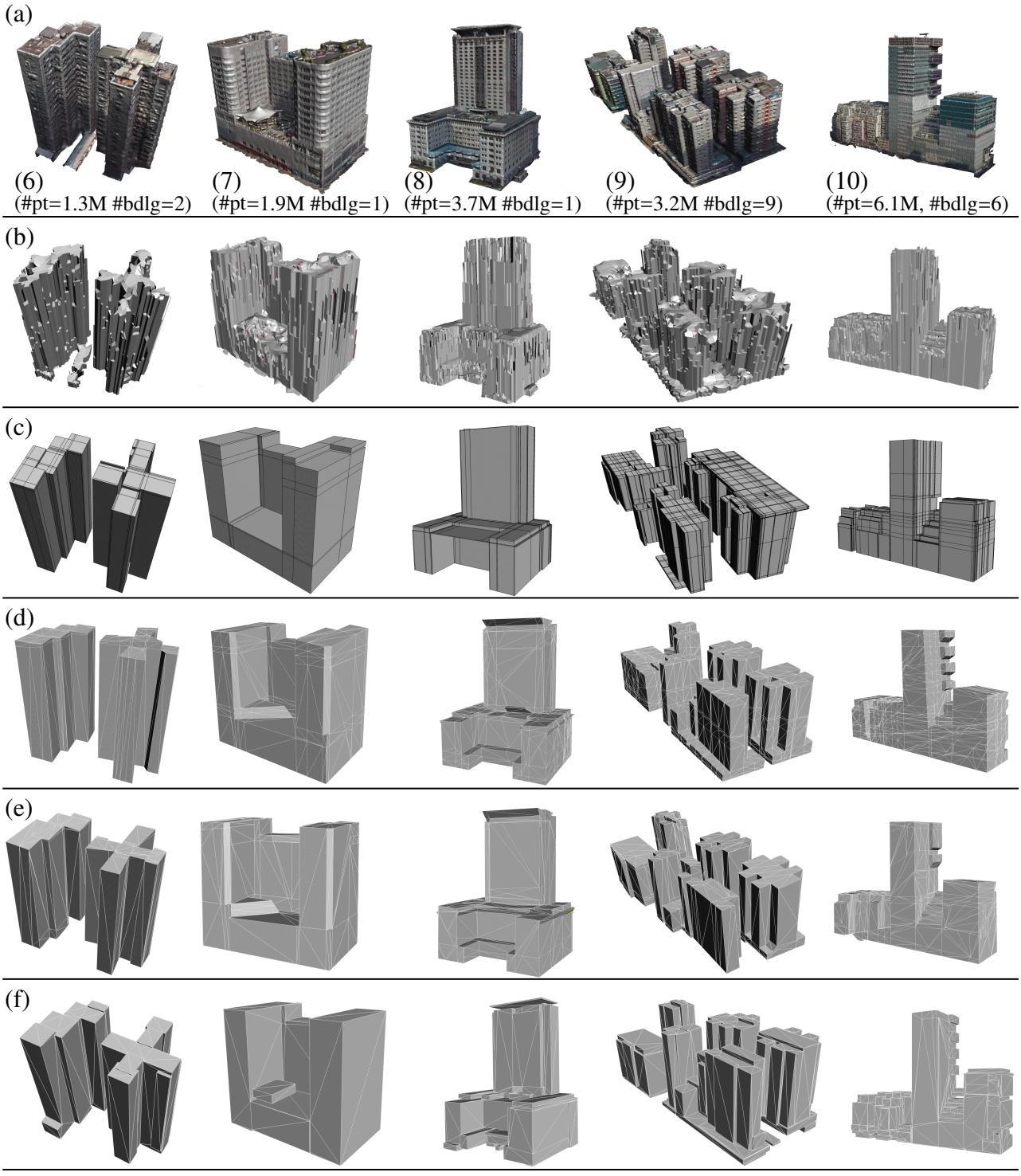


Figure A.2: Results of Samples (6) to (10). The corresponding distance deviations are plotted in Fig. A.5. (a) Inputs (#pt and #bdlg refer to the number of input points and buildings, respectively); (b) Results of DualCont; (c) Results of ManBox; (d) Results of PolyFit; (e) Results of KSR; and (f) Results of our BSS method.

