A New Approach to Urban Modelling Based on LIDAR

Rebecca (Oi Chi) Tse
rtse@glam.ac.uk

Christopher Gold
christophergold@voronoi.com

Dave Kidner
dbkidner@glam.ac.uk

School of Computing, University of Glamorgan, Pontypridd CF37 1DL, Wales, UK

ABSTRACT

Estimating building forms from LIDAR data has usually been done by attempting to fit standardized building types to the residual data points after an estimated bare earth terrain surface has been removed. We propose an approach based on segmenting the raw data into high and low regions, and then modelling the walls and roofs by extruding the triangulated terrain surface (TIN) using CAD-type Euler operators. The segmentation may be done by the addition of building boundary data to the TIN so as to force triangle edges to match the boundaries, and then using Euler operators to extrude the building, producing vertical walls rather than the more usual sloping walls formed from TIN models alone. If boundary data is not available then an automated segmentation method based on adaptive Voronoi cells may be used, so that each cell contains either high or low LIDAR data, but not both. This prismatic model, with flat-topped cells, approximates the building forms without hypothesizing specific building types. Once the segmentation is achieved, and the walls constructed, we attempt to model the roofs by calculating the eigenvalues and eigenvectors of the vector normals of the TIN sections within the building boundaries. The smallest eigenvalue gives a predicted roof orientation, and the resulting roof profile is then modelled.

Keywords
Urban modelling, LIDAR, Euler operators

1 INTRODUCTION AND PREVIOUS WORK

LIDAR (Light Detection and Ranging) data is widely used to construct 3D terrain models to provide realistic impressions of the urban environment and models of the buildings. The latest airborne laser scanning technology allows the capture of very dense 3D point clouds from the terrain and surface features. The most common method is to remove all the buildings, trees and terrain objects and generate a bare-earth model [Vos03]. Building boundaries are obtained and estimated building forms are reconstructed by CAD software and pasted on top of the bare-earth model. This approach is sufficient for visualization, but inadequate if the resulting model is to be used for analysis:

flood, simulation, trafficability, etc. We therefore wish to integrate our buildings with the terrain surface. A second concern is the source of building outlines: if they are not available from map data they must be extracted from the LIDAR data itself. This is usually imprecise. Suveg and Vosselman [SV04] give an extensive survey of building modelling, but based on photogrammetric data.

Thus, in section 2, we use a traditional method to filter all the terrain objects, generate a bare-earth model and put Ordnance Survey Landline data as the building outlines on the terrain surface. Where map data is not available, building outlines must be extracted directly. Image analysis techniques may be used to identify linear features, but these are difficult to connect to form complete buildings. We describe a new alternative region-based approach where Voronoi cells are adjusted to contain either "building" or "non-building" LIDAR data points.

Section 3 describes how topologically integrated buildings may be extruded from the landscape, based Euler operators. This permits interactive editing of the 3D model [TG01].

Having constructed the basic flat-roofed building, Section 4 describes a method of reconstructing the roof shape from the interior LIDAR data. These points are first triangulated, and the triangle orientations pre-

served. Unlike Hofmann, Maas and Streilein [HMS03], who used slope, azimuth and a distance measure, we project the vector normals of the triangles onto the unit hemisphere, simplifying our clustering algorithms and allowing estimation of the roof axis, if present.

2 Methodologies for Building Reconstruction Using LIDAR data

We mentioned two segmentation methods which are:

- Landline Tracing

- Voronoi City Modelling

2.1 Landline Tracing

We use Ordnance Survey (OS) Landline, original and filtered LIDAR data points to create the urban model. In this paper we use the University of Glamorgan campus as an example. The original and filtered data were captured by Infoterra in March 2003. A filtering algorithm is used to remove all terrain objects, for example, buildings, trees and cars. Morphologic filtering is one of the most common algorithms used to create a bare-earth model, for example slope based filtering [Vos03] and modified slope based filtering [Rog02].

