LoD2 Building reconstruction with graph-cut optimisation

Ravi Peters
5 December 2019
2nd International Workshop Point Cloud Processing, Stuttgart
Motivation

Two projects:

1. National semantic 3D model by Dutch Cadastre
   LoD1 → LoD2

2. Noise simulation input data for Dutch governments
   LoD1 → LoD1.3/LoD2
Goals and scope

- Automated LoD2 and LoD1.3
- From classified airborne lidar + cadastral footprints
- Open source implementation

Biljecki et al. (2016)
Goals and scope

- Automated LoD2 and LoD1.3
- From classified airborne lidar + cadastral footprints
- Open source implementation
Requirements

1. Low model complexity, good accuracy
2. Flexible to variety of roof shapes
   a. No orientation constraints
   b. No predetermined roof type library
3. Geometrically valid (ISO19107)
   a. No self-intersections
   b. 2-manifold
   c. Watertight
4. Reasonably robust to occlusion

Constraints

1. 2.5D
2. Piecewise planar
Method
Method overview

1. Plane detection
2. Line detection + regularisation
3. Construct planar arrangement
4. Perform graph-cut optimisation
5. Extrusion from planar arrangement
1 Plane detection

1. Region growing plane segmentation based on
   a. Angle between normals
   b. Distance to region plane

2. Refit region plane as region grows
2 Line detection

1. Boundary lines
   a. Compute $\alpha$-shape of plane segments
   b. Extract boundary
   c. Region growing line segmentation based on distance to region line

2. Compute plane intersection lines
2 Line regularisation

Goal: merge doubly detected lines

1. Cluster lines by angle
2. Cluster lines by distance
3. Keep one line per cluster
3 Construct planar arrangement

Using fully topological data-structure (DCEL)
4 Graph-cut optimisation

\[ E(f) = \sum_{p \in P} D_p(f_p) + \lambda \cdot \sum_{\{p,q\} \in N} V_{p,q}(f_p, f_q) \]

Dual graph of planar arrangement

possible labels:

1.  
2.  
3.
4 Graph-cut optimisation

\[ E(f) = \sum_{p \in P} D_p(f_p) + \lambda \cdot \sum_{\{p, q\} \in N} V_{p,q}(f_p, f_q) \]

Data term:
Volume between candidate planes and 2.5D heightfield of point cloud

Volume wrt each candidate plane
4 Graph-cut optimisation

\[ E(f) = \sum_{p \in P} D_p(f_p) + \lambda \cdot \sum_{\{p,q\} \in N} V_{p,q}(f_p, f_q) \]

Smoothness term:
Length of shared boundary
5 Extrusion

Correct topology guaranteed from planar arrangement

Remove edges between faces with same plane label
Results
Neighbourhood

Input footprints
LoD1.3 reconstruction
Building 1

Planar segments

LoD2 reconstruction
Building 1

Arrangement before graph cut

Arrangement after graph cut
Building 1

Distance from point cloud to reconstructed model [m]
Building 5

Plane segments
Building 5

Boundary of $\alpha$-shape
Building 5

Detected lines
Building 5

Arrangement before graph cut

2.5D heightfield
Building 5

Planar segments
Building 5

Detected lines
Building 5

LoD2 reconstruction
Building 5

Distance from point cloud to reconstructed model [m]
Future work

1. Reduce complexity of planar arrangement prior to graph cut
2. Use facade planes if available
3. Investigate different smoothness terms in optimisation
4. Run it for the whole of the Netherlands

Binary Space Partitioning

Jung et al. (2017)
Source code available

https://github.com/geoflow3d/gfp-building-reconstruction
Thank you!
Heightfield
Graz data

1. 2D input Footprints generated from union of restituted 3D roof-outlines
2. Very high point density compared to previous data I worked with -> makes thing easier!
   a. Plane detection possible with stricter parameters
3. Issues
   a. Classification issues trees/ground points in footprint
   b. Shadow
Step 4: Extrusion of roof parts

Point cloud → Roof planes → Roof edges → 2D arrangement → LoD 1.3 model
Some applications require LoD1.3 (e.g., noise simulation).

1. Big advantage of LoD2 reconstruction: better quality control since model fits better with point cloud.
2. LoD1.3 can easily be generated from LoD2.
Difference LoD2.2—LoD1.3
Difference LoD2.2—LoD1.3
Kadaster point cloud vs AHN3?

- So far the method has been developed with Lidar-based AHN3 point cloud
- Kadaster point cloud is acquired with fundamentally different technique (dense image matching)
  - Differences in update interval, accuracy, occlusion/completeness, vegetation penetration, classification, etc.
  - Comes in different qualities depending on flight (and budget) restrictions
- Building reconstruction results will vary with quality input points!
Kadaster point cloud vs AHN3?

Worst case scenario: TMA zone with 60% overlap
Kadaster point cloud vs AHN3?

**Worst case scenario**: TMA zone with 60% overlap
Work in progress

1. Improve plane detection and line detection
   a. Fast plane detection with Hough transform

2. Further improve graph-cut approach
   a. Improve computation of data-term to deal with overhangs
   b. Investigate less complex decomposition of footprint from detected lines

Binary Space Partitioning

Jung et al. (2017)