

A Quantitative Texture Feature Preserving Method for Simplifying 3D Building Models

Po Liu, Chengming Li* Zhanjie Zhao and Zhendong Liu
Chinese Academy of Surveying and Mapping, Beijing, 100083, China
*Corresponding author

Abstract—Three-dimensional (3D) city models have become an important component of spatial information infrastructure, and widely used in studies of virtual geographical environments, urban planning, and navigation, as well as other fields. Level of detail (LOD) representation of 3D models provides a foundation for the large-scale visualization and application of building models, and simplification is one of the most popular approaches used to generate multi-resolution 3D building models. This paper proposes a texture feature preserving approach to the quantitative simplification of manually produced building models. First, the original models are converted to polygonal models with the B-Rep method. The structural features of the buildings are then extracted and used to perform quantitative simplification of the building models while preserving the structural features. Next, the textures of new triangles are synthesized from the original textures. Finally, a post-processing algorithm is used to detect overlapping and/or intersecting triangles to improve the simplification results. Experimental results show that the proposed method can be adapted to models of ordinary buildings and landmarks. It can be directly applied to large-scale building network visualization.

Keywords—building models; LOD; quantitative simplification; structural feature; texture synthesis

I. INTRODUCTION

Recent advances in the capturing of 3D spatial data have led to the generation of large numbers of highly detailed 3D building models. 3D city models have become an important component of spatial information infrastructure, and they are widely used in studies of virtual geographical environments, urban planning, and navigation, as well as other fields (Ehlers, 2014; Goodchild et al., 2012; Liu, Gong, & Yu, 2014, 2015). To exchange and share these existing models, five levels of models have been defined to meet different application requirements in the OGC CityGML standard (Gröger & Plümer, 2012; Hong et al., 2015; Kolbe, 2009). LOD representation of 3D models is the foundation of the large-scale visualization and application of models. In an urban environment, building models are the most important elements of 3D urban maps. Thus, we primarily investigate the simplification of building models in this paper.

Although these highly detailed models yield a realistic appearance for the user, the presentation of such models has two main disadvantages. First, the user may be distracted from the essential parts due to the presence of large amounts of unimportant detail (Jiang, Yang, & Sun, 2011). Second, the large data volumes involved present considerable challenges in

terms of storage, transmission and real-time visualization (Tan Kim & Daman, 2004). LOD representation of 3D models provides an efficient solution for this problem; however, many building models have no or only partial LOD information (Rau, Chen, Tsai, Hsiao, & Hsu, 2006). The LOD representations of 3D building models are often created interactively, which is time consuming and error-prone. It is necessary to derive a method that can automatically create multiple abstractions for complex 3D building models that also maintains geometric, topological, and semantic consistency. The simplification of three-dimensional building models has received considerable attention in the field of geographic generalization (Meng & Forberg, 2007; Regnauld, 2001).

Although many approaches for the simplification of building models have been proposed, they mainly focus on geometric and semantic characteristics; texture feature are rarely considered (Mao & Ban, 2013). Texture features are closely related to the visual perceptions of people. People usually identify urban buildings based on their surface texture features, which thus constitute a unique urban city image (Remco et al., 2006). Moreover, in the manual production of building models, complex geometric details are usually replaced with rich textures to reduce the number of triangles. Building model simplification should preserve geometric and topological features and also reconstruct textures to maintain the surface appearances of buildings. This paper proposes a texture-preserving quantitative simplification algorithm for use with manually produced building models. The primary contributions of our method are as follows. 1) It represents a robust feature-constrained quantitative simplification algorithm to generate multi-resolution representations of 3D building models. 2) The textures are synthesised from the original models based on the contraction history. 3) It can be applied to models of both ordinary buildings and landmarks, thus enabling network-based transmission and visualization.

II. A TEXTURE-PRESERVED BUILDING SIMPLIFICATION ALGORITHM

The main steps of the algorithm are as follows. 1) Boundary representation (B-Rep) model from the original model. 2) Geometric simplification: Based on the topological information, the boundaries, feature lines and feature corners of the building are extracted. Based on quadric error metrics, the edges are collapsed until the shape is represented using a certain number of triangles, based on a certain penalty coefficient that enables structural feature constraint with

normal and area validation. 3) Texture synthesis: based on the edge contraction history, the texture of every triangle is synthesised from the original texture. 4) Post-processing: the overlapping or intersecting triangles are detected, and their vertices are offset in the normal direction.

