Postprint of a Simplification Method for 3D Building Models on Mobile Devices

Authors: royal authority, Song Chunhua, Niu Baoning

Date: 2018-05-18T00:00:00+00:00

Abstract

Currently, model simplification algorithms for individual buildings primarily focus on simplifying wall surfaces and roof-eave structures in isolation. However, for building groups, these methods overlook the characteristics of adjacent buildings and their interconnection relationships with surrounding features. To address these issues, this paper proposes an automatic model simplification method that preserves the structural characteristics of building groups by leveraging roof structural features. Buildings are segmented into distinct structural components through the adjacency relationships of their floor plans; roof features are identified, and a top-down projection method is employed to reconstruct building models, with the various parts finally being combined to form a simplified model. Experimental results demonstrate that, on the basis of effectively retaining top-floor and wall features, this simplification method can fully exploit the model' s topological structure, describe the model with reduced data volume, and generate models with accuracy suitable for mobile application requirements.

Full Text

Abstract

Current model simplification algorithms for individual buildings primarily focus on simplifying walls and roof eave structures separately. However, for building groups, these methods neglect the characteristics of adjacent buildings and their interconnected relationships with surrounding features. To address these issues, this paper proposes a model simplification method that automatically preserves the structural characteristics of building groups by leveraging roof structure features. The method divides buildings into different structural parts based on the adjacency relationships of their footprint plans, reconstructs the building model using a top-down projection approach by analyzing roof features, and finally combines the parts to form a simplified model. Experiments demonstrate that this simplification method can effectively utilize model topology to describe the model with minimal data while preserving roof and wall features, generating precision models suitable for mobile terminal applications.

Keywords: 3D model simplification; building model reconstruction; roof eave structures; mobile terminals

0 Introduction

With the advancement of spatial information technology and computer graphics, three-dimensional building models have been widely applied in various fields such as urban planning, personalized tourism, real estate management, cultural heritage preservation, and navigation. Realism and efficiency are the two most important metrics for evaluating the quality of 3D city models. Existing 3D modeling and rendering technologies can realistically present real-life scenes but require substantial computational resources, including CPU, storage, and network bandwidth. Mobile terminals, however, are characterized by limited computational resources: fine-grained models contain large data volumes that occupy significant storage space; extensive computations rapidly consume battery power, resulting in short battery life; and massive data transmission consumes network traffic, increasing usage costs. These factors severely impact real-time interactive operations and applications of 3D building models on mobile terminals.

Currently, fine-grained 3D building models can be efficiently displayed on PC platforms. However, due to resource constraints on mobile terminals, models of the same level of detail cannot be directly displayed. Three measures are typically adopted for 3D building model applications on mobile devices: (a) using low-precision geometric descriptions for all buildings; (b) using high-precision geometric descriptions for landmark buildings and low-precision descriptions for others; and (c) using high-precision geometric descriptions with texture mapping for landmark buildings on maps, while employing 2D representations for remaining buildings. Current research primarily focuses on reducing building model data volume and optimizing real-time rendering to balance model precision and mobile device performance. Consequently, constructing multi-detail-level models for complex 3D buildings has become an inevitable choice [1]. Additionally, numerous studies have addressed simplification algorithms for individual building models, while research on building group models remains relatively scarce. Existing simplification algorithms concentrate on overall building model simplification, simultaneously considering roofs, walls, and footprint planes. Although these approaches can reduce redundancy to some extent, they still struggle to meet the real-time computational requirements of lightweight mobile terminals. From the perspective of mobile terminal modeling characteristics, it is necessary to further simplify existing 3D building models.

The objective of this paper is to maximally reduce geometric model complexity while preserving model cognition, outputting high-quality simplified models that better adapt to application requirements. The main innovations are as follows: the research focuses on simplifying building group models, fully considering the relationships and interactions between neighboring buildings, performing block decomposition on building groups, utilizing roof structures to construct walls for split individuals, and reconstructing complete models. The method of constructing models using roof structure features can better preserve geometric characteristics without excessive reliance on model topological features, simplifying both the model processing procedure and data transmission volume, thereby improving modeling efficiency for building groups on mobile terminals.

