

GENERATING LIGHTWEIGHT BUILDING MODELS WITH PRESERVED STRUCTURAL FEATURES FROM NOISY 3D MESHES

R.Z. Guo¹, T. Yu¹, W.X. Wang¹, X.M. Li¹, S.J. Tang¹, L.F. Xie^{1,*}

¹ Research Institute for Smart Cities, School of Architecture and Urban Planning, Shenzhen University, 518060 Shenzhen, China -
(guorz, wangwx, lixm, shengjuntang, linfxie)@szu.edu.cn, 2210324012@email.szu.edu.cn

Commission II, WG II/1

KEY WORDS: Building models, 3D reconstruction, Polygonal mesh, Mesh simplification, Connectivity graph.

ABSTRACT:

Obtaining building instance models directly through photogrammetry and other means results in high polygon counts and structural detail levels. Therefore, it is an important challenge to preserve the features of instant building models and generate lightweight building models from them. In this paper, we propose an improved lightweight reconstruction method for 3D building models based on planar primitives extraction, topology correction, and optimal planes selection. Improvements due to our method arise from three aspects: (1) After plane segmentation based on a simple region growing method, a second plane segmentation is performed on the building model based on the similarity of normal vectors. (2) According to the building structural characteristics, the initial plane segmentation is checked to generate new plane regions within the necessary connection regions. (3) The intersection of candidate planes is improved by enlarging the region of candidate faces, and the topological connection is improved. Furthermore, the topological relations of all planar primitives that have been optimized are recorded in an undirected graph. Finally, a watertight and manifold lightweight model is extracted from the faces of a candidate set by energy minimization. Experiments on different data sets show that the improved method is more reasonable in plane segmentation, and has superior performance in algorithm robustness and necessary structure recovery of buildings. Even when dealing with imperfect data, a watertight model is still obtained.

1. INTRODUCTION

With the continuous development of photogrammetry technology and the constant improvement of data collection methods, it was previously necessary to obtain high-precision 3D models through a large amount of complex manual modeling. Nowadays, with the help of photogrammetry, these models can be easily constructed.

Building instance segmentation is a method of processing mesh models obtained through photogrammetry to obtain a single instance model of a building. However, a typical city scene usually contains dozens or even hundreds of building models. For unprocessed building models, the file size of a regular building model can reach hundreds of megabytes. For large buildings such as schools, shopping malls, and high-rise residential buildings, the data size of a single building model may even reach several gigabytes.

The massive amount of data not only hinders the large-scale application and promotion of 3D urban geographic information (Li et al., 2011). In addition, directly generated building models typically include unnecessary details such as textures, exquisite structures, and model errors in augmented reality navigation, disaster assessment simulation, etc. Therefore, lightweight building models have become an effective solution to address these issues. Lightweight building models preserve the structural characteristics of the original model while significantly

reducing its data size, thereby improving the efficiency and experience of these scene applications.

Many methods have been proposed in recent years for generating lightweight building models, such as data-driven methods (Verdie et al., 2015; Chen et al., 2017; Ochmann et al., 2019), nonlinear modeling methods (Schnabel et al., 2007; Zhou and Neumann, 2013), rule-based and hypothesis-based modeling methods (Li et al., 2016a; Li et al., 2016b; Nan and Wonka, 2017; Li and Nan, 2021). Although data-driven methods can simplify the complexity of building models, they are sensitive to the density, noise, and missing data, resulting in low geometric accuracy. While nonlinear modeling methods can improve the geometric accuracy of models, they weaken the prominence of model boundary expression to a certain extent. Rule-based and hypothesis-based methods not only reduce the sensitivity of algorithms to data quality, but also enhance the abstract expression of models, preserving the structural features of building models that comply with prior assumptions.

Planar structure is an important part of most urban building models. Methods based on the plane hypothesizing strategy have been proposed in large numbers (Boulch et al., 2014; Nan and Wonka, 2017; Bouzas et al., 2020; Liu et al., 2022; Liu et al., 2023). However, it is not difficult to see that automatic generation of lightweight building polygonal models still faces some challenges. For example, Due to surface defects of the model and deficiencies in existing plane-segmentation methods, there are problems of under-segmentation or over-segmentation. It is also

* Corresponding author

a difficult problem to recover the necessary topological connections from the defective building model. Similarly, the incorrect plane segmentation also brings difficulties to the topology recovery. For these reasons, the rationality and similarity between the final polygonal model and the input model are not satisfactory.