A. B-Rep Model

Prior to model simplification, the B-Rep model of a building mesh is constructed from the original data. According to the vertex positions, the original models are reconstructed, together with the related index and texture information. The B-Rep model is composed of vertexes, edges, and triangles. A vertex has an ID, a position (X, Y, Z) and a valid flag. Vertexes are connected by edges, and an edge has a start and an end point, an ID, and a valid flag. Triangles are made up of three interconnected edges. A triangle has three edge IDs and a corresponding texture coordinate, a texture image and a valid flag. B-Rep can quickly retrieve the adjacent vertex, edge and triangle ID information, and the topological relationships are preserved in the mesh simplification.

B. Geometric Simplification

This paper introduces a feature-constrained simplification method. The structural characteristics of building models, including boundaries, shape edges and shape vertexes, are first extracted. These features are then assigned certain penalty coefficients, and the edges are further collapsed based on priorities using area and normal constraints. Finally, the building models are quantitatively simplified while preserving the structure of the buildings.

1) Structural Features

We consider the features of a model as those that are both geometrically important and visually important to human perception. For a building model, it is necessary to preserve the building structure characteristics that are important for representing the visual surfaces of buildings. The building structure characteristics boundaries, feature lines and feature corners are defined. A boundary is an edge that has only one adjacent triangle. The quadric errors mentioned earlier do not make any allowance for open boundaries and shape edges. Feature lines are sharp edges whose two adjacent faces have a dihedral angle that is less than some threshold value. These lines reflect the overall geometric appearance of a model and are visually important; Feature corners. Vertexes that have three or more connected adjacent edges and have two or more edges are feature lines. The measurement of local vertex roughness enables improved preservation of high-frequency details in the mesh.

2) Edge Collapse

Assume that a set of planes contain the triangle faces that meet at a vertex V and that the set of planes is defined as: $f(v) = \{(a, b, c, d) \mid ax + by + cz + d = 0\}$, the quadric error $\Delta'(v)$ of this vertex with respect to this set as the sum of squared distances to its planes is:

$$\Delta'(v) = \sum_{p \in \text{planes}(v)} d_p^2(v) = \sum_{p \in \text{planes}(v)} v^T(k_p)V = v^T \left\{ \sum_{p \in \text{planes}(v)} K_p \right\} V \quad (1)$$

and:

$$K_p = pp^T = \begin{bmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix} \quad (2)$$

Triangles with smaller areas are less important. Each vertex v of the original mesh is assigned the sum of the quadrics on its adjacent faces, weighted by the area of each face:

$$Q = \sum_{p \in \text{plane}(v)} \text{Area}_p * K_p \quad (3)$$

Here, Area_p is the area of triangle face p . If edge $V_i V_j$ collapses to \bar{v} , we must derive a new matrix \bar{Q} which approximates the error at \bar{v} . We have chosen to use a simple additive rule:

$$\bar{Q} = \mu(Q_i' + Q_j') \quad (4)$$

Here, μ is the penalty factor for structural features; the default value of this factor is 1. To perform the contraction, we must also choose a position v that minimizes \bar{Q} . Our decision is based on the assumption that new points v do not move far from their original positions. The edge collapse chosen is the one with the minimum distance. Assuming that matrix \bar{Q} is invertible, we can compute the minimum value v :

$$V = \begin{bmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (5)$$

If this matrix is not invertible, a simple scheme would be to select either v_i , v_j , or $(V_i + V_j) / 2$.

To compute the initial quadric matrices required for our pair contraction algorithm, each vertex must accumulate the planes containing the faces that meet at that vertex. These constraint features, such as boundaries, sharp lines and sharp corners, are then converted into quadrics, weighted using a large penalty factor, and added into the initial quadrics for the endpoints of the edge. All the pairs are placed in a list sorted by cost with the pair having the minimum cost at the top. Pairs

are then removed iteratively from the list, the edges contract, and the costs of the related edges are updated.