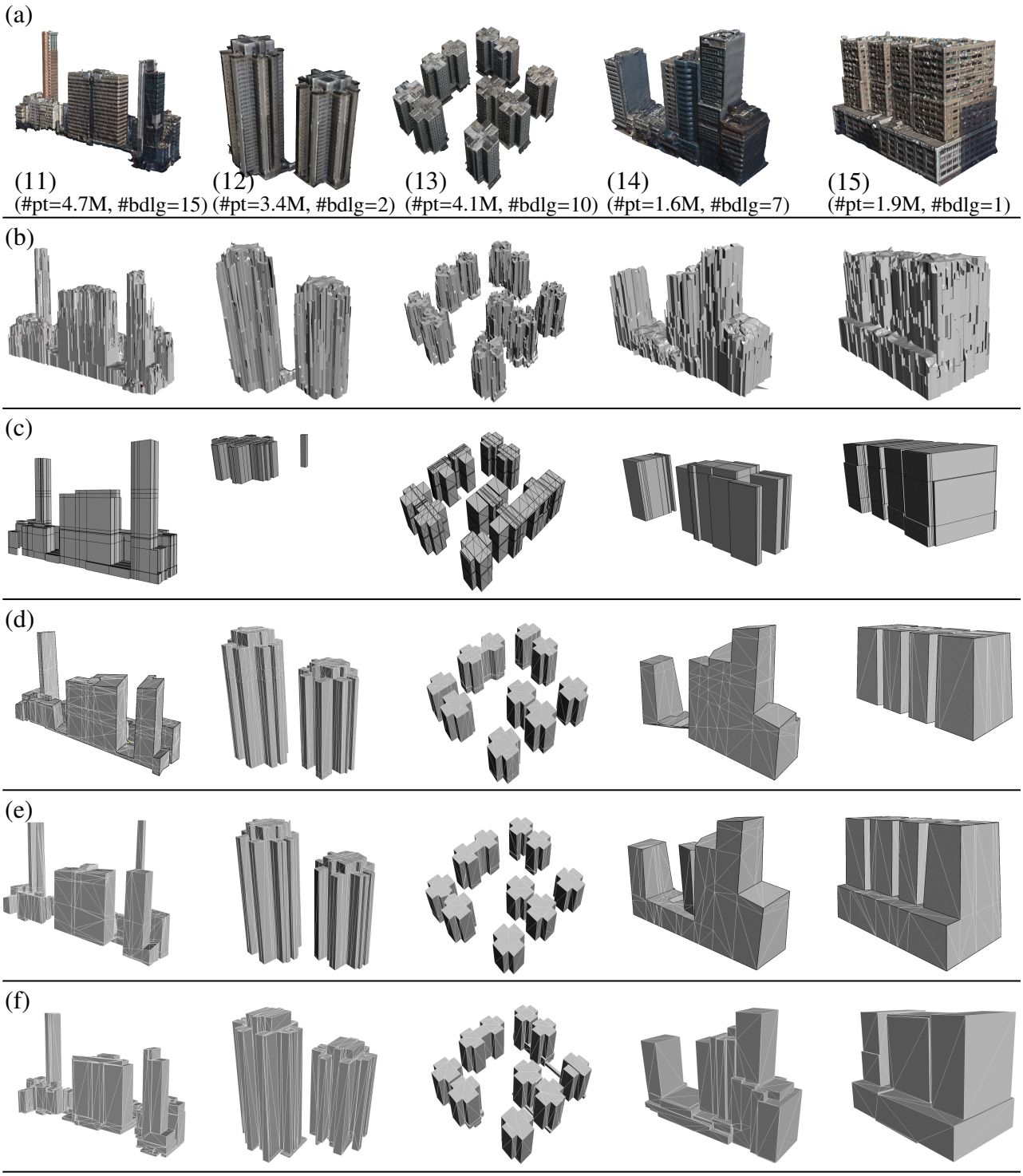


Figure A.3: Results of Samples (11) to (15). The corresponding distance deviations are plotted in Fig. A.6. (a) Inputs (#pt and #bdlg refer to the number of input points and buildings, respectively); (b) Results of DualCont; (c) Results of ManBox; (d) Results of PolyFit; (e) Results of KSR; and (f) Results of our BSS method.