1 Related Work

In traditional 2D maps, operators such as aggregation, typification, displacement, and exaggeration [2] are used to reasonably display map information according to people' s cognition of the objective world under zoom transformations. Three-dimensional building generalization extends from 2D generalization. Kada et al. [3] argue that different applications require different levels of detail, with the main focus being on eliminating secondary information while highlighting important information to improve data transmission and visualization efficiency. For 3D building generalization, Biljecki et al. [4] consider generalization as a process of generating a series of coarse-to-fine level-of-detail (LoD) 3D models, during which the geometric and semantic attributes of individual building objects as well as relationships between buildings must be simplified, aggregated, and typified.

To fully utilize existing 3D building model data in digital cities, it is necessary to reduce redundancy while preserving model appearance features and generate multi-LoD models to meet the visualization requirements of 3D urban navigation systems. CityGML is an open format for 3D model data exchange and storage that defines different hierarchical representations (LoD0-LoD4) for the same building model to meet practical needs. [Figure 1: see original paper] illustrates different LoDs for the same building model. LoD0 represents a 2D plan; LoD1 represents block structures substituting for 3D building models without roofs; LoD2 models include height information and roof structure features; LoD3 adds more feature information such as windows and doors on top of LoD2; and LoD4 includes detailed building information including interior decorations and furniture, with data obtainable through Building Information Models (BIM) and radar [5,6].

Liu Bo et al. [7] proposed a method for constructing individual building geometry models using aerial photographs and ground images to build geometry and textures in three layers (upper, middle, lower), generating model libraries. This approach is suitable for constructing model libraries of buildings with relatively fixed styles, such as residential complexes, which can then be repeatedly called upon. Baig et al. [8] employed edge length threshold constraints on edges, curves, and corners in plan views to extract LoD1 models from LoD3 building models, applicable only to flat-roof buildings with regular structures. Fan et al. [9] proposed a method for extracting low-precision models from high-precision ones, sequentially simplifying footprint outlines and roof eave structures, then connecting footprint edges with eave structures to reconstruct 3D building models. Mao et al. [10] proposed a dynamic typification method based on minimum spanning trees, clustering buildings around a center, grouping buildings according to subtree spatial patterns, and finally performing 3D typification generalization similar to line simplification. These methods all simplify footprint plans based on predefined geometric length thresholds without considering adjacency relationships between buildings, making them suitable only for individual buildings.

For building group generalization, Glander [11] proposed cell-based clustering and partitioning for building merging. Ellul et al. [12] applied geometric length threshold simplification to buildings in 3D scenes, with subsequent research utilizing topological data structures to minimize the description of individual building models and reduce geometric modeling data volume. Although these methods can significantly reduce data volume and improve performance, the building models formed by extruding footprint plans are 2.5D block buildings lacking roof and wall information, resulting in poor realism. Sun et al. [13] proposed a building structure preservation algorithm that generates voxel-based models, performs convex decomposition using boundary block center points, simplifies convex blocks, and merges all blocks to reconstruct simplified models. However, this approach struggles to achieve ideal structural segmentation for buildings with complex and rich detail features, making it unsuitable for diverse building styles. Li et al. [14] used concave-convex analysis to extract embedded, compositional, and connected structures, splitting buildings and establishing corresponding rules to simplify respective structural parts. While this algorithm can simplify models and process complex aggregated models, it involves substantial computation, is time-consuming, and can easily lead to discretization distortion of result surfaces. He et al. [15] proposed a height-based splitting simplification method that divides building group models into two categories: those with approximately the same height and those with different heights, simplifying the footprint plans of model groups but without considering roof simplification.

This paper employs scale-space theory that conforms to human visual perception, considering both the perception and cognitive effects of individual buildings and the influence and interaction of neighboring buildings during the simplification process. Whether for flat-roof or sloped-roof building models, construction is performed only through roof eave structures as global features, which can effectively preserve vertical edges, open footprint boundaries, and roof features. This approach both simplifies the model processing procedure and reduces model data transmission volume.