3) Normal and Area Validation

The general algorithm outlined so far performs well on most building models. However, there are a few important enhancements which improve its performance on certain types of building models. Pair contractions may cause mesh inversion and do not necessarily retain concave structures, which greatly influence texture generation. For building models, it is necessary to verify area and normal in the contraction. When considering a possible contraction, we first compare the normal of each neighboring face before and after the contraction. The angle change θ of the normal is calculated as:

$$\theta = \arccos \left[\frac{\vec{a} \cdot \vec{b}}{|\vec{a}| |\vec{b}|} \right] \quad (6)$$

Where \vec{a} and \vec{b} are the normal of the face before and after simplification, respectively. If θ exceeds some threshold, then some neighboring faces may fold over on each other. Second, we compute the total areas S_1 and S_2 of the neighboring faces before and after contraction. The total area change Δ_s is:

$$\Delta_s = S_1 / S_2 \quad (7)$$

If Δ_s exceeds a certain threshold, then the structure may be lost. If the normal or area does not pass the validation, the contraction is penalized heavily and disallowed. The costs of the related edges are then updated, and the other edges are contracted iteratively.

4) Simplification threshold

The structural features can also be used to compute the threshold N_{thres} of an edge collapse:

$$N_{thres} = \lambda (N_1 + N_2 + N_3) \quad (8)$$

N_1 , N_2 and N_3 are the numbers of boundaries, sharp lines and sharp corners of the mesh model, respectively, whereas λ is the multiplication coefficient.

5) Texture Synthesis

When the positions of vertexes change, it is necessary to synthesize new textures and calculate texture coordinates. The main steps in the procedure are as follows.

- Record the history of edge collapse: When an edge is contracted, the IDs of the related original triangles are recorded. We then obtain the relationships between

every simplified triangle and the corresponding triangles in the original model.

- According to the positions of the vertexes of the new triangle, we calculate the resolution of the texture image and the texture coordinates. The width and height of the image are raised to the power of 2.
- We rotate the plane of the new triangle and determine the rotation matrix. Using the rotation matrix, the vertices of the corresponding original triangles are projected onto this plane.

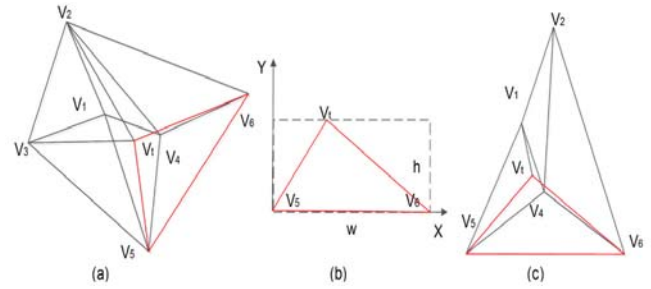


FIGURE 1. TRIANGLE PROJECTION: A) THE NEW TRIANGLE IS DEFINED; B) ROTATION OF THE TRIANGLE; C) TRIANGLE PROJECTION

6) Rotation Matrix and Triangle Projection

Assume that a new triangle has its longest edge along the horizontal axis, and the direction to the last point is the longitudinal axis. Using this rotation matrix, the corresponding original triangle vertices are projected onto the plane, and a new set of projection triangles are obtained. As shown in Figure 1, segment V1V4 is contracted to Vt, and the texture of plane VtV5V6 is resampled from planes V4V5V6, V2V4V6, V1V4V2 and V4V1V5. These planes are all projected onto the plane VtV5V6. The texture coordinates of Vt, V5 and V6 are (h/w, 1), (0, 0) and (1, 0), respectively.

7) Texture Generation

Due to the limited resolutions of images, chart boundaries are revealed if interchart areas are left unpainted. This phenomenon may give rise to crack artefacts between adjacent triangles. The first issue can be solved by extending the surrounding triangles and features that are retained by the geometric simplification. We provide a detailed solution for crack artefacts here. The main steps in this algorithm are as follows.