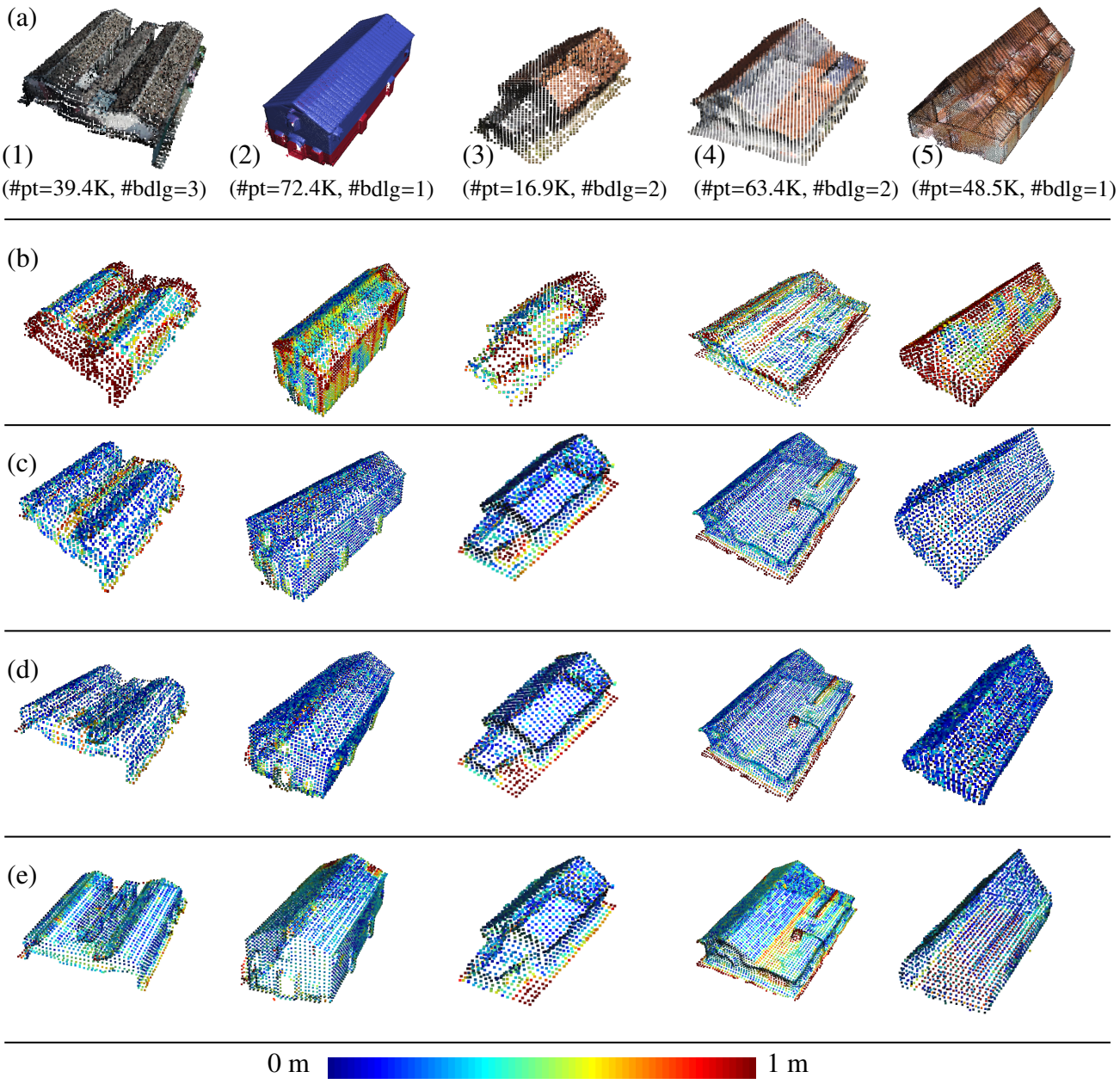


Figure A.4: Distance deviations from the input point clouds to the reconstructed meshes of Samples (1) to (5). The corresponding reconstructed meshes are shown in Fig. A.1. (a) Inputs (#pt and #bdlg refer to the number of input points and buildings, respectively); (b) Deviations of DualCont; (c) Deviations of PolyFit; (d) Deviations of KSR; and (e) Deviations of our BSS method.

