Segmentation of Buildings for 3D-Generalisation

Frank Thiemann, Monika Sester

Institute for Cartography and Geo-Informatics, University of Hanover
Appelstr. 9a, 30167 Hanover, Germany
Frank.Thiemann@ikg.uni-hannover.de, Monika.Sester@ikg.uni-hannover.de

KEYWORDS: Three-dimensional, Building, Segmentation, Simplification, Generalisation

1 Abstract

Three-dimensional city models are getting increasingly popular and widely used. There are many applications from visualisation over planning to simulations. However, different applications need different levels of detail. The generation of lower detailed data from higher is a generalisation problem. The generalisation has to preserve the main characteristics of the building like parallelity, orthogonality, horizontality, verticality, co planarity, etc. To save these characteristics of buildings the smaller semantic structures on the boundary have to be considered. Windows, doors, chimneys, balconies, etc. are such structures.
The paper presents an approach using an adaptation of the algorithm of Ribelles et al. [2001]. The building model is intersected with one or more planes of its boundary. All bumps, holes and notches will be detected. The so-called features will be ranked in quality order. Only the best feature is kept. Then the resulting features and the rest will be recursively segmented. For good performance, it is not possible to intersect the polyhedron with all combinations of the bordering planes. Therefore, the intersection is done first only with one plane and only if there are no good results a second, third or fourth plane is added. The results of the segmentation are in most cases convex features stored as cell complex and as CSG-tree. The cell model represents the topological adjacency of the features; the CSG-tree logs the history of the partitioning.
Based on the structures found, the semantics of the features can be assigned which, in turn, is used for an object dependent generalisation, e.g. merging of adjacent windows, enlarging a door or omitting a window, etc.

2 Introduction

A lot of towns and regions are developing digital city models. There are many applications in visualisation in architecture as well as in city and traffic planning, visualisation in navigation systems and simulation of propagation of noise, pollutants or electromagnetic waves. However, the different applications need different levels of detail. Some users want highly detailed and photo-realistic models. Most users need reduced details to accelerate the calculations and visualisations. Especially for use on small displays e.g. in navigation systems, abstracted, cartographically generalised visualisations are needed. An overview about target groups and applications for digital city and terrain models is given in [Albert et al. 2003]. A standard data model for 3D-city-models with different levels of detail is worked out in the SIG3D (special interest group 3D) in the initiative GDI NRW (geodata infrastructure North-Rhine/Westphalia) and presented in [Gröger et al. 2004].
Efficient methods for data capturing for 3D-city-models are under development (see e.g. [Brenner 2003]), but for larger areas, it is time and cost-intensive. Therefore, it makes sense to reuse the data for various users. The generation of lower detailed data from higher is a generalisation problem.
In the past, in Computer Graphics many algorithms have been developed for simplification of models with a high number of vertices, edges and meshes. An overview is given in [Heckbert & Garland 1997]. These algorithms decide primarily on geometric parameters if a vertex, edge or mesh can be removed or not. They are designed for highly redundant triangulated surfaces - using them for a low redundant building model, the main characteristics of the building like parallelity, orthogonality, horizontality, verticality, co planarity, etc. get lost. To preserve these characteristics of buildings the semantic structures on the boundary have to be taken into account, namely windows, doors, chimneys, balconies, etc.
In recent years, the issue of building generalisation has gained some interest: Forberg & Mayer [2002] use scale spaces theory: With opening and closing operations small parts can be removed or small holes can be filled. The algorithm works only with orthogonal objects, so that the buildings have to be squared in a pre-process. Kada [2002] extends an edge reduction algorithm by constraints: First, the building has to be scanned for special structures like bulge, notch or peak. Depending on the structure, different simplification functions are used. Sester & Klein [1999] describe a rule-based approach for the generalisation of building structures using operations from 2D-generalisation. The approach starts from the assumption that a representation of the building in terms of semantically meaningful parts is available. Thiemann [2002] proposes an overall concept based on a segmentation into meaningful parts, in order to also enable cartographical generalisation operations like aggregation, enhancement and typification.

The approach for 3D-building generalisation consists of three steps:

1. Segmentation,
2. Analysis & generalisation and
3. Generation of a multi-scale data structure.

The goal of the segmentation is to decompose the building into semantically meaningful parts. In this approach, only the faxes of the boundary are a determining factor for the segmentation. The result is a CSG-representation and a cell model of the building. In the analysis step, attributes of the segmented parts are calculated. Based on these attributes, appropriate generalisation functions can be applied. If the aim is to generate an object dependent generalisation, it is not sufficient to use only geometric parameters but also to take the meaning of the building parts into account and generalise them appropriately. In the presentation, there is a concentration on the segmentation process.