2 3D Building Group Model Simplification Algorithm Design

The 3D building group model simplification algorithm is designed in three stages: preprocessing, building splitting, and simplified roof reconstruction.

In the preprocessing stage, CityGML models are processed to extract point coordinates of roof and wall structures. During the building group splitting stage, initial splitting is performed through footprint plans, and unsplit portions are further divided according to height and roof structure features until no further splitting is possible, dividing the building group into well-structured individuals. In the roof simplification and model reconstruction stage, a roof structure simplification algorithm is applied to simplify the roof planes of split individuals and remove redundant features. After simplifying individual roofs, the individuals are combined by merging adjacent planes and reconstructing to form the simplified building group model.

2.1 CityGML Model Preprocessing

CityGML uses Boundary Representation (BRep) to represent geometric objects in urban models [17], where each building is represented by multiple planes as boundaries. A CityGML-modeled building is essentially a set of plane combinations containing corresponding semantic information. Since CityGML defines multi-level-of-detail descriptions for the same object while maintaining geometric and semantic consistency, this characteristic can be utilized for conversion between different building model levels. Parsing and processing CityGML models is essential for inter-model conversion. CityGML4j is an open-source library for parsing and converting CityGML formats. Following Deng et al.' s [18] top-down approach, the parsing program locates all buildings in the input 3D city model and uses CityGML4j methods to extract building components such as roof structures, wall structures, and openings for subsequent processing.

2.2 Building Group Model Splitting Method

After determining building complexity, complex buildings are divided into different regular sub-building parts. The divided individuals contain geometric information, facilitating extraction of building roof structures, simplification, and final ground projection operations. In GIS (Geographic Information Systems) 2D and 3D spaces, objects are typically represented by peripheral boundary lines consisting of basic geometric primitives including points, lines, and surfaces. In 2D Euclidean space, points are represented by ordered pairs (x, y), while in 3D Euclidean space they are represented by triples (x, y, z). In Euclidean space, they share the commonality that points constitute lines and lines constitute surfaces, differing in whether surfaces can form 3D geometric entities. During splitting, since building groups contain rich structural features, division cannot rely solely on height comparison but must also consider roof features. To achieve better partitioning, this paper proposes a method that uses both footprint plans and roof features to determine adjacency relationships for building splitting, enabling precise division and avoiding errors. The specific splitting process is described in Algorithm 1.

2.2.1 Building Connectivity Determining building adjacency relationships in model splitting involves two steps: first, establishing planar connectivity, and second, determining 3D connectivity through height and roof structure.

a) **Determining Planar Connectivity**: If primitive is part of another primitive , then is attached to . If and share lower-dimensional primitives, the two primitives are adjacent. If the shared primitive contains only a single point, the two primitives are weakly connected; if it contains at least one line segment, they are strongly connected. Weak connections are collinear in 3D, while strong connections are coplanar. Weakly connected parts are split directly, while strongly connected parts require re-evaluation using height and roof features.

b) **Determining 3D Connectivity**: Based on footprint plans, building groups with different heights are split into the tallest building unit and adjacent sub-parts, with the same cyclic operation applied to sub-parts until no further splitting is possible. During execution, a height threshold is set, and splitting occurs only when relative height differences exceed this threshold to avoid excessive refinement. For building units with heights below the threshold, roof features are determined, and the principle of merging is applied to buildings with identical roof types, combining them into a whole; otherwise, splitting continues. The height of the merged new individual is determined by the heights and planar areas of the original individuals, where h(b) is the height of individual b and area(b) is the area of individual b . The average height is calculated based on each individual b composing the new individual:

$$h = \frac{\sum_{i=1}^{n} h(b_i) \times \text{area}(b_i)}{\sum_{i=1}^{n} \text{area}(b_i)}$$

[Figure 3: see original paper] respectively shows the footprint plan, 3D relationships, and boundary representation of three adjacent buildings.