- The original projective triangles are rendered with the OpenGL interface. We enable the GL_DEPTH_TEST and GL_SMOOTH statesets before rendering the textures.
- The texture of the new triangle is narrowed by b (the default value is 2) pixels size in the width and height directions, and the scale matrix S is calculated. The texture coordinate of the original triangles are all multiplied by matrix S. These triangles are then rendered with the new texture.
- Three vertexes are rendered with their texture coordinates, and the sizes of the points are b+1 pixels.

- The three edges of the triangle are rendered, and the line width is $b+1$ pixels.
- If the model has baked textures, the baked textures are rendered at the same time as the normal texture.

As shown in Figure 2a, the solid line shows the original edge, and the dashed line indicates the edge after it has been scaled using matrix S . The green line in Figure 2b shows the edge in the synthesised texture. The outside texture is synthesised using line and vertex rendering.

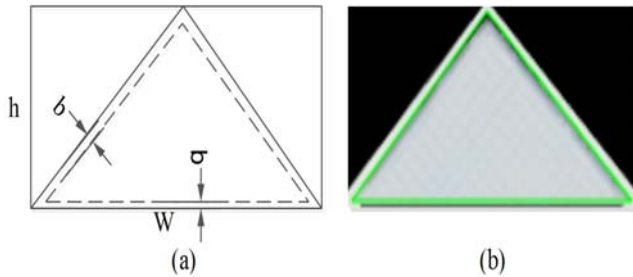


FIGURE II. TEXTURE SYNTHESIS: A) TEXTURE SCALING; B) THE GREEN LINES SHOW THE EDGES OF THE TRIANGLE

8) Post-Processing

A modern building model usually has many features, such as windows, balconies and chimneys. These parts may degenerate into individual triangles after simplification. Moreover, these triangles may intersect with or overlap the planes that define the main body of the building and flash when the building is visualized on a computer. It is necessary to detect these overlap triangles after simplification.

III. EXPERIMENTAL

To evaluate the simplification approach, manually produced LOD3/LOD4 models of buildings located in Beijing are quantitatively and qualitatively evaluated. These models represent a large number of ordinary buildings and landmarks. The total number of models is 24,460, and the data size is about 6G. The penalty factors used for boundaries, feature lines and feature corners are 1000, 500 and 500, respectively. The default value of the penalty factor for normal and area validation is 200, and the default value of λ is 4.

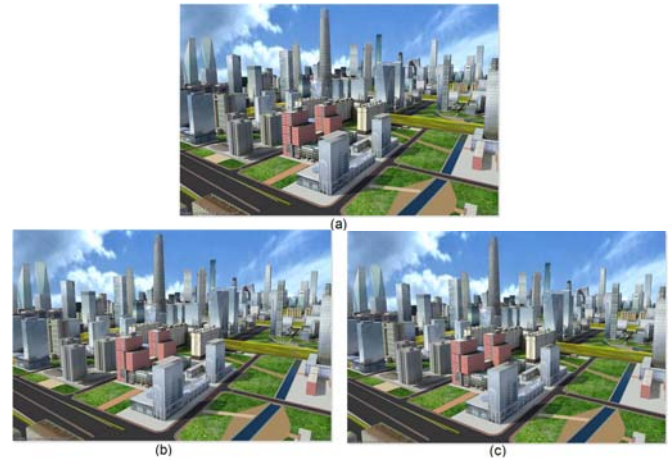


FIGURE III. BUILDING SIMPLIFICATION: A) LOD3 MODELS; B) LOD2 MODELS; C) LOD1 MODELS

A. Qualitative Evaluation

Qualitative evaluation involves visual assessment of the quality of a simplification and texture synthesis. The method proposed in this paper obtains good results for models of both ordinary buildings and landmarks. The original building model shown in Figure 3a and 3b (which shows a wireframe representation of the model) contains 1878 triangles. Figure 4c and 4d show visualizations of the LOD2 and LOD1 models. The number of triangles is reduced to 50% and 25% of the number used in the original model, respectively. The structure of the building is well represented by all of these models.