Several steps are used to reconstruct buildings using Landline Tracing. They are:

- Create a TIN model with the filtered LIDAR data

- Add Landline data to the terrain surface using our line tracing algorithm

- Use our point-location algorithm to find out the LIDAR data points for each building and calculate the average heights (z-value) of each building

- Extrude the building using CAD-type Euler Operators. The details will be shown in the next section

There are several steps in the line-tracing algorithm:

1. Insert two points on the terrain surface (Points A and B in figures 1)

2. Check if any previous triangle edge connects these two points

   - Stop if this is true, or
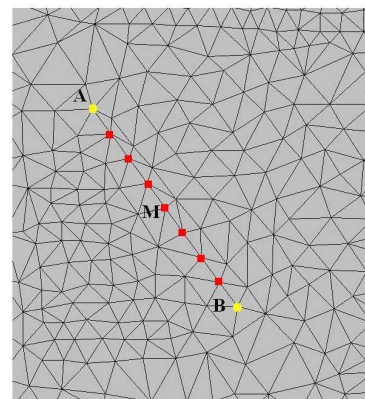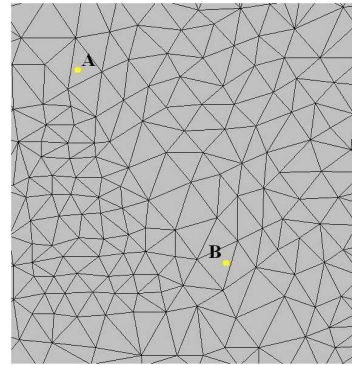   - Insert a point half way between these two points if no edge connects them.

Figure 1: Connects points A and B by adding several (square shaped) points in between

3. Repeat step 2 for each half of the line recursively until points A and B are connected by triangle edges.

We add points on the surface and estimate the height of the points by using the surface interpolation method of [DG02].

We use the point-location algorithm to find the LIDAR data points which are inside the building boundary and to find the average height of each building.

The steps of the point-location algorithm are:

1. Delete all points inside the building boundary

2. Order the vertices of the building boundary in anti-clockwise order

3. Load the LIDAR data point from the file

4. Walk through to locate the triangle which contains the LIDAR data point

5. Check if the order of the vertices are anticlockwise

   - Store the point in a list for further calculation if it is true (point A in figure 2), or
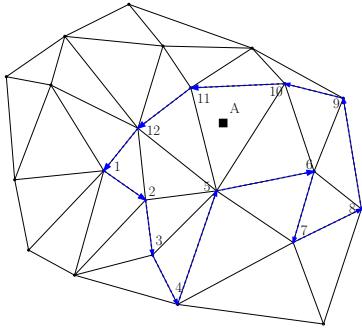
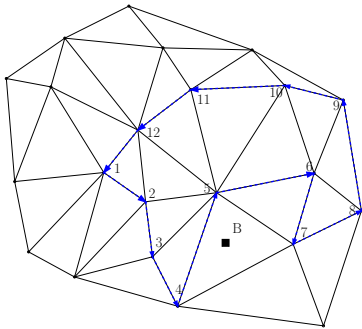Figure 2: The order of the vertices is anticlockwise 5-10-11

Figure 3: The order of the vertices is clockwise 4-5-7

- Do nothing and go to another point if it is not anti-clockwise (point B in figure 3)

6. Calculate the average height in each list for the building height

7. Repeat from step 3 until all the points are searched

We have added the building boundary to the terrain surface. We will show how to use the Euler Operators to extrude buildings in the next section. Figure 4 shows the building boundaries (top) and the extruded buildings (bottom) of the University of Glamorgan campus.
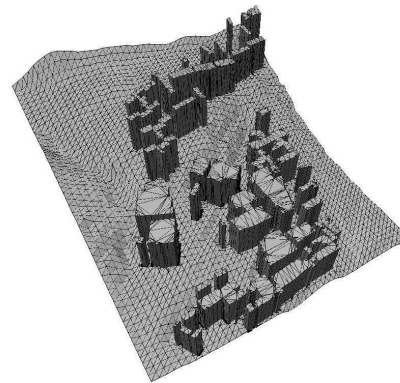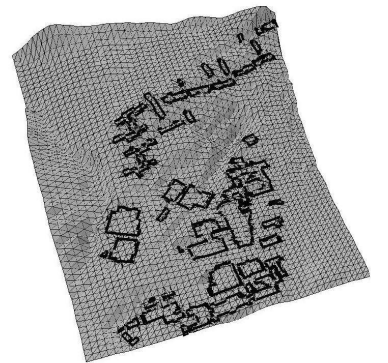
2.2 Voronoi City Modelling

Automatic extraction of building boundaries based on edge detection is difficult. Here we superimpose a coarse layer of random Voronoi cells on the LIDAR data and adjust their positions so that each cell is entirely "inside" or "outside" a building. A subset of the cell boundaries then forms the building boundary.