To solve these problems, we optimize the plane extraction and topology recovery. Our input is an instantiated building mesh model extracted from an urban scene. Based on the Structure-Aware Building Mesh Polygonization (SABMP) method (Bouzas et al., 2020), we establish several objectives in the reconstruction process: (1) the primitives extracted by our proposed method have good noise resistance and are as close to each other as possible. (2) the necessary plane segmentation of the structural part of the building model needs to be recovered. (3) Even with imperfect data as input, our method can still extract the overall outline of the building model. Therefore, the significant contributions of our work are as follows:

- (1) A two-stage region growing algorithm is proposed. Based on the initial plane region segmentation, the algorithm grows again to reduce the influence of noise and make the initial extracted planes as adjacent as possible.
- (2) The proposed method inspects the defect and loss parts of the plane segmentation, and generates new planes in the corresponding regions with relatively relaxed conditions, so that the final flat main primitive is as free from connection errors as possible.
- (3) Topology recovery between planar primitives fails due to various reasons such as missing data caused by tree occlusion. A strategy is designed to solve this problem by detecting plane intersections in an expanded range of candidate faces.

2. RELATED WORK

Lightweight building model generation involves a large number of computer graphics related techniques. By indexing the previous literature, this part of the paper reviews several aspects related to our proposed method: (1) planar shape detection. (2) building reconstruction based on planar primitives. (3) mesh simplification.

2.1 Planar shape detection

Extracting correct planar primitives from noisy and defective mesh is crucial for constructing accurate polygonal models. The common practice for this particular task is the Random Sample Consensus (RANSAC) algorithm (Fischler and Bolles, 1981; Zuliani et al., 2005; Li et al., 2009) and variants based on region growing. The seed region is determined according to some rules, and then expanded outward according to the distance or the difference of the normal vector to complete the extraction of the plane region. In recent years, due to the popularity of deep learning, the method of extracting shape primitives based on model database training has also been proposed (Fang et al., 2018; Li et al., 2019). These methods have achieved good results in the experiment.

However, due to the influence of the initial parameter setting and the lack of consideration of the specific situation, the detection of planar primitives usually under-segmented or over-segmented, which affects the recovery of the necessary building structure from the original building mesh model. Therefore, correct planar shape detection remains an important challenge in model generation based on planar primitives.

2.2 Building reconstruction based on planar primitives

The method of extracting the surface primitives of the object model and then reconstructing it can not only greatly reduce the amount of data of the original model, but also retain and enhance the main features of the model. which is widely used in building reconstruction. As the main feature of buildings, the reconstruction of building models based on extracting planar primitives is widely used.

The methods of dividing 3D space into polyhedral regions by extend planar primitives, and then select a subset of candidates' faces to form an output model (Boulch et al., 2014; Oesau et al., 2014; Verdie et al., 2015), the main problems of which are the tremendous consumption of time and memory space.

In order to reduce the impact of the computational cost, PolyFit (Nan and Wonka, 2017) filters and simplifies detected primitives to obtain the final model from a candidate face set based on binary linear programming formulation. In order to further reduce the problem of selecting a large number of candidate faces after plane segmentation, inspired by the graph of proxies of Structure-Aware Mesh Decimation (SAMD) (Salinas et al., 2015), SABMP constructs the connection graph according to the strict adjacency relationship between primitives, which avoids the intersections between invalid planar primitives and greatly reduces the time and computational cost. However, too strict adjacency relations can also lead to the failure of polygonal model generation.

2.3 Mesh simplification

The edge collapse-based method is very common in grid model simplification (Garland and Heckbert, 1997; Lindstrom and Turk, 1999; Salinas et al., 2015), which processes the mesh model by introducing a cost function using the two vertices of a given edge connecting vertices.

Although this method is highly versatile and does not require the original model's topological integrity, it is difficult to guarantee the model's similarity when the model is significantly simplified. A refined mesh decimation method is proposed (Li and Nan, 2021) based on mesh filtering preprocessing and a hierarchical error metric by which the detected planes could be well preserved in the edge collapse iterations.

For building modeling, methods based on the plane hypothesizing strategy (Boulch et al., 2014; Nan and Wonka, 2017; Bouzas et al., 2020; Liu et al., 2022; Liu et al., 2023) can greatly reduce the original model data volume while retaining the main structural features of the building. These methods are reliable to some extent. However, the main problems of these methods are the under-segmentation and over-segmentation of planar primitives, the construction of correct topological relations between primitives and the time consumption.

3. PROPOSED METHOD

The framework for our method is designed into three main stages: (1) plane segmentation, (2) topological map construction, and (3) polygonal mesh reconstruction. Figure 1 shows the overall workflow of the proposed method.

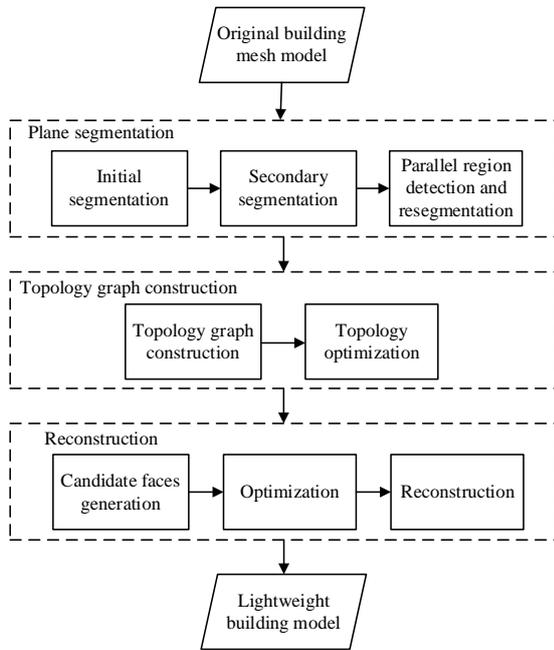


Figure 1. Pipeline of the proposed method.