2.2.2 Distinguishing Flat and Sloped Roof Features Building roof features are determined as follows: Currently, most basic urban buildings can be categorized into three types based on roof characteristics: flat-roof block buildings, gable-roof buildings, and double-gable-roof buildings. The primary distinction in roof type lies in whether the roof-forming planes are inclined; no inclination indicates a flat-roof building. Based on practical experience, buildings with roof plane tilt angles between 0° and 90° are classified as sloped-roof buildings. Tilt angles can be solved mathematically: assuming plane F is A x + B y + C z + D = 0, where i represents the i-th plane of the building; D represents the shortest distance from the plane to the origin coordinate system; [A , B , C] represents the normal vector of plane F , the formula for calculating the angle between planes F and F is:

$$\cos\theta = \frac{A_iA_j + B_iB_j + C_iC_j}{\sqrt{A_i^2 + B_i^2 + C_i^2} \times \sqrt{A_j^2 + B_j^2 + C_j^2}}$$

2.3 Simplifying Roofs and Reconstructing Simplified Building Groups

Roof structures are typically composed of planes with different normal vectors and tilt angles, forming aesthetically pleasing shapes that present three-dimensional effects in 3D space. Although 2D simplification is relatively easier than 3D simplification, projecting 3D planes onto 2D space introduces distortions such as changes in edge lengths and angles, missing relative angles due to projection of adjacent polygon planes, and difficulties in restoring simplified planes. Therefore, roof structure simplification is performed in 3D space. Following Fan et al.' s [9] simplification method, unimportant curves, corners, and edges are removed in 3D space to eliminate redundant features. [Figure 4: see original paper] illustrates the roof structure simplification and merging process.

The roof structure simplification algorithm includes the following steps: a) Eliminate unimportant structural parts such as chimneys and skylights; b) Simplify roof polygons according to planar groups; c) Merge adjacent planes after simplification (coplanar and quasi-coplanar planes).

After simplifying the roof structure, walls are formed by projecting roof edge inflection points downward to reconstruct the building. The simplification and reconstruction process is implemented through Algorithm 2. Walls can be viewed as polygons with multiple vertices. By projecting roof edge inflection points to the ground and intersecting with the ground plane, a complete building individual without gaps between roof and walls is reconstructed. Finally, these individuals are combined into an assembly through union operations. The specific operation involves removing coplanar faces between individuals and merging retained faces to form a building entity satisfying topological structure and approximating the original building model.

Algorithm 1: Building Group Splitting Algorithm

```
Input: Point set (PSet) and edge set (ESet) of all 2D footprint plans, footprint plan colle
Output: 3D building group block partition collection (PartSet).

procedure Segmentation(B-rep)
    PartSet = []  // Store result set
    for all e in ESet
        for all footprint in D do
            if only one footprint contains e
                insert(PartSet)  // Insert into result set
```

```
            else if (isSameHeight(D) && isSimilarBuildings(D))
                // Single-story buildings containing edge e have similar height and roof fea
                new_footprint = [footprint1, footprint2]  // Merge qualifying footprint pla
            else
                Respectively inserted into the PartSet  // Insert separately into PartSet
    return PartSet
end procedure
```

Algorithm 2: Building Simplification and Reconstruction Algorithm

```
Input: Building group partitioned block structure collection PartSet.
Output: Simplified building group model.

procedure Reconstruction(PartSet)
    results = []
    parts = PartSet  // Assign initial value from partitioned block collection
    while notEmpty(parts) do
        p = delete(parts)  // Select block from parts collection
        parts_roof = selectTop(p, PartSet)  // Select roof structure from p
        parts = generateProto(parts_roof)  // Generate individual through roof structure
        insert(results, parts)  // Insert into results
    return results
end procedure
```

3 Experimental Design, Results, and Analysis

The proposed algorithm was implemented in MATLAB on an experimental platform with an Intel(R) Core(TM) i5-4670 CPU @ 3.4 GHz processor, 4 GB RAM, 1 TB hard disk, and Windows 7 operating system. Some experimental data can be obtained from the official CityGML website.