The paper is structured as follows: After a brief overview on possible 3D-representation schemes for buildings, the segmentation process will be described in detail. Examples demonstrate the capabilities of this approach. Then, the concept for setting up a rule base for the interpretation of the building parts will be given. The paper concludes with a summary and an outlook on future work.

3 3D-modelling for buildings

3.1 CSG, BRep, cell model

With Constructive Solid Geometry (CSG), a 3D-body is modelled as a volume. Such a solid is constructed from primitive elementary 3D-objects using Boolean operations (set operations). The sequence of operations is stored in the CSG tree. In contrast to this constructive method, Boundary Representation (BRep) is a describing method. A volume is described by its closed boundary with the topological elements mesh, edge and vertex and geometric elements e.g. a point. A special kind of boundary representation is a cell model or Euler model, which explicitly stores the topology between adjacent topological primitives (see [Kruschwitz 1996]). Each face is incident with two cells (in front of the face and behind the face). Therefore, edges can be shared between more than two faces. The cell adjacency information is stored in the faces.

3.2 ACIS for 3D modelling

ACIS is the 3D modelling kernel of the Spatial Corp. (www.spatial.com). The 3D-objects are modelled with boundary representation. In addition, it allows constructing solids with CSG-operations. It provides functions to create primitive solids like cuboid, sphere, cone, etc. and Boolean operations like union, difference, intersection. Other construction metaphors like sweeping, bending, etc. are available. The topological model of the boundary is as follows: A solid body can consist of one or more lumps. A lump is a connected part. It is bordered by one outer shell and none, one or more inner shells. Each shell consists of several faces. A face is bordered by one outer loop and none, one or more inner loops. Each loop consists of several co-edges. The edges represent the adjacency between the faces. Vertices are the borders of the edges. The geometric information is separately given with the classes point, curve and surface. The ACIS model is highly redundant to enable efficient geometrical and topological operations. For more information about ACIS see [Corney & Lim 2001].

4 Segmentation

In [Ribelles et al. 2001] an approach is given to find "features". A feature is defined as "a connected region that can be easily separated from the rest of the surface". The approach uses one or more planes – from the boundary of the polyhedron – to split the features from the rest of the body. These separated features can be located inside (behind the plane) or outside (in the front) of the resulting body.

Most buildings can be represented as polyhedra with planar faces. Ribelles' approach works with triangulated faces, but it can be modified easily. In the following, the operations are described.

Each plane divides the space into two half spaces. Defining the space behind the plane as solid and the space in the front as empty, one can operate with Boolean operations. By intersecting the body with one or more half spaces, protruding features will be removed. From this it follows that the feature is the difference of body and half space.

a) Protrusion

feature = polyhedron \ half space
rest = polyhedron \ feature

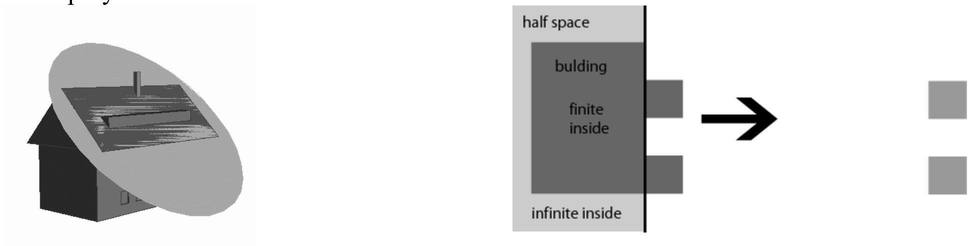

Figure 1) Operation for indication of protrusions:
Right: The two lumps of the resulting feature must be separated into two bodies.