3.1 Plane segmentation

For plane segmentation, the proposed method adopts a staged plane segmentation algorithm.

In the initial segmentation stage, following the approach of Bouzas et al (Bouzas et al., 2020), we calculate the planarity of all mesh vertices in the k -ring and derive the planarity of all mesh faces from the vertices' planarity, and calculate the planarity of all mesh faces according to planarity of mesh vertices. Subsequently, we select the mesh face with the highest planarity as the seed face for region growing. Employing Principal Components Analysis (PCA) (Pauly et al., 2002), we analyze the k -ring neighborhoods of the seed. If the vertices within the neighborhoods satisfy the distance threshold, we add the faces to the region. Finally, the whole process is repeated until all faces of the mesh are divided into different planar regions.

Following the initial segmentation, we assign an importance value to each planar segment, representing the ratio of its area to the total surface area of the mesh. By defining a threshold for importance, we partition the plane regions into the important planar segment region R_{imp} and the unimportant planar segment region R_{unimp} . The secondary assignment of the plane segmentation is done by redistributing the facets in region R_{imp} to its adjacent region R_{unimp} according to Equation 1. Where n_1 represents the normal vector of face in region R_{unimp} , n_2 represents the normal vector of the fitted plane in region R_{imp} , and θ_{thr} represents the set threshold. The value of θ_{thr} is set to 45° in the experiments in this paper.

$$abs(n_1 * n_2) > \cos(\theta_{thr}) \quad (1)$$

The specific implementation process is shown in Figure 2, traversing the faces in region R_{unimp} , and proceeds according to the following process:

- (1) Find the corresponding faces in region R_{unimp} , whose adjacent faces are in region R_{imp} , generally located at the outer boundary of this region.

- (2) Compare the normal vector n_1 of the corresponding face and the normal vector $\{n_2\}$ of the set of adjacent plane regions $\{R_{imp}\}$, and find the Angle between the normal vector n_2 for the corresponding region R_{imp} and n_1 to the smallest Angle. If the Angle between their normal vectors satisfies Equation 1, the face is merged into the region R_{imp} . If not, the face is classified as unclassified region $R_{unclass}$. In this paper, the color of the faces in region $R_{unclass}$ is black.
- (3) When the region is completely divided or all the outer boundaries of the region do not satisfy Equation 1, the loop is broken out.

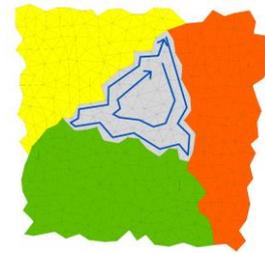


Figure 2. Secondary allocation schematic diagram. Where the yellow, green, and red regions represent R_{imp} , and the gray region represents R_{unimp} . The blue lines indicate the traversal facet order

Figure 3 illustrates the process of secondary assignment of the plane segmentation shown in Figure 2. Faces with an identical color correspond to a separate region. The initial segmentation divides the initial mesh into different regions, with the region set $\{R_{unimp}\}$ indicated by red and blue boxes. It can be observed that the faces within the red box are affected by noise, causing them to be assigned to another region. The faces within the blue box are also assigned to region $\{R_{unimp}\}$. If not addressed, these faces would not participate in subsequent operations, ultimately leading to the failure of generating the subsequent topological graph.

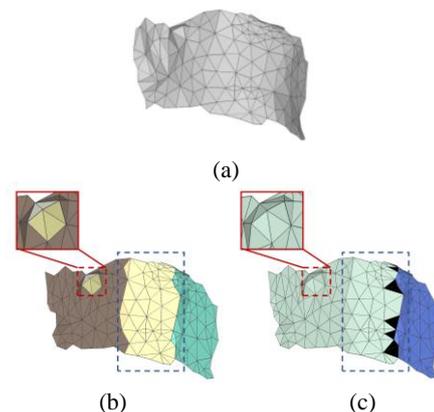


Figure 3. An overview of the process of secondary assignment of the plane segmentation. (a)Initial mesh; (b) Initial segmentation; (c) Secondary assignment

As shown in Figure 3(c), after secondary allocation, the faces that still has some regions are divided into unclassified region $R_{unclass}$ due to non-meeting of the specified requirements. In most cases, the effect on the results is small. As shown in some research (Liu et al., 2022; Liu et al., 2023), if the faces of the region between two parallel regions is not utilized, it will lead to the failure of the final polygon model generation. In order to solve this problem, the following treatment is performed on the

unclassified regions, Figure 4 shows the results achieved by this treatment:

- (1) The breadth-first traversal (BFS) of the reallocated mesh is performed, and the unclassified region $R_{unclass}$ is further divided into different regions $\{R_{sub_unclass}\}$ based on connectivity.
- (2) Traverse each region $\{R_{sub_unclass}\}$, check whether there is a parallel plane region in the adjacent region of $R_{sub_unclass}$, and determine whether the adjacent regions are parallel by the angle of the lines where the normal of planes is located, and the angle threshold is set to 15° in this experiment.
- (3) For the parallel regions that meet the angle threshold, the normal vector-based area growth division plane is adopted, and the threshold can be relatively relaxed in the process of dividing the plane, such as setting the regional growth conditions to 75° in this experiment.