The steps used in Voronoi City Modelling are:

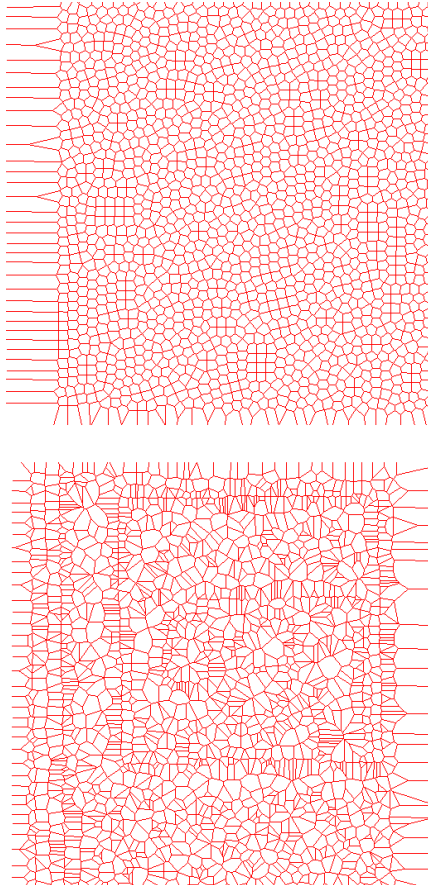1. Sample the raw LIDAR data to get a lower density of data point and create a Voronoi diagram

Figure 4: Building boundaries (top) and the extruded buildings (bottom) of the University of Glamorgan campus

Figure 5: The original (top) and iterated Voronoi Cells in 2D view

Figure 6: The original and iterated Voronoi Cells in 3D view

from the sampled data points.

2. For each Voronoi cell:

3. Compare the Voronoi cell with its neighbours, and calculate the within-cells and between-cells variance of the underlying LIDAR points. Calculate the F-ratio ("between" / "within").

4. Move the cell and leave it in its new location if the F-ratio increased due to the move.

5. Repeat 2 to 4 until all cells are processed.

6. Iterate steps 1 to 5 until the best building shapes show in the 3D view.

Figures 5 shows the original Voronoi Cells (top) and those after a few iterations (bottom) in a 2D view. Figure 6 shows the 3D view where the building outlines have more or less formed.
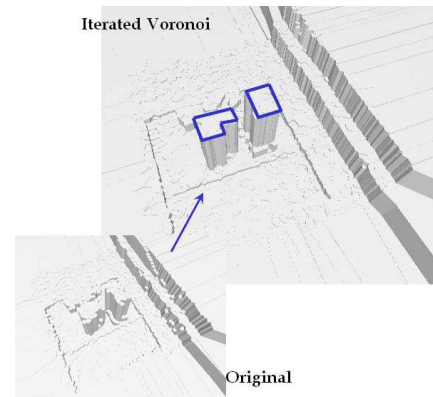
3 Building Extrusion Using Euler Operators

Constructive Solid Geometry (CSG) and VRML are the two common methods for modelling and rendering buildings on the terrain. [Bre99] and [SV04] used CSG to generate a complex building with the Boolean operations of union, intersection and differences. [RB03] used VRML to display the generated buildings. However topological relationships are not kept during the construction. All the buildings are superimposed on the terrain surface without actually connecting to it. If the topological connectivity is preserved, more kinds of spatial analysis can be performed.

We start by creating a 2.5D TIN model and used Euler Operators to extend the model. Using Euler Operators to extrude a building has four stages [TG01]:

1. The Basic Quad-Edge data structure (with only two topological operators) [GS85].

2. The implementation of Euler Operators using the Quad-Edge data structure.

3. The implementation of triangulation models using Euler Operators.

4. Interactive building modification using additional Euler Operators.

3.1 The Basic Quad-Edge Data Structure

We used the Quad-Edge data structure as the basis of our model. More particularly, the Quad-Edge structure was used to implement a set of Euler Operators sufficient for the maintenance of surface triangulations. The Quad-Edge data structure fulfills a sufficient adjacency topology representation according
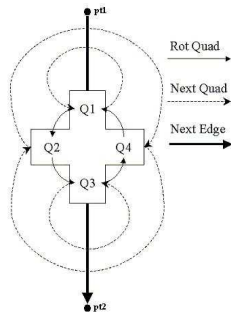
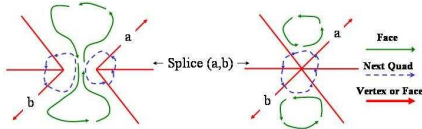Figure 7: Make Edge creates a new independent edge

Figure 8: Splice and its own inverse, either splits a face or merges two faces

to Weiler [Wei88]. "Make-Edge" and "Splice" (figures 7 and 8) are the two simple operations on the Quad-Edge structure, which is formed from four connected "Quads" objects, using the simple implementation of [Gol98].