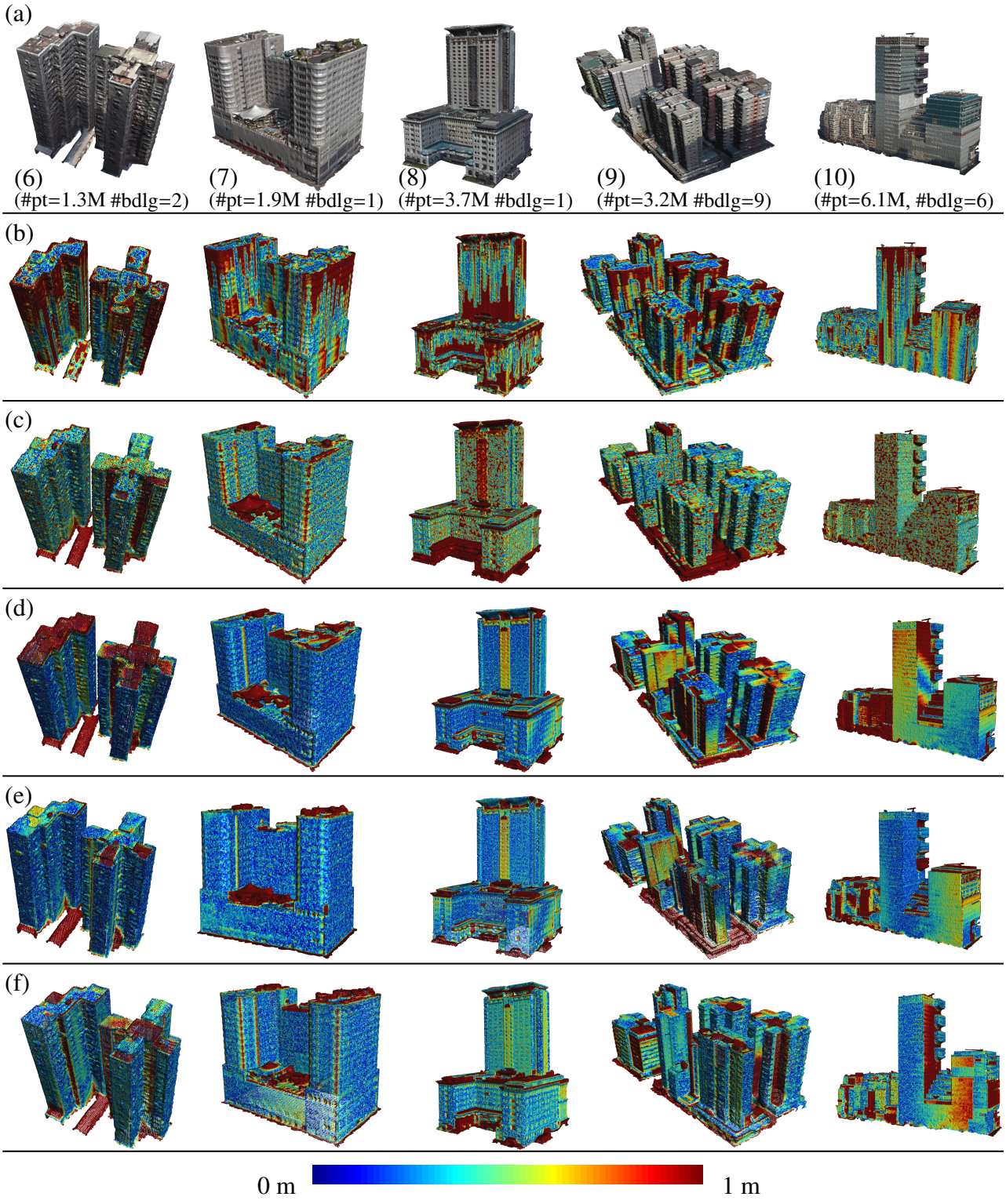


Figure A.5: Distance deviations from the input point clouds to the reconstructed meshes of Samples (6) to (10). The corresponding reconstructed meshes are shown in Fig. A.2. (a) Inputs (#pt and #bdlg refer to the number of input points and buildings, respectively); (b) Deviations of DualCont; (c) Deviations of ManBox; (d) Deviations of PolyFit; (e) Deviations of KSR; and (f) Deviations of our BSS method.