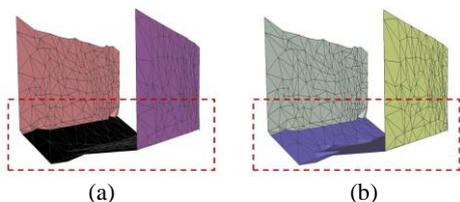


Figure 4. The generation of a new planar region between parallel regions. (b) is the result of the generation of the case shown in (a).

3.2 Topology graph construction

After the operations in Section 3.1, the necessary planar primitives are generated. In addition, we need to construct undirected graphs to preserve the intrinsic relationships between planar primitives. Existing methods define the connection relationship between nodes in a graph by sharing at least one edge in the region corresponding to the two planar primitives (Bouzaz et al., 2020; Fang and Lafarge, 2020).

However, due to the inherent flaws in the occlusion or input mesh data, this strict connection constraint can lead to the failure of subsequent processes. As shown in the red box mark in Figure 5(a), due to the defects of the mesh itself, the generated topological graph in (b) is incomplete, resulting in the generation of the scaffold line of the model in (c) is also incomplete, and finally the polygon model generation fails.

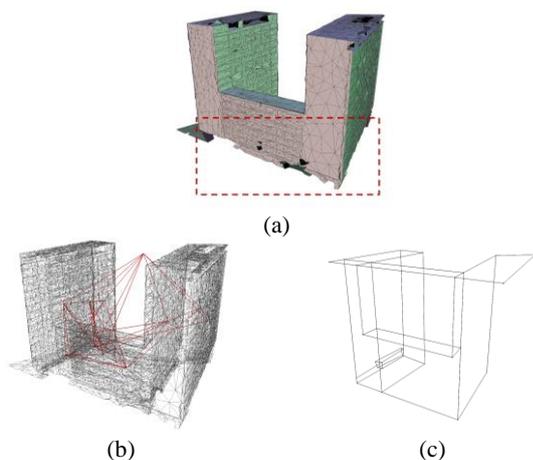


Figure 5. Generation of topology maps under strict constraints. (a) Defective building models; (b) Red connecting lines indicate visualization of the topology map; (c) Building scaffold line

In order to solve this problem, the method of intersection by searching in the expanded range is proposed (Xie et al., 2021; Yang et al., 2022). Inspired by this method, this process is done only for part of the planes by filtering the planes before solving the plane intersections in the enlarged range. After the strict graph generation, the intersection points and intersection lines can be solved by intersection:

- (1) The planes represented by two adjacent nodes v_i and v_j intersect to generate a straight line.
- (2) The three v_i, v_j, v_k adjacent to each other to form a closed loop to generate the intersection point.

As shown in Figure 6(a), there is only one intersection line on the straight line in the red box, then the generation of the building scaffolding line will fail, and the finding of the problem plane is completed by finding the plane generating the intersection line. The plane indicated by the red arrow in Figure 6 (b) is the one that generates the intersecting lines in the red box in Figure 6 (a).

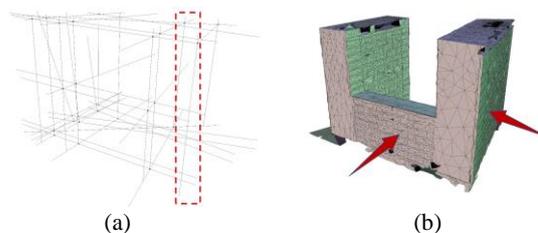


Figure 6. (a) Examples of lines intersecting defects; (b) The defect line corresponds to the plane

For the problem plane, the facets that lie in the plane are projected onto the plane and the continuous boundaries of these facets are extracted using α -shapes. At the same time, the plane is intersected with the plane in the expanded range, and the intersecting line is also projected onto the plane, as shown in Figure 7, where the blue line is the 2D line after projection. The intersection should satisfy the following two conditions:

- (1) The intersection line should not intersect with the interior of the α -shapes polygon.
- (2) The intersecting lines must intersect within the expanded range.

If the condition is satisfied, an edge is added between nodes v_i, v_j corresponding to the plane. Figure 8 (a) shows the visualization result of the improved graph, and Figure (b) shows the scaffolding line of the building after doing this step.

