International Journal of Applied Earth Observation and Geoinformation 35 (2015) 247–258

Contents lists available at ScienceDirect

International Journal of Applied Earth Observation and Geoinformation

journal homepage: www.elsevier.com/locate/jag

A Gestalt rules and graph-cut-based simplification framework for urban building models

Yuebin Wang[a], Liqiang Zhang[a,*], P. Takis Mathiopoulos[b,c], Hao Deng[a]

[a] State Key Laboratory of Remote Sensing Science, Beijing Normal University, Beijing 100875, China
[b] Institute for Astronomy Astrophysics Space Applications and Remote Sensing, National Observatory of Athens Metaxa and V. Pavlou, Penteli, Athens 145 36, Greece
[c] Department of Informatics and Telecommunications, National and Kapodestrian University of Athens, Athens 157 84, Greece

ARTICLE INFO

Article history:
Received 2 October 2013
Accepted 18 September 2014
Available online 8 October 2014

Keywords:
Urban buildings
Gestalt principles
Optimization
SceneTree

ABSTRACT

To visualize large urban models efficiently, this paper presents a framework for generalizing urban building footprints and facade textures by using multiple Gestalt rules and a graph-cut-based energy function. First, an urban scene is divided into different blocks by main road networks. In each block, the building footprints are partitioned into potential Gestalt groups. A footprint may satisfy several Gestalt principles. We employ the graph-cut-based optimization function to obtain a consistent segmentation of the buildings into optimal Gestalt groups with minimal energy. The building footprints in each Gestalt group are aggregated into different levels of detail (LODs). Building facade textures are also abstracted and simplified into multiple LODs using the same approach as the building footprint simplification. An effective data structure termed SceneTree is introduced to manage these aggregated building footprints and facade textures. Combined with the parallelization scheme, the rendering efficiency of large-scale urban buildings is improved. Compared with other methods, our presented method can efficiently visualize large urban models and maintain the city's image.

Introduction

Real 3D city models have wide applications in urban planning, mapping and virtual tourism. With the rapid development of photogrammetry, computer vision, scanners and 3D modeling technologies, it is currently possible to construct detailed 3D city models in a practical and cost-efficient manner. This is possibly the main reason that 3D digital urban models have become so increasingly popular. These developments are changing the way the user conceives 3D data. Typically, a large city often contains hundreds of millions of buildings, and it is difficult to view digitized city models in real time. Therefore, the visualization of large-scale 3D city models for a variety of professional and mass-market services has received significant attention in the photogrammetry and computer graphics communities. To make such services appealing to a large audience, these 3D models should reach a sufficient level of realism and accuracy. Many solutions have been proposed to generate 3D models of huge urban environments (Royan et al., 2007). For example, Sheppard and Cizek (2009) proposed criteria for evaluating landscape visualizations under the categories of (i) accuracy, (ii) representativeness, (iii) visual clarity, (iv) interest, (v) legitimacy, (vi) access to visual information, and (vii) framing and presentation. To visualize a large-scale city efficiently, some generalization algorithms (such as Forberg, 2007; Zhang et al., 2012) are used to simplify urban building models.

To help people recognize a city more effectively and maintain spatial coherence, Gestalt principles are introduced to identify the distribution pattern of urban buildings. In this paper, we apply Gestalt rules and graph-cuts to cluster similar buildings into the same group. Then, the building footprints and façade textures are aggregated, and the whole urban models can be generalized into multiple levels. To obtain a good generalized representation of a city that abides by the Gestalt principles, the spatial relations, including distance, orientation, similarity and continuity, should be taken into account when building models are clustered. To reduce the load time, a multiple representation data structure termed SceneTree is proposed to store hierarchical models of the aggregated building footprints and facade textures. When these buildings are viewed, different levels of the building models are retrieved from SceneTree and rendered by the parallelization scheme. Compared with other methods, the novelty of our approach is that we implement an optimization framework for generalizing building

* Corresponding author.
E-mail address: zhanglq@bnu.edu.cn (L. Zhang).

http://dx.doi.org/10.1016/j.jag.2014.09.012
0303-2434/

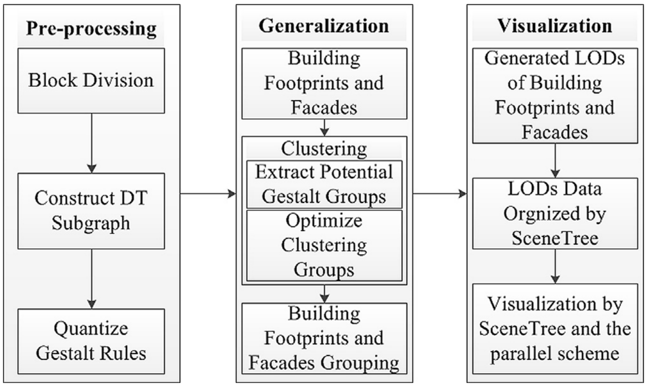

Fig. 1. The framework for urban building simplification and visualization.

footprints and facade textures simultaneously. In this framework, a graph-cut-based optimization method is employed to solve the conflicts that a footprint or facade texture may satisfy with several Gestalt principles, and SceneTree is created to store the generalized footprints and textures at deferent levels.