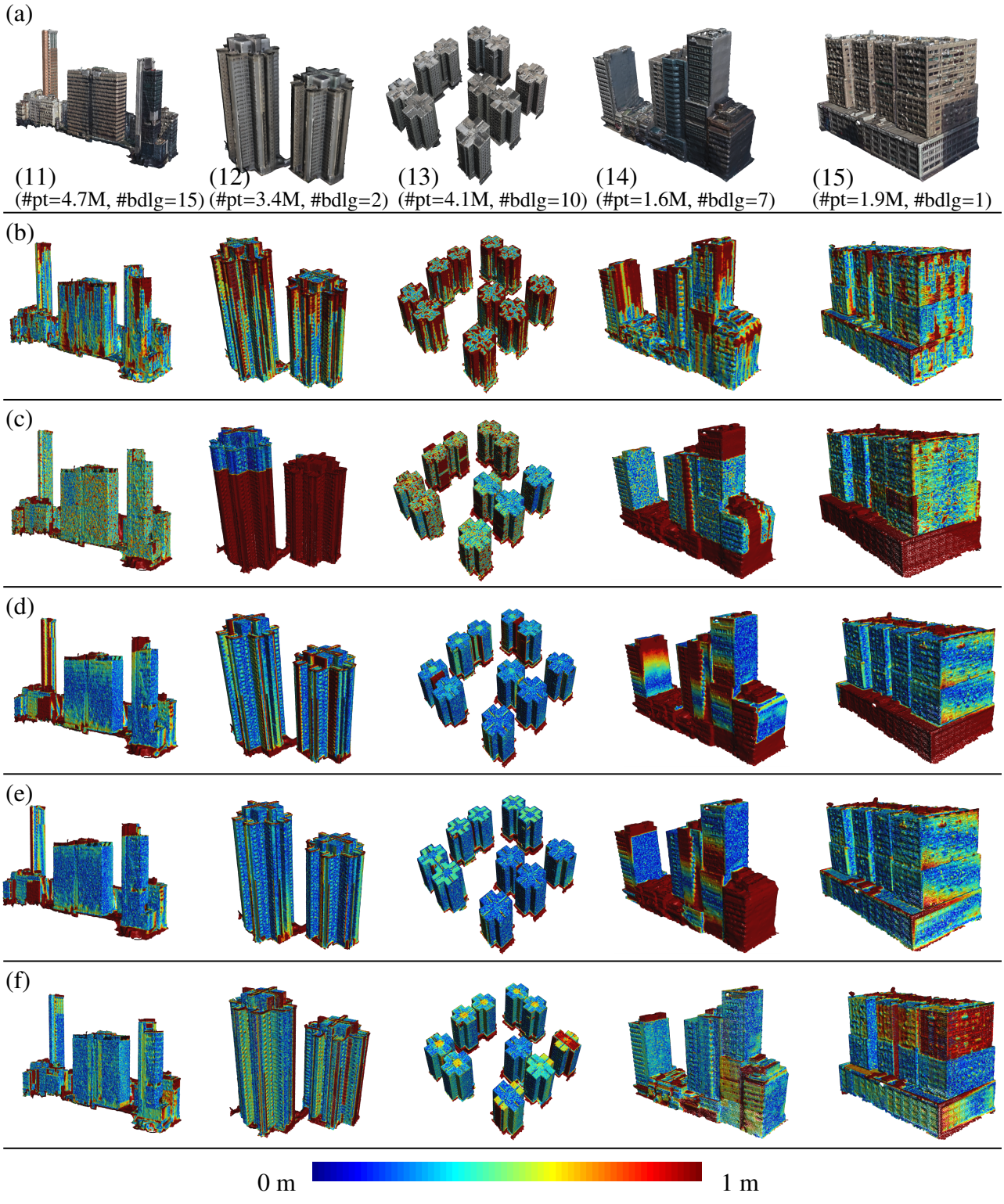


Figure A.6: Distance deviations from the input point clouds to the reconstructed meshes of Samples (11) to (15). The corresponding reconstructed meshes are shown in Fig. A.3. (a) Inputs (#pt and #bdlg refer to the number of input points and buildings, respectively); (b) Deviations of DualCont; (c) Deviations of ManBox; (d) Deviations of PolyFit; (e) Deviations of KSR; and (f) Deviations of our BSS method.