3.2 Implementation of Euler Operators Using the Quad-Edge Data Structure

Five spanning Euler Operators (plus the Euler-Poincaré formula) suffice to specify the number of elements in any boundary representation model [BHS78]. In TINs there are no loops (holes in individual faces), so these will not be considered. Thus four spanning Euler Operators suffice for TINs with tunnels or bridges. We use "Make Edge Vertex Vertex Face Shell" (MEVVFS) to create an initial shell, and its inverse "Kill Edge Vertex Vertex Face Shell" (KEVVFS) removes it. Figure 9 shows the operator MEVVFS and its inverse.
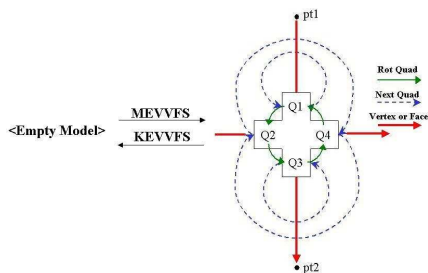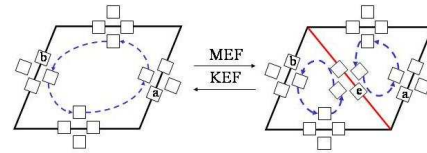

Figure 9: MEVVFS and its inverse KEVVFS
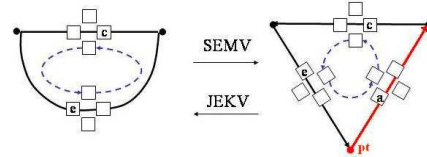

Figure 10: MEF and its inverse KEF

Figure 11: SEMV and its inverse JEKV

In figure 10, "Make Edge Face" (MEF) and its inverse "Kill Edge Face" (KEF) creates and kills an edge and a face. "Split Edge Make Vertex" (SEMV) splits an edge and creates a vertex, and its inverse is "Join Edge Kill Vertex" (JEKV) which show in figure 11

3.3 Implementation of the TIN Model Using Euler Operators

In the TIN model we have three main functions, which are:

- Create the first triangle
- Insert or its inverse delete a point
- Flip an edge

We use MEVVFS, MEF and SEMV to create the first triangle from an empty model in figure 12. The first triangle should be big enough to contain all the data points. JEKV, KEF and KEVVFS are used to kill the first triangle.

Figure 13 shows inserting and its inverse, deleting, a new point which adds three edges and two faces. MEF
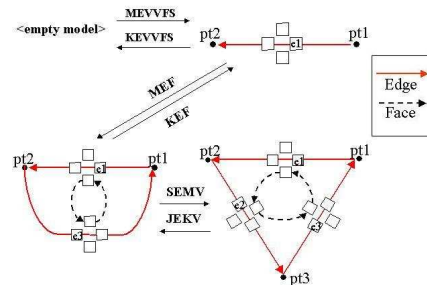

Figure 12: First Triangle should be big enough to contain all data points
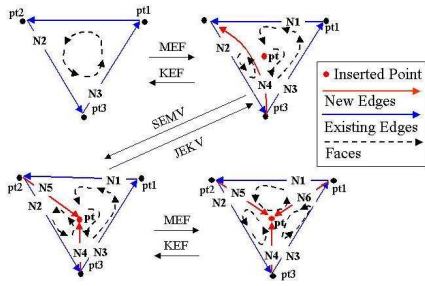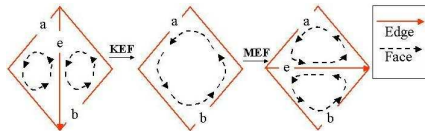
Figure 13: Insert and its inverse delete a point

Figure 14: Flip22

creates an edge N4. SEMV splits an edge N4. In the last step MEF creates a new edge N6. We use KEV and JEKV to delete a point. To delete a point, we use MEF to kill an edge and face. Then we use JEKV to join two edges and kill a vertex. We use KEF to kill edge N4 and its associated face.

Flip22 (in figure 14) is a procedure for flipping an edge between two triangles in a TIN. For the Delaunay Triangulation, we use the "in-circle" test to test the triangle, and use the flip22 operator to change edges. The Delaunay Triangulation is based on the empty circumcircle criterion [GS85].