3.1 Simplification Experiment Design

The simplification process is illustrated in [Figure 5: see original paper], which consists of three main modules: data preprocessing, model simplification and reconstruction, and data conversion to CityGML. In the data preprocessing module, since original CityGML building data models cannot be directly loaded by MATLAB, only building shell vertices are extracted for subsequent processing. Vertex sets of building shell parts are extracted from CityGML datasets and divided into two groups: roof structure parts and wall structure parts. In the simplification and reconstruction module, building adjacency relationships are determined through footprint plan adjacency, buildings are split, roof structures of split individuals are simplified, and finally reconstructed to form simplified building groups. In the data conversion module, simplified models are converted into CityGML format readable by LandXplorer CityGML Viewer 2009.

3.2 Simplification Results for Different Roof Building Models

Simplification runtime and data compression are determined by the geometric complexity of building structures. To verify algorithm effectiveness, flat-roof, single-gable, combined-gable, and mixed building models were tested. Additionally, comparative experiments were conducted with three classical algorithms.

Scheme 1 is a combined flat-roof building composed of two flat-roof buildings with different heights. Initially, the footprint plan and roof structure graph edges with local detail features have identical contours. [Figure 6: see original paper] shows the simplification process. Roof structure edges shorter than the threshold length are removed, along with structural parts smaller than the threshold on roof planes. Comparing before and after simplification, the retained roof plan is very similar to the original roof plan contour. The model generated using top-down projection is close to the original model.

Scheme 2 divides building models into single-gable and combined-gable buildings. [Figure 7: see original paper] shows a single-gable building model. The simplification process does not require splitting; the gable building model simplification method is applied directly to generate the simplified model. For combined-gable building models, [Figure 8: see original paper] shows a building model whose roof structure is formed by different gable combinations. The simplification process must consider whether to split. Since the combined model' s roof structure features are gabled with approximately the same height, no splitting is performed, and the combination is processed for simplification. The gable building model simplification algorithm first simplifies then projects to generate the simplified combined gable building model.

Scheme 3 is a mixed building group model containing both gable and flat roofs with different building heights. The simplification process must first comprehensively consider height and roof type to determine splitting, then simplify individuals, and finally merge simplified individuals. This avoids irregular top structure extraction that would complicate the simplification process. The simplification strictly follows the proposed algorithm, with the specific process shown in [Figure 9: see original paper].

When building models contain roof structure features, the simplification process must account for roof characteristics. Liu Bo et al.' s [7] building model layered simplification method comprehensively divides building models into three layers (upper, middle, lower) for separate simplification, suitable for urban residential building models. The upper layer' s top structure features are not simplified; the middle layer, due to structural similarity, is constructed by establishing a single-layer model and repeatedly stacking it; the lower layer contains more detail features and is simplified separately. [Figure 10: see original paper] shows building models generated using Liu' s simplification method.

Fan et al.' s [9] 3D building simplification method consists of three steps: wall simplification, roof eave structure generalization, and building model reconstruction. The wall simplification process projects walls onto the ground to form 2D footprint plans, removes redundant edges and corners to form simplified footprint plans; the roof structure simplification process adopts a similar strategy for polygon plane generalization; the reconstruction process calculates intersection lines through topological relationships between intersection points and roof polygon planes, removes gaps between walls and roofs, and forms a simplified 3D building model. [Figure 11: see original paper] shows building models generated using Fan' s simplification method.

Ellul et al.' s [12] 3D building simplification method primarily applies 2D map generalization methods to footprint plans. By applying geometric length threshold judgments to building footprint plans, redundant edges are removed to form simplified 2D footprint plans, which are then extruded to form 3D block building models containing only flat-roof structure features. [Figure 12: see original paper] shows building models generated using Ellul' s simplification method.