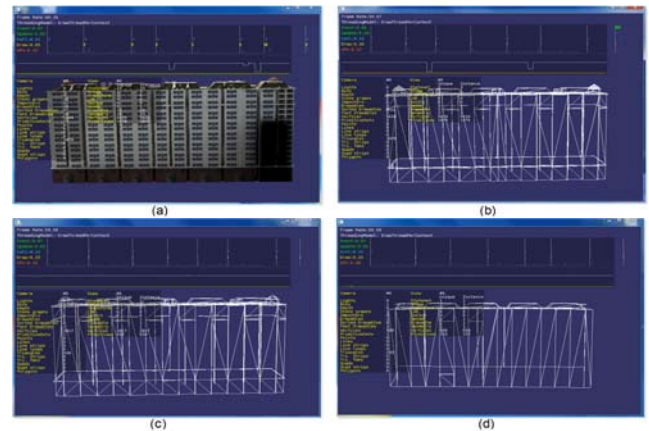


FIGURE IV. BUILDING MODEL SIMPLIFICATION: A) ORIGINAL MODEL; B) LOD3 MODEL; C) LOD2 MODEL; D) LOD1 MODEL

Figure 5 shows the effect of simplifying a landmark with a detailed texture; such landmarks are very common in Beijing. Figure 4a is the original model of the landmark, and Figure 5b is its wireframe visualization. As is shown in Figure 5c and 5d, the number of triangles is reduced to 50% and 25% compared to the original model, respectively, and the overall structure of the landmark is well preserved by all of the models.

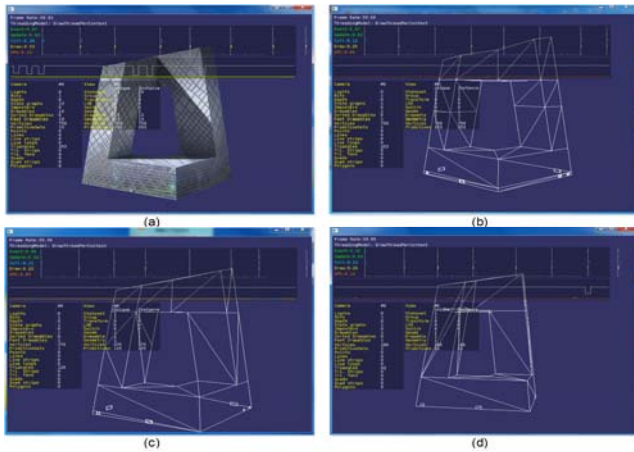


FIGURE V. LANDMARK SIMPLIFICATION: A) ORIGINAL MODEL; B) LOD3 MODEL; C) LOD2 MODEL; D) LOD1 MODEL

B. Quantitative Evaluation

To quantitatively assess the quality of multi-resolution models, similarity measure criteria have been defined. As shown in Table 1, compared to the QEM-based simplification approach, our approach is more suitable for building model simplification; the model similarity is improved, as assessed using the structural feature and area constraints. The geometric similarity decreased gradually from high levels to low levels at the same time.

TABLE I. STATISTICAL RESULTS

Name-level	Number of triangles	Similarity
Model 1-LOD2	939	99.1%
LOD1	233	93.7%
Model 2-LOD2	125	98.1%
LOD1	62	94.5%
Model 3-LOD2	1024	98.7%
LOD1	513	95.7%
Model 4-LOD2	1060	98.2%
LOD1	530	96.2%

IV. DISCUSSION AND CONCLUSIONS

In this paper, a texture feature preserving building model quantitative simplification approach is proposed. The structural features of buildings are initially extracted, and a robust feature-constrained quantitative simplification algorithm is used to generate multi-resolution 3D building models. Second, the textures of new triangles are generated using the texture synthesis method to maintain the surface appearance of the building. Third, a post-processing algorithm is used to improve the simplification and visualization. The experimental results show that the proposed method can be adapted to models of both buildings and landmarks. It can be applied to both large-scale visualization of building models and the production of LOD models.

Although our method has been shown to be effective through application to a large number of manually produced

building models, the method can still be improved. For example, it is primarily intended for use with building models that have regularly or irregularly shaped surfaces, and it has not been validated for component-based 3D building models (Zhao, Zhu, Du, Feng, & Zhang, 2012). On the other hand, many small textures are generated during the process of building simplification, and each triangle corresponds to a small patch with a given texture (Liu et al., 2017).