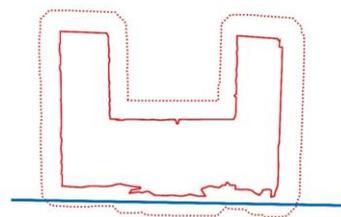


Figure 7. The solid red line shows the α -shapes polygon, the dashed line shows the extended boundary, and the blue line shows the intersection cases that are satisfied

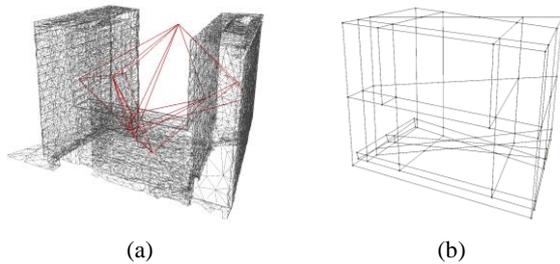


Figure 8. (a) Visualization of the improved graph; (b) the Improved scaffolding line of the building model

3.3 Reconstruction

After the building scaffolding line is generated, the corresponding candidate faces are also generated. The final reconstruction needs to select and assemble the appropriate candidate faces to generate the final polygonal mesh model. The energy function as shown in Equation 2 is established, where E_f is the Face coverage energy term, E_c is the Data fitting energy term, and E_m is the Model complexity. The energy terms, $\{\lambda_i\}$ are their coefficients. A polygonal grid model with watertightness and manifold is obtained after minimization. Refer specifically to the work of SABMP.

$$\begin{aligned} & \min_x \lambda_f \cdot E_f + \lambda_c \cdot E_c + \lambda_m \cdot E_m \\ & s. t. \begin{cases} \sum_{j \in N(e_i)} x_j = 2 \text{ or } 0, & 1 \leq i \leq |E| \\ x_i \in \{0,1\}, & 1 \leq i \leq N \end{cases} \end{aligned} \quad (2)$$

4. RESULTS AND DISCUSSION

The proposed method was implemented in C++ with CGAL (<https://www.cgal.org/>), and all methods are optimized using the same mathematical solver. The optimizer used in solving the final model is SCIP solver (Vigerske and Gleixner, 2018). This section will explore the experimental results of the proposed method compared to the original method on different datasets. The experiments were conducted on one core of an Intel(R) Xeon(R) W-2275 CPU @ 3.30GHz 3.31 GHz, with 64.0 GB RAM.

4.1 Dataset and implementation details

We applied our method to a real-life building model from Shenzhen, China, generated by Context Capture, and to 2 buildings of various architectural styles in the SUM Helsinki 3D dataset (Gao et al., 2021). In this paper, we call them building 1, Building 2, Building 3 and they are shown in Figure 9(a), Figure 10(a), Figure 11(a). These data are all an open mesh data, and the input data structure of building 1 is obviously missing.

For the parameter settings, θ_{thr} in Equation (1) is set to 45° , The normal Angle threshold for region generation in parallel regions in 3.1 is set to 75° . In Equation (2), $\lambda_f=0.43$, $\lambda_c=0.27$, $\lambda_m=0.3$. And compared with the original method, we set the number of planes after division to the same state, and discuss the generation of lightweight polygon models after generating the same number of planes.

4.2 Experimental results

The experimental results of the two methods in building 1, building 2 and building 3 are shown in Figure 9, Figure 10 and Figure 11. Where (a) represents the input building model ,

(b)(d)(f)(h) show respectively the results of plane segmentation, topology map visualization, building scaffolding lines and polygon mesh generation of SABMP. (c)(e)(g)(i) respectively represent the corresponding results of the proposed method. We assess the conformity of our simplified version to the original model by computing the Hausdorff distance between the two meshes (Guthe et al., 2005). From Table 1, we observe that the RMSE error is small for these meshes, even though these models may have model imperfections (building 1), uneven ground (building 2), or tree occlusions (building 3), which indicates that our simplified versions closely follow the initial building models.

Mesh	#1	#2	#3
Faces(original)	19313	43437	18128
Faces(simplified)	124	360	132
RMSE(%BBox Diagonal)	0.012	0.020	0.019

Table 1. Comparison of the number of facets and RMSE error between the simplified polygon mesh model generated by the proposed method and the original mesh model

From the building scaffolding line and the final polygon mesh generation, generation effect of SABMP in these three kinds of buildings is not satisfactory. The generated building scaffolding lines are not complete, and the generated polygon mesh is only a part of the original model. There are mainly the following two reasons:

- (1) Plane under-segmentation. As shown in Figure. 11 (b) and 12 (b), there is no segmentation between the planar regions between the parallels, resulting in no generation of building scaffolding in the corresponding region, which directly leads to the generation of the final model.
- (2) Incompleteness of the topological graph. The generated topology map and the processing time of the whole process are shown in Table 2. SABMP makes strict constraints on the connection between graph nodes, requiring that at least one edge adjacency is required to add an edge between the corresponding nodes. This strict constraint is not friendly for the case that the input model is defective, and the requirement for planar segmentation is relatively high. As shown in Figure. 11 (b), due to the defects in the input model, there is no adjacent edge between the facade of some building models and the ground, which ultimately leads to the failure of model generation. In the model of Figure. 11 (a), due to the large fluctuation between the ground and the building, the plane cup of the ground is divided into multiple planes in the plane division, which eventually leads to defects in the topology map.