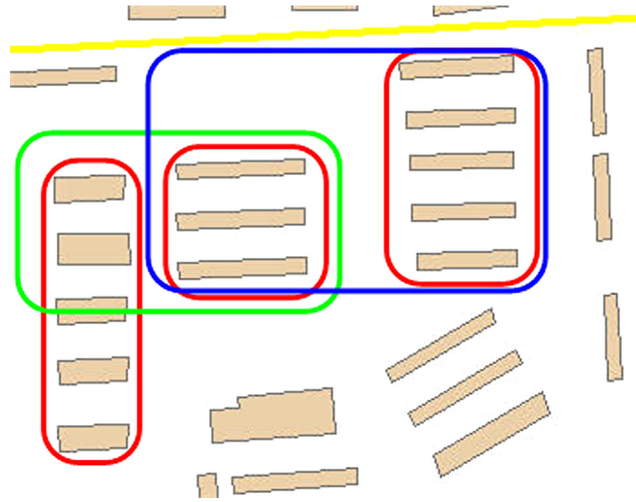

Fig. 2. Building clustering different Gestalt groups. The red wireframe represents the regularity group, the green wireframe represents the proximity group, and the blue wireframe represents the similarity group. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Related work

Numerous studies have developed a range of tools and techniques for visualizing city buildings. Additionally, an extensive amount of literature can be found in the related fields of computer vision, object recognition and geometric modeling. However, there remain many challenges in applying landscape visualization techniques for effectively communicating processes and changes in building landscapes. In this section, previous generalization results of 3D building models are summarized.

Geometric-based generalization of urban building models

Most of the algorithms used for mesh simplification and discrete levels of detail (LOD) (Luebke, 2001) work well for simplifying single objects with a large number of polygons. However, these algorithms are ineffective for rendering large collections of simple models (Chang et al., 2008). For rendering city-sized buildings, Kada (2006) presented an approach for generalizing 3D building models. His approach remodels buildings with only a few optimized planes. It simplifies roof geometry while preserving their coplanar, parallel and orthogonal features. In the approach of Kada and McKinley (2009), a complex building was first divided into cells using the main lines of its footprint. The borders of these cells were then used to form facades to simplify the model. To generalize building roof structures, feature detection was applied explicitly by instancing roof shape primitives for each cell and selecting the cell with the best fit. Kada (2011) extended the building generalization and aggregation approach based on the cell decomposition to use morphological operations on a raster representation of the initially vectorial data. This approach effectively classified cells into two categories: building and non-building cells, as well as simplified building models. However, it was found that repeated execution of morphological operations may change the shape of an object's boundaries. Yang et al. (2011) implemented an approach to generalize and render urban building models in the context of Gestalt psychology and urban legibility. Later Zhang et al. (2013) extended this approach by applying more Gestalt rules to cluster building footprints.

Table 1
The procedure of computing a DT subgraph of buildings in blocks.

Input: The set of the centroids of the building footprints in one block: $C = \{C_i\}$, ε, β, $k := 2$

Output: The DT subgraph $G = \{C, E\}$ of which the nodes are C

begin
- Construct Delaunay triangulation $DT=\{C, E_{DT}\}$ of C
- // Construct $MST=\{C, E_{MST}\}$ of C
- BuildMST(MST)
- $E := E_{MST}$
- $E_{sub} := E_{DT} \backslash E_{MST}$
- **foreach** e_i **in** E_{sub} **do**
  - **if** dist(e_i)<ε **and** $e_i \in \beta$-bone edge **then**
    - $E := E \cup e_i$
    - $E_{sub} := E_{sub} \setminus e_i$
- **foreach** C_i **in** C **do**
  - **if** degree(C_i)=1 **then**
    - E_{Ci}:= relevant edges of C_i in E_{sub}
    - Find the shortest edge $e_{shortest}$ which meets the condition of β-bone edge in E_{Ci}
    - $E := E \cup e_{shortest}$

End