3.3 Experimental Results Analysis

Comparative experimental results demonstrate that all simplification algorithms can simplify 3D building models to some degree. When processing flat-roof building models, the four methods produce essentially identical models. When processing gable, combined-gable, and mixed building models, differences in simplified models mainly appear in roof structure features, though overall effects are similar. Comparing the four simplification algorithms, Fan' s method requires considering footprint plan simplification and handling intersections between footprint plans and top structures, while our method does not require these considerations, reducing runtime and improving efficiency. Ellul' s method achieves the highest data compression but lacks roof features, causing excessive detail loss. Liu' s layered simplification method does not perform roof feature simplification, resulting in overly redundant roof details. Additionally, the proposed building group simplification algorithm can obtain semantically and geometrically well-formed building individuals through block partitioning, facilitating footprint projection extraction and simplification operations to generate refined building group models. Whether for individual or combined models, our algorithm can generate streamlined building models suitable for mobile terminal loading and rendering.

compares data volumes between original and simplified models. In 3D building models, complex roof structures and wall detail features constitute a large portion of data volume, which is the primary target of simplification. Therefore, different levels of model detail result in different compression rates. For simple flat-roof and gable building models, since footprint plans and roof structure outlines are approximately identical, all four methods produce similar models with identical data compression. For gable and mixed building models, the four algorithms produce different simplification degrees.

[Figure 13: see original paper] compares vertex counts across different building model simplifications, while [Figure 14: see original paper] compares compression rates. Analysis of [Figure 13: see original paper] and [Figure 14: see original paper] across different experimental schemes shows that Liu' s method produces the lowest simplification degree, relatively completely preserving original model features with the lowest compression rate of 0% and highest of 18.2%, failing to meet mobile terminal precision requirements. Comparing the other three algorithms, Ellul' s method achieves high data compression at the cost of visual effects. Simplified models lack roof structure features, making them difficult to identify clearly. While preserving model contour features, Fan' s method achieves good simplification effects. Compared with Fan' s method, our method achieves higher compression rates, with a minimum of 25.6% (9.3% improvement over Fan' s method) and maximum of 61.2% (20.3% improvement over Fan' s method), better meeting application requirements for describing entire building models with minimal data.

This result stems from: (a) splitting complex building group models into simple individuals for easier simplification; and (b) generating building bodies through top-down projection from roof structures, simplifying overall model structure.

Experimental verification using various typical 3D building models demonstrates that simplifying roof structures to a certain degree while preserving basic shapes and reconstructing building models can generate models matching original models. Shape similarity measurement evaluation reveals that, under the same vertex count, our method produces models with higher similarity that can be more easily matched with original models, meeting mobile terminal model requirements. Additionally, during data transmission, only roof structure vertex coordinates need to be transmitted from server to client, reducing data transmission volume and network load.

4 Conclusion

This paper proposes a simplification algorithm primarily targeting 3D building groups, aiming to efficiently and rapidly extract low-precision building models from high-precision models to replace original models with building shell models. Previous methods separately simplified base structures and roof structures before final combination, resulting in cumbersome, time-consuming processes applicable only to individual buildings. Our method fully considers human perception and cognition of building features while accounting for interactions between neighboring buildings. By proposing a split-first approach followed by reconstruction based on roof structure features, the method can handle both individual buildings and building groups. Using roof structure features to construct models better preserves geometric features and avoids excessive dependence on model topological characteristics. This approach both simplifies the model processing procedure and reduces model data transmission volume, significantly improving simplification efficiency and reducing topological constraints.

Various typical 3D building model simplification experiments demonstrate that the method can effectively simplify individual or building group models containing flat or gable roofs while reducing data volume. Without affecting visual effects, compared with other simplification methods, our method achieves better data compression. Overall, the proposed method significantly improves simplification robustness, reduces model topological constraints, and outputs high-quality simplified models.