However, as can be seen from Table 2, when the topology graph is more complex, the whole processing time usually will be longer. For example, in building 3, only 3 nodes were added, but 16 edges were added. This is related to the setting of the expanded border inspection range threshold, but it is also related to the structure of the model itself. The more complex the structure is, the more likely this problem is to exist. However, on the whole, the increased time is generally acceptable.

		graph node	graph edge	time
#1	SABMP	12	29	2.4s
	ours	14	40	1.9s
#2	SABMP	39	119	7.6s
	ours	40	121	16.2s
#3	SABMP	18	48	0.8s
	ours	22	64	1.8s

Table 2. Comparison of topological graph and lightweight mesh model generation time

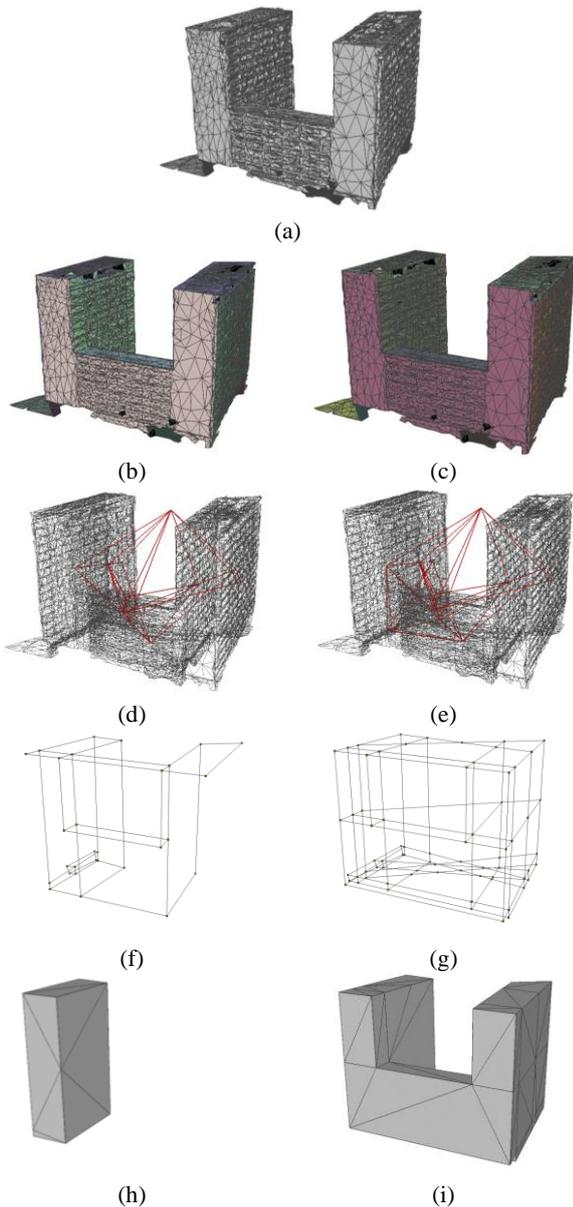


Figure 9. Comparison of the two methods for building 1.

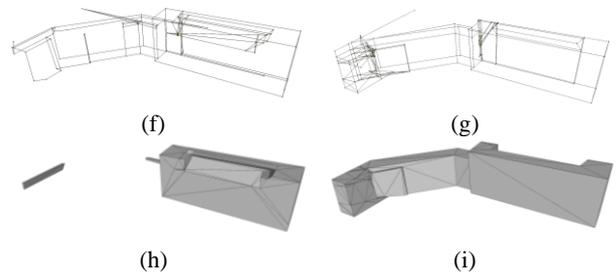
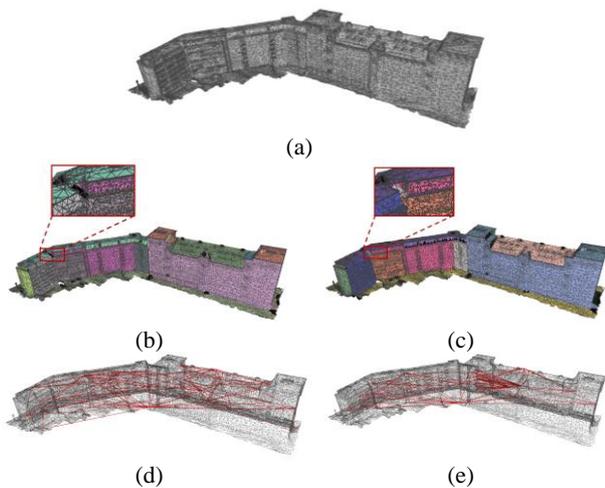


Figure 10. Comparison of the two methods for building 2.