3.4 Building Extrusion Using Euler Operators

We used one more Euler Operator to simplify the procedure. "Make Zero-Length Edge Vertex" (MZEV) and its inverse "Kill Zero-Length Edge Vertex" (KZEV) is used to create and kill a zero length edge and a vertex (figure 15). MZEV can be replaced by using MEF, SEMV and KEF which simplify the whole procedure.

Figures 16 and 17 show extruding a four triangles building from the terrain. We can do more interactive modelling by creating tunnels and bridges from figures 18 and 21
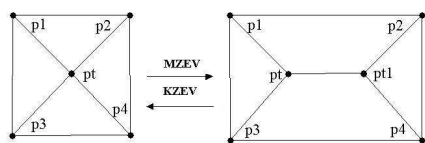

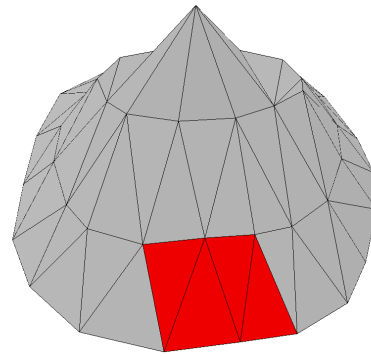Figure 15: MZEV and its inverse KZEV

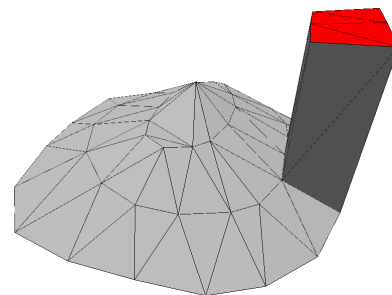Figure 16: Four triangles are selected to extrude a building

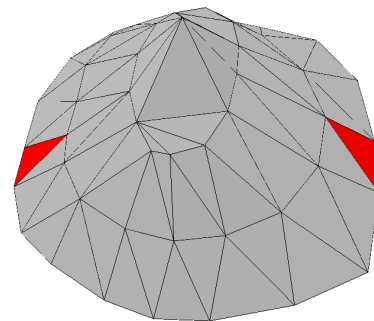Figure 17: The extruded building

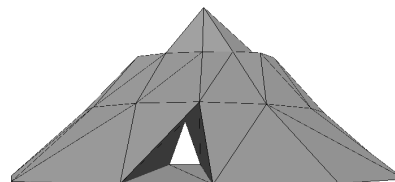Figure 18: Select two unconnected triangles to create a tunnel
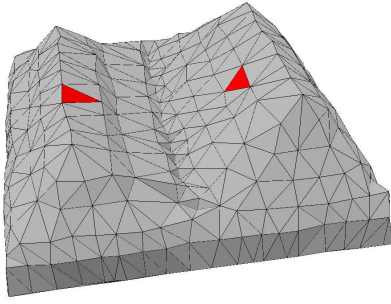

Figure 19: A tunnel is created

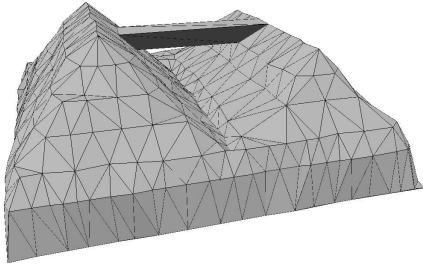Figure 20: Select two unconnected triangles to create a bridge

Figure 21: A bridge is created

4 Modelling the Roof Structures

We get extruded buildings on the terrain, but they are all flat-topped. We would like to find a way to model the roof structures. Some can be complicated. Researchers are still looking for a perfect solution for roof reconstruction.

For a simple roof we calculate the eigenvalues and eigenvectors of the vector normals of the triangles which are located inside the building boundary. The smallest eigenvalue gives the orientation of the roof ridge and shows the shape of the roof. Figure 22 shows a 2D view of the roof. The orientation of the smallest eigenvalue gives a view of the roof shape shown in figure 23. For more complex roofs we can plot the vector normals on the unit hemisphere, construct the Delaunay triangulation and then use the minimum spanning tree to extract clusters of points. Each cluster represents one roof face orientation.