It is more difficult to find features, which form holes. The difference of half space and body is calculated. This difference contains always an unclosed infinite part, which must be removed. If a closed finite part remains, a hole is detected.

b) Hole

feature = half space \ polyhedron,   but only closed shells = finite parts
rest = feature + polyhedron

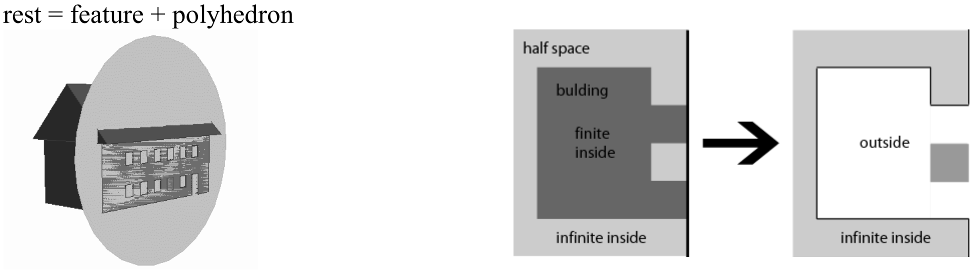

Figure 2) Operation for detection of holes:
Right: The feature is the finite lump; the infinite lump must be removed.

To find complex holes, more than one split-plane is required. A complex hole touches more than one plane of the original shell. Two splitting planes e.g. are needed if the hole lies on an edge or pierces through the body; if the hole lies on a vertex three or more planes are affected (see Figure 3, left). If an edge lies completely in a hole, four split-planes are needed to cover it. It is also possible but not necessary to cut protrusions with more than one plane.

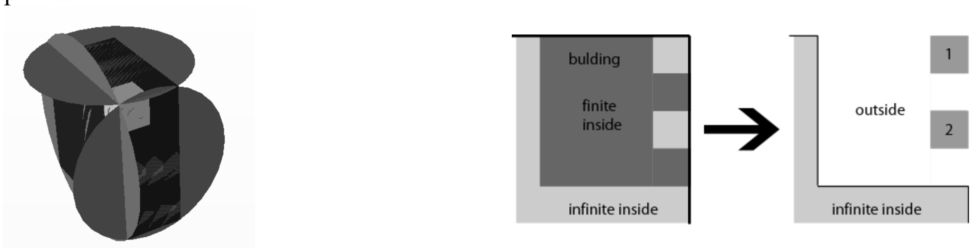

Figure 3) Operations for detecting complex holes with more than one split planes:
Right: Feature 1 can be found with two split-planes, the feature 2 also with only one split-plane.

The resulting feature of these operations can be a multi-part body. That means it consists of more than one lump. However, these lumps can be separated easily into different bodies.
As obvious from comparing Figures 1, 2 and 3, different operations are possible to segment the same polyhedron. With each plane or combination of planes, many different features can be found. To decide what the best feature is, Ribelles et al. [2001] use the quotient of the new surface area and the area of all facets of the old surface lying in the splitting planes.

quality value = new area (of the splitting face) / old area (of facets lying in the split-plane)

The smaller the value the better the feature. Figure 4 shows two different segmentations of the same body. In the left, a protrusion is detected with a quality value of 3, in the right a complex hole with the better quality value 3/7 is found.

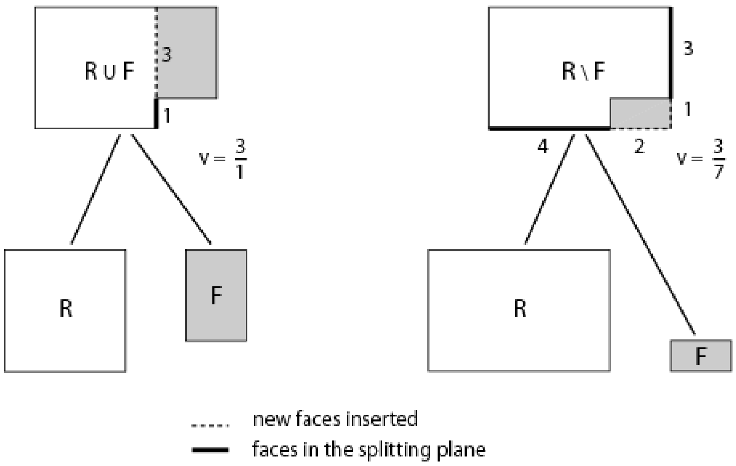

Figure 4) Two different segmentations of the same prism with different quality values (*v*).
F stands for Feature, R for Rest. The areas of side-planes are proportional to side lengths in ground plan.

The best feature is kept and then this feature and rest will be segmented recursively. The result is a CSG tree where each split is represented as a node. If splitting a protrusion, the node represents a union operation else if filling a hole the node represents the difference (see Figure 4).