The method still has limitations: it only addresses 3D building geometric models and cannot effectively handle complex building models with special structures such as ancient pagodas, archways, and other historic buildings or models with other roof structure types. Future research will consider simplifying such building models and utilizing texture reproduction technology to enhance 3D scene realism, achieving a balance between efficiency and realism in 3D city construction.

References

[1] Liu Lei. Research on comprehensive algorithms for simplified 3D city models [D]. Nanjing: Nanjing University, 2013.

[2] Zhao Junqiao. Automatic simplification method for multi-level details of complex 3D building models [J]. Acta Geodaetica et Cartographica Sinica, 2013, 42(1): 156-156.

[3] Kada M, Wichmann A, Filippovska Y, et al. Animation strategies for smooth transformations between discrete lods of 3D building models [J]. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2016, 41(B2): 413-420.

[4] Biljecki F, Ledoux H, Stoter J, et al. Formalisation of the level of detail in 3D city modelling [J]. Computers Environment & Urban Systems, 2014, 48(16): 1-15.

[5] Zhu Liang, Deng Fei. Research on BIM and 3D GIS integration method based on semantic mapping [J]. Journal of Geomatics, 2016, 41(3): 16-19.

[6] Cheng J C P, Deng Y, Anumba C. Mapping BIM schema and 3D GIS schema semi-automatically utilizing linguistic and text mining techniques [J]. Electronic Journal of Information Technology in Construction, 2015, 20: 211-228.

[7] Liu Bo, Wang Zhangye, Wang Liying, et al. Efficient modeling and real-time rendering of large-scale urban scenes [J]. Journal of Computer-Aided Design & Computer Graphics, 2008, 20(9): 1153-1162.

[8] Baig S U, Rahman A A. A three-step strategy for generalization of 3D building models based on CityGML specifications [J]. GeoJournal, 2013, 78(6): 1013-1020.

[9] Fan Hongchao, Meng L Q. A three-step approach of simplifying 3D buildings modeled by CityGML [J]. International Journal of Geographical Information

Science, 2012, 26(6): 1091-1107.

[10] Mao Bo, Harrie L, Ban Yifang. Detection and typification of linear structures for dynamic visualization of 3D city models [J]. Computers Environment & Urban Systems, 2012, 36(3): 233-244.

[11] Glander T. Multi-scale representations of virtual 3D city models [D]. Potsdam: Universität Potsdam, 2013.

[12] Ellul C, Joubran J A. Investigating approaches to improving rendering performance of 3D city models on mobile devices [J]. Geo-spatial Information Science, 2014, 17(2): 73-84.

[13] Sun Xuan, Yang Bisheng, Li Qingquan, et al. Linear geometric coding of complicated 3D building models [J]. Geomatics & Information Science of Wuhan University, 2013, 38(2): 225-229.

[14] Li Qingquan, Sun Xuan, Yang Bisheng, et al. Geometric structure simplification of 3D building models [J]. ISPRS Journal of Photogrammetry & Remote Sensing, 2013, 84(10): 100-113.

[15] He Shuang, Moreau G, Martin J Y. Footprint-based 3D generalization of building groups for virtual city visualization [C]//Proc of the 4th International Conference on Advanced Geographic Information Systems, Applications, and Services. Valencia: GEOProcessing, 2012: 177-182.

[16] Fan Hongchao, Mao Bo, Shen Jie, et al. Shell model representation as a substitute of LoD3 for 3D modeling in CityGML [J]. Geo-spatial Information Science, 2011, 14(2): 79-84.

[17] Gröger G, Plümer L. CityGML: interoperable semantic 3D city models [J]. ISPRS Journal of Photogrammetry & Remote Sensing, 2012, 71: 12-33.

[18] Deng Yichuan, Cheng J C P, Anumba C. Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison [J]. Automation in Construction, 2016, 67: 1-21.

[19] Xu Yongyang, Xie Zhong, Chen Zhanglong, et al. Shape similarity measurement model for holed polygons based on position graphs and Fourier descriptors [J]. International Journal of Geographical Information Science, 2017, 31(2): 253-279.

Note: Figure translations are in progress. See original paper for figures.