5 Conclusions

We have proposed methods for embedding building models within the terrain model, in both the presence and the absence of mapped building boundaries. We have also suggested new roof modelling techniques based on the clustering of triangle vector normal orientations. While we are still refining these methods
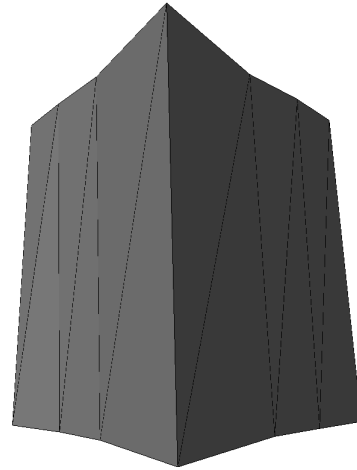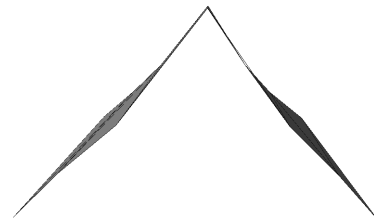

Figure 22: A simple roof

Figure 23: Roof Shape

we believe they address several previously unresolved issues in modelling buildings with LIDAR data.

6 Acknowledgement

References

[BHS78] I.C. Braid, R.C. Hillyard, and I.A. Stroud. Stepwise construction of polyhedra in geometric modelling. In K.W. Brodlie, editor, *Mathematical Methods in Computer Graphics and Design*, pages 123–141. A Subsidiary of Harcourt Brace Jovanovich, Leicester, 1978.

[Bre99] C. Brenner. Interactive modelling tools for 3D building reconstruction. In D. Fritsch and R. Spiller, editors, *Photogrammetric Week '99'*, pages 23–34, Wchmann Verlag, Heidelberg, 1999.

[DG02] M. Dakowicz and C. M. Gold. Extracting meaningful slopes from terrain contours. In P.M.A. Sloot, C.J.K. Tan, J.J. Dongarra, and

A.G. Hoekstra, editors, *In Proceedings : Computational Science - ICCS 2002, Lecture Notes in Computer Science*, volume 2331, pages 144–153, Amsterdam, The Netherlands, 2002. Springer-Verlag Berlin Heidelberg.

[Gol98] C. M. Gold. The quad-arc data structure. In T.K. Poiker and N.R. Chrisman, editors, *8th International Symposium on Spatial Data Handling*, pages 713–724, Vancouver, BC, Canada, 1998.

[GS85] L. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Transactions on Graphics*, 4(2):74–123, 1985.

[HMS03] A. D. Hofmann, Hans-Gerd Mass, and André Steilein. Derivation of roof types by cluster analysis in parameter spaces of airborne laserscanner point clouds. In H.-G. Maas, G. Vosselman, and A. Streilein, editors, *Proceedings of the ISPRS working group III/3 workshop '3-D reconstruction from airborne laserscanner and InSAR data'*, volume 34, Part 3/W13, Dresden, Germany, 2003. Institute of Photogrammetry and Remote Sensing Dresden University of Technology.

[RB03] F. Rottensteiner and C. Briese. Automatic generation of building models from LIDAR data and the integration of aerial images. In H.-G. Maas, G. Vosselman, and A. Streilein, editors, *Proceedings of the ISPRS working group III/3 workshop '3-D reconstruction from airborne laserscanner and InSAR data'*, volume 34

Session IV, Dresden, Germany, 2003. Institute of Photogrammetry and Remote Sensing Dresden University of Technology.

[Rog02] M. Roggero. Object segmentation with region growing and principal componenet analysis. In *ISPRS commission III Sysmposium "Photogrammetric Computer Vision"*, pages 298–294, Graz, Austria, 2002.

[SV04] I. Suveg and G. Vosselman. Reconstruction of 3D building models from aerial images and maps. *ISPRS Journal of Photogrammetry & Remote Sensing*, 58(3–4):202–224, 2004.

[TG01] R.O.C. Tse and C.M. Gold. Terrain, dinosaurs and cadastres - options for three-dimension modelling. In C. Lemmen and P. van Oosterom, editors, *Proceedings: International Workshop on "3D Cadastres"*, pages 243–257, Delft, The Netherlands, 2001.

[Vos03] G. Vosselman. 3d reconstruction of roads and trees for city modelling. In H.-G. Maas, G. Vosselman, and A. Streilein, editors, *Proceedings of the ISPRS working group III/3 workshop '3-D reconstruction from airborne laserscanner and InSAR data'*, volume 34, Part 3/W13, Dresden, Germany, 2003. Institute of Photogrammetry and Remote Sensing Dresden University of Technology.

[Wei88] K.J. Weiler. Boundary graph operators for non-manifold geometric modeling topology representations. *Geometric Modeling for CAD Applications*, 1988.