The algorithm is "brute force", because it tries all splits with all combinations of planes. Running the algorithm with four or more planes is extremely time consuming. The problem is the immense number of combinations of planes and the high complexity of the operations with polyhedra. To reduce the complexity of the algorithm the number of Boolean operations must be reduced. Thus, the following extensions of the original algorithm have been developed.
The first step is to use only one split-plane at a time, and only if there is no result then two planes or more will be used. A resulting problem is that complex holes cannot be filled with only one plane. On the other hand, all protruding parts will be cut. The point is that protrusions with a bad quality value will be separated before good complex holes. To balance this, only parts with a value smaller than one ($v < 1$) are considered as valid (Figure 4). This heuristics leads to a considerable reduction in computing time.
When using more than one plane, not all combinations have to be tried. Parallel planes with the same orientation cannot form a convex feature, so only combinations of planes with different normals are possible.
If some features are found with the same split-planes and the same value, they all can be separated at the same time: for example if there are many windows of the same size in the facade.

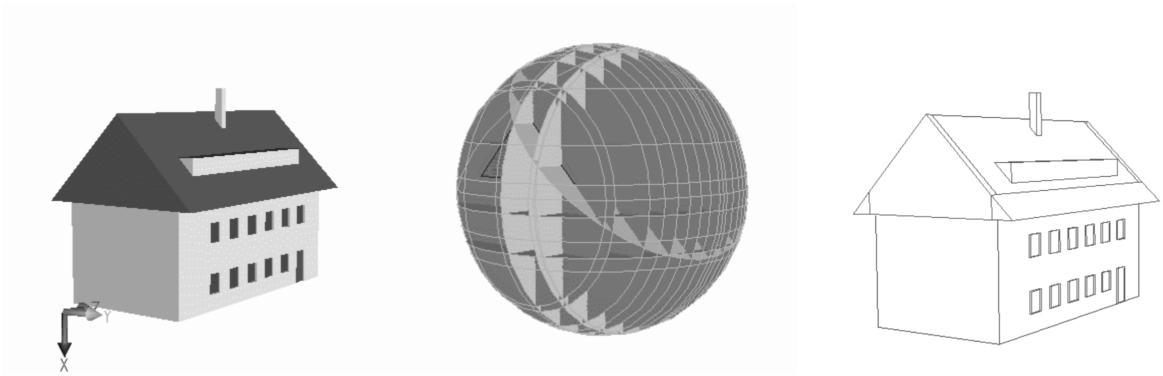

Figure 5) Segmentation of a sample building with 34 split planes. The cell model contains 19 parts.

Figure 5 shows a segmentation of a building with faces that lie in 34 different planes. The cell model contains 19 cells. The CSG tree has 19 leafs and 8 inner nodes – each of the 27 nodes stands for a recursion. In a "Brute Force" approach in the first recursion 34 splits with one plane, 561 splits with 2 planes, 5984 splits with 3 planes and 46376 splits with 4 planes have to be tested. Therefore, if with only one split plane already a split with a quality value smaller one is found, combinations of splitting planes will not be tested. Processing the example, in first recursion, the chimney would be found with quality value 0.003, testing 34 plans. As the chimney is already a convex feature, no features could be found in the recursion of the chimney, therefore 6 planes, 15 combinations of 2 planes, 20 combinations of 3 planes and the 15 combination of 4 planes – in summary 56 intersections have to be run. For the whole building 201 protrusion detections and 505 simple and complex hole detections were done. Comparing this number with the worst-case scenario of over 46376, it is a considerable improvement. In total 727 features were evaluated.

5 Analysis and generalisation

The result of the segmentation is a partition of the complex object into parts. However, this segmentation is not yet a generalisation of the building. In order to produce a generalisation sequence of the building that allows to go from a coarse to a fine representation and vice versa, the following considerations have to be made: all parts extracted in the segmentation have to be evaluated with respect to their significance for given scales or resolutions. This evaluation can be done on two levels: on the one hand on a generic level, that determines the significance of geometric parts in general, on the other hand on a specific level, that does the same but now taking the characteristics of the objects under consideration - here the buildings - into account. Concerning the first issue, this leads to the use of the generalisation function "omission". When, however, like in the second issue, object specific knowledge is available, also other generalisation functions can be applied, similar to the approach described by Sester & Klein [1999].