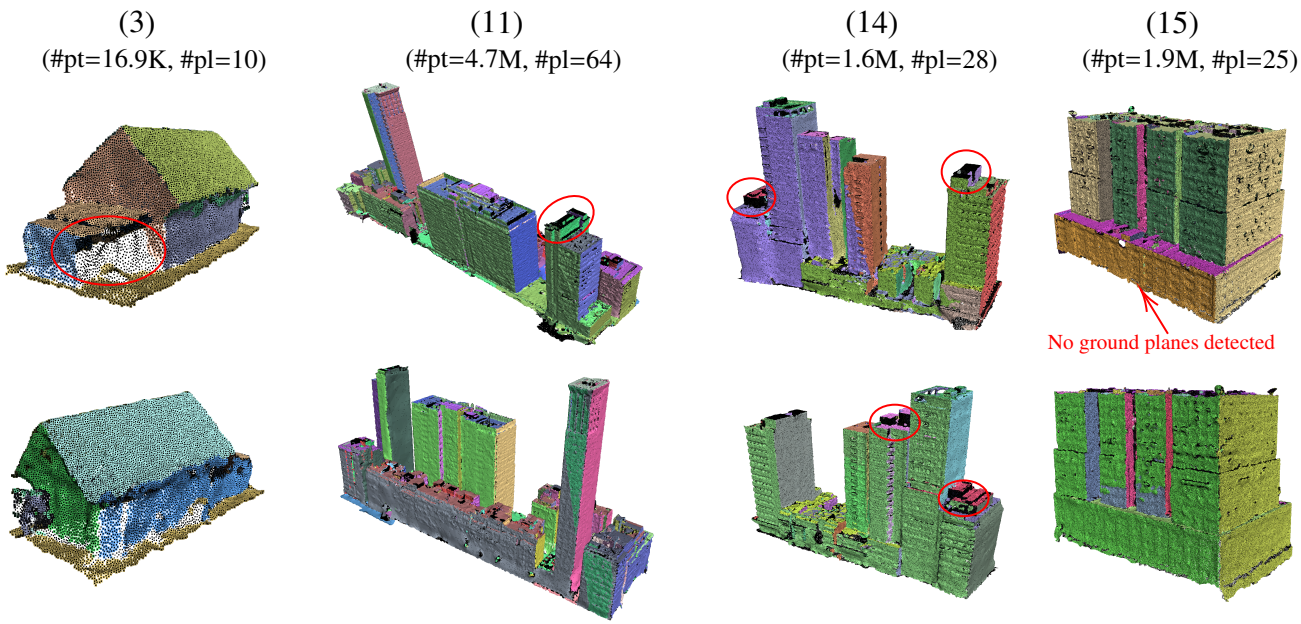


Figure A.7: The inputs to PolyFit and KSR for the four samples (3, 11, 14, and 15) shown in Fig. 9. The planar segments were detected by RANSAC (Schnabel et al., 2007). #pt and #pl refer to the number of points and detected planar segments, respectively. Points are colorized in terms of planar segments; Points belong to no planar segments are in black. The red circles highlight the missing planes that affected the reconstruction.