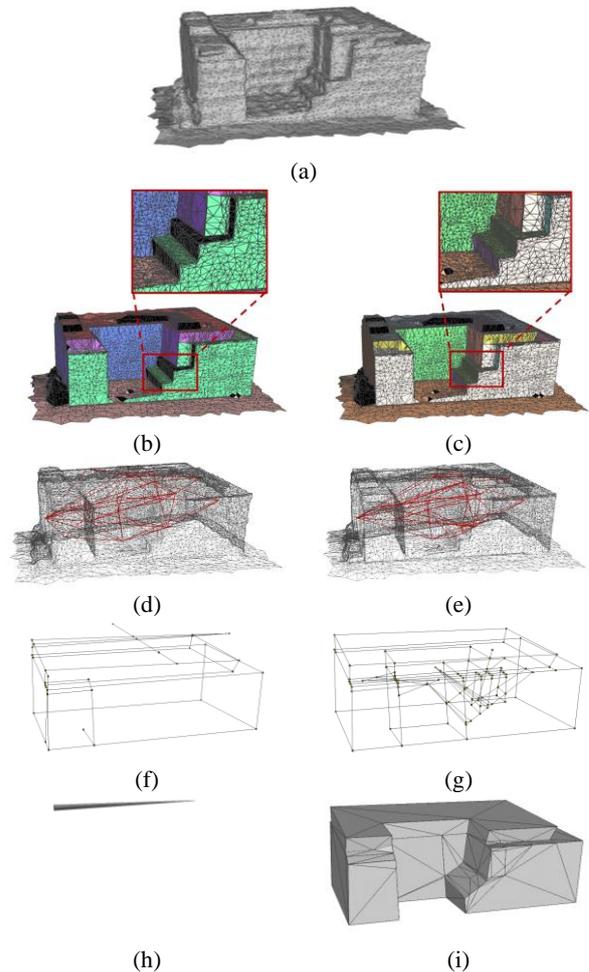


Figure 11. Comparison of the two methods for building 3.

As shown in Figure 11(i), the proposed method directly generates a plane at the region of the model shown in Figure 11 (c). This is due to the fact that the model is smooth in this part, and the threshold set for region generation is relatively loose, resulting in plane under-segmentation. However, it can be seen that the final generated polygonal grid model basically retains the main characteristics of the original model, and the impact of the plane under-segmentation here is small.

5. CONCLUSION AND FUTUREWORK

In this paper, we improve the SABMP algorithm, and do further processing in the plane segmentation and topology graph generation. Through experiments on different experimental data, compared with SABMP method, it can be concluded that the improved algorithm greatly improves the robustness of the

algorithm, and produces a reliable lightweight polygon mesh model.

However, the time consumption of generating a lightweight model for a building model with too complex structure is still a problem. The uncertain connection relationship between primitives caused by model defects still needs to be solved.

ACKNOWLEDGEMENTS

This work is supported by the National Key Research and Development Program of China (Projects Nos. 2022YFB3903700), the National Natural Science Foundation of China (42001407, 41971341, 41971354), the Shenzhen Science and Technology Program (Grant No. RCBS20221008093124064), the Guangdong Science and Technology Strategic Innovation Fund (the Guangdong–Hong Kong–Macau Joint Laboratory Program) (Project No.: 2020B1212030009).

REFERENCES

Boulch, A., de La Gorce, M., and Marlet, R., 2014: Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization, *Comput. Graph. Forum*, 33, 55-64. 10.1111/cgf.12431.

Bouzas, V., Ledoux, H., and Nan, L. L., 2020: Structure-aware Building Mesh Polygonization, *ISPRS-J. Photogramm. Remote Sens.*, 167, 432-442. 10.1016/j.isprsjprs.2020.07.010.

Chen, D., Wang, R. S., and Peethambaran, J., 2017: Topologically Aware Building Rooftop Reconstruction From Airborne Laser Scanning Point Clouds, *IEEE Trans. Geosci. Remote Sensing*, 55, 7032-7052. 10.1109/tgrs.2017.2738439.

Fang, H. and Lafarge, F., 2020: Connect-and-Slice: an hybrid approach for reconstructing 3D objects, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13490-13498.

Fang, H., Lafarge, F., and Desbrun, M., 2018: Planar shape detection at structural scales, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2965-2973.

Fischler, M. A. and Bolles, R. C., 1981: A paradigm for model fitting with applications to image analysis and automated cartography (reprinted in readings in computer vision, ed. ma fischler, *Comm. ACM*, 24, 381-395.

Gao, W. X., Nan, L. L., Boom, B., and Ledoux, H., 2021: SUM: A benchmark dataset of Semantic Urban Meshes, *ISPRS-J. Photogramm. Remote Sens.*, 179, 108-120. 10.1016/j.isprsjprs.2021.07.008.

Garland, M. and Heckbert, P. S., 1997: Surface simplification using quadric error metrics, *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 209-216.

Guthe, M., Borodin, P., and Klein, R., 2005: Fast and accurate Hausdorff distance calculation between meshes.