ACKNOWLEDGMENT

This research is supported the National Key Technology R&D Program of China (2015BAJ06B01), and Research projects of Surveying and the Fundamental Research Project of the Chinese Academy of Surveying and Mapping (7771720)

REFERENCES

- [1] Ehlers, Manfred. (2014). Advancing Digital Earth-Beyond the Next Generation. *International Journal of Digital Earth*, 7(1), 3-16.
- [2] Goodchild, Michael F, Guo, Huadong, Annoni, Alessandro, Bian, Ling, de Bie, Kees, Campbell, Frederick, Jackson, Davina. (2012). Next-generation digital earth. *Proceedings of the National Academy of Sciences*, 109(28), 11088-11094.
- [3] Gröger, Gerhard, & Plümer, Lutz. (2012). CityGML—Interoperable semantic 3D city models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 71, 12-33.
- [4] Hong, Sungchul, Jung, Jaehoon, Kim, Sangmin, Cho, Hyoungsig, Lee, Jeongho, & Heo, Joon. (2015). Semi-automated approach to indoor mapping for 3D as-built building information modeling. *Computers, Environment and Urban Systems*, 51(0), 34-46. doi: <http://dx.doi.org/10.1016/j.compenvurbsys.2015.01.005>
- [5] Jiang, Shaobo, Yang, Bisheng, & Sun, Xuan. (2011). *Multi-resolution representation of 3D complex building models with features preservation*. Paper presented at the 2011 19th International Conference on Geoinformatics.
- [6] Kolbe, ThomasH. (2009). Representing and Exchanging 3D City Models with CityGML. In J. Lee & S. Zlatanova (Eds.), *3D Geo-Information Sciences* (pp. 15-31): Springer Berlin Heidelberg.
- [7] Liu, Po, Gong, Jianhua, & Yu, Miao. (2014). Graphics processing unit-based dynamic volume rendering for typhoons on a virtual globe. *International Journal of Digital Earth*, 8(6), 431-450. doi: [10.1080/17538947.2014.915349](http://dx.doi.org/10.1080/17538947.2014.915349)
- [8] Liu, Po, Gong, Jianhua, & Yu, Miao. (2015). Visualizing and analyzing dynamic meteorological data with virtual globes: A case study of tropical cyclones. *Environmental Modelling & Software*, 64(0), 80-93. doi: <http://dx.doi.org/10.1016/j.envsoft.2014.11.014>
- [9] Liu, Po, Li, Chengming, Li, Fei, Liu, Po, Li, Chengming, & Li, Fei. (2017). Texture-Cognition-Based 3D Building Model Generalization. *International Journal of Geo-Information*, 6(9), 260.
- [10] Mao, Bo, & Ban, Yifang. (2013). Generalization of 3D building texture using image compression and multiple representation data structure. *Isprs Journal of Photogrammetry & Remote Sensing*, 79(1), 68-79.
- [11] Meng, L., & Forberg, A. (2007). 3D building generalisation. *Challenges in the portrayal of geographic information*. Amsterdam, The Netherlands: Elsevier Science.
- [12] Rau, Jiann-Yeou, Chen, Liang-Chien, Tsai, Fuan, Hsiao, Kuo-Hsin, & Hsu, Wei-Chen. (2006). Lod generation for 3d polyhedral building model *Advances in Image and Video Technology* (pp. 44-53): Springer.
- [13] Regnaud, Nicolas. (2001). Contextual building typification in automated map generalization. *Algorithmica*, 30(2), 312-333.
- [14] Remco, Chang, Thomas, Butkiewicz, Caroline, Ziemkiewicz, Zachary, Wartell, Nancy, Pollard, & William, Ribarsky. (2006). Hierarchical simplification of city models to maintain urban legibility: ACM Press.
- [15] Tan Kim, Heok, & Daman, D. (2004, 26-29 July). *A review on level of detail*. Paper presented at the Computer Graphics, Imaging and

- Visualization, 2004. CGIV 2004. Proceedings. International Conference on.
- [16] Zhao, Junqiao, Zhu, Qing, Du, Zhiqiang, Feng, Tiantian, & Zhang, Yeting. (2012). Mathematical morphology-based generalization of complex 3D building models incorporating semantic relationships. *ISPRS Journal of Photogrammetry and Remote Sensing*, 68(1), 95-111.