5.1 Generic determination of significance of features

A geometric part can be evaluated concerning its significance with respect to other parts based on elementary geometric considerations. E.g., object parts dominantly "sticking out" of the rest of the 3D-body can be considered as insignificant, depending on their size. Further aspects are described in the following. In an analysis step, geometric attributes of the features are collected and ranked:

The attributes are:

- new area inserted and quality value from segmentation
- volume, dimensions of the minimum bounding box
- difference of bounding box

These characteristics have to be combined appropriately. The quality value used in the segmentation process is suited to decide about the quality of the feature, but not to decide if it can be deleted. Also, the splitting area alone is not suited because it tells nothing about the size of the feature.

For some purposes, volume or surface area could be the main criteria. To evaluate the visibility, the minimum and maximum dimensions (which can be calculated with a minimum embedding cuboid) are better values for decision. However, it is better to decide based on more than one value if a feature can be omitted (see Figure 6). For example, a feature with a small volume can be extremely long (e.g. a chimney or flagpole) or can have a large area (e.g. a plate). Whereas in the case of the quasi-1D from all viewing points all silhouettes have only a small area, the quasi-2D has two (front and back) silhouettes with a large area.

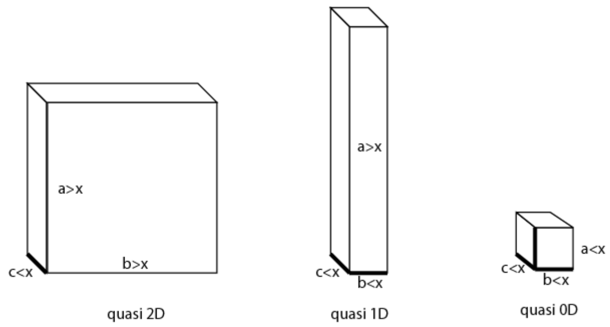

Figure 6) Different kinds of features: quasi 2D, 1D, 0D – x is the threshold.

The idea is to calculate a minimum bounding cuboid parallel to the z-axis. The edge-lengths will be sorted in descending order ($a \geq b \geq c$) so that *a* is the length of the longest edge, *b* is the length of the medium and *c* of the shortest edge. Firstly, all big features with a minimum dimension (*c*) greater than a value x_c will not be generalised. Then small features with a maximum dimension (*a*) lower than a value x_a have to be generalised (e.g. omitted, enlarged, or symbolized). Next all quasi-1D features with *b* smaller the value x_b will be generalised. Now in another step, features with a small maximum silhouette area ($A = a*b$) also can be generalised ($a*a > A > a*b$).

Now only quasi-2D objects are left. Depending on the position of the large faces relative to the biggest touching part, different generalisation operations are required. The position can be calculated by the difference between the bounding boxes (see Figure 7).

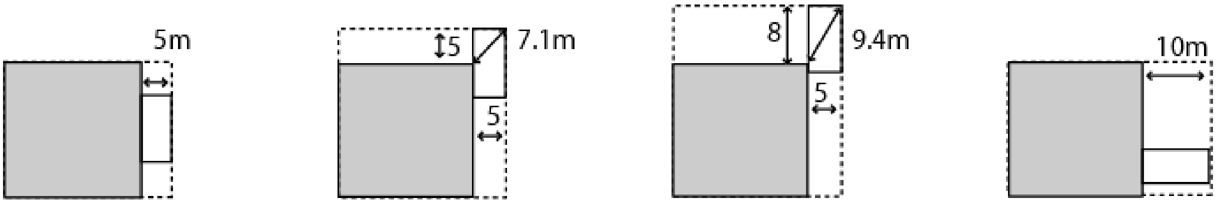

Figure 7) Depending on the position of the thin side of a quasi-2D feature relative to the biggest touching part, different generalisation operations are required.

5.2 Generalisation taking building characteristics into account

A prerequisite for an object dependent generalisation of buildings similar to Sester & Klein [1999] is an interpretation of the individual building parts in terms of windows, doors, chimneys, roofs, balconies, etc. Such an interpretation can be derived by setting up formal models to identify those building objects. These object models are composed of elementary geometric and topologic characteristics, as well as using the context. The following list of attributes can be derived automatically from the segmented parts:

- inclination of the plane, horizontal or vertical orientation
- height of feature over ground level
- similarity of features in given neighbourhood
- context: containment of object in other object (e.g. window in a wall)

Furthermore, additional common sense knowledge about the building features can be used:

- typical size of windows
- typical height of stories
- typical number of windows / doors per building / per façade
- typical position of chimney
- typical position of balcony

Based on an appropriate combination of these features, an interpretation of the segmentation result into meaningful parts is possible. Then, an object dependent generalisation