Li, B., Schnabel, R., Jin, S. Y., and Klein, R., 2009: Variational Surface Approximation and Model Selection, *Comput. Graph. Forum*, 28, 1985-1994. 10.1111/j.1467-8659.2009.01577.x.

Li, L., Sung, M., Dubrovina, A., Yi, L., and Guibas, L. J., 2019: Supervised fitting of geometric primitives to 3d point clouds, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2652-2660.

Li, M., Schmitz, A., and Kobbelt, L., 2011: Pseudo-immersive real-time display of 3d scenes on mobile devices, 2011 *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, 41-48.

Li, M., Wonka, P., and Nan, L., 2016: Manhattan-world urban reconstruction from point clouds, *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV* 14, 54-69.

Li, M. L. and Nan, L. L., 2021: Feature-preserving 3D mesh simplification for urban buildings, *ISPRS-J. Photogramm. Remote Sens.*, 173, 135-150. 10.1016/j.isprsjprs.2021.01.006.

Li, M. L., Nan, L. L., Smith, N., and Wonka, P., 2016b: Reconstructing building mass models from UAV images, *Comput. Graph.-UK*, 54, 84-93. 10.1016/j.cag.2015.07.004.

Lindstrom, P. and Turk, G., 1999: Evaluation of memoryless simplification, *IEEE Trans. Vis. Comput. Graph.*, 5, 98-115. 10.1109/2945.773803.

Liu, X. Y., Zhu, X. Z., Zhang, Y. J., Wang, S. Y., and Jia, C., 2023: Generation of concise 3D building model from dense meshes by extracting and completing planar primitives, *Photogramm. Rec.*, 38, 22-46. 10.1111/phor.12438.

Liu, Y. W., Guo, B. X., Wang, S., Liu, S. K., Peng, Z. M., and Li, D. M., 2022: Urban Building Mesh Polygonization Based on Plane-Guided Segmentation, Topology Correction and Corner Point Clump Optimization, *Remote Sens.*, 14, 24. 10.3390/rs14174300.

Nan, L. and Wonka, P., 2017: Polyfit: Polygonal surface reconstruction from point clouds, *Proceedings of the IEEE International Conference on Computer Vision*, 2353-2361.

Ochmann, S., Vock, R., and Klein, R., 2019: Automatic reconstruction of fully volumetric 3D building models from oriented point clouds, *ISPRS-J. Photogramm. Remote Sens.*, 151, 251-262. 10.1016/j.isprsjprs.2019.03.017.

Oesau, S., Lafarge, F., and Alliez, P., 2014: Indoor scene reconstruction using feature sensitive primitive extraction and graph-cut, *ISPRS-J. Photogramm. Remote Sens.*, 90, 68-82. 10.1016/j.isprsjprs.2014.02.004.

Pauly, M., Gross, M., and Kobbelt, L. P.: Efficient simplification of point-sampled surfaces, *IEEE Visualization, 2002. VIS 2002.*, 163-170.

Salinas, D., Lafarge, F., and Alliez, P., 2015: Structure-Aware Mesh Decimation, *Comput. Graph. Forum*, 34, 211-227. 10.1111/cgf.12531.

Schnabel, R., Wahl, R., and Klein, R., 2007: Efficient RANSAC for point-cloud shape detection, *Comput. Graph. Forum*, 26, 214-226. 10.1111/j.1467-8659.2007.01016.x.

Verdie, Y., Lafarge, F., and Alliez, P., 2015: LOD Generation for Urban Scenes, *ACM Trans. Graph.*, 34, 14. 10.1145/2732527.

Vigerske, S. and Gleixner, A., 2018: SCIP: global optimization of mixed-integer nonlinear programs in a branch-and-cut framework, *Optim. Method Softw.*, 33, 563-593. 10.1080/10556788.2017.1335312.

Xie, L. F., Hu, H., Zhu, Q., Li, X. M., Tang, S. J., Li, Y., Guo, R. Z., Zhang, Y. T., and Wang, W. X., 2021: Combined Rule-Based and Hypothesis-Based Method for Building Model Reconstruction from Photogrammetric Point Clouds, *Remote Sens.*, 13, 23. 10.3390/rs13061107.

Yang, S. M., Cai, G. R., Du, J., Chen, P., Su, J. H., Wu, Y. D., Wang, Z. Y., and Li, J. A. T., 2022: Connectivity-aware Graph: A planar topology for 3D building surface reconstruction, *ISPRS-J. Photogramm. Remote Sens.*, 191, 302-314. 10.1016/j.isprsjprs.2022.07.024.

Zhou, Q. Y. and Neumann, U., 2013: Complete residential urban area reconstruction from dense aerial LiDAR point clouds, *Graph. Models*, 75, 118-125. 10.1016/j.gmod.2012.09.001.

Zuliani, M., Kenney, C. S., and Manjunath, B.: The multiransac algorithm and its application to detect planar homographies, *IEEE International Conference on Image Processing 2005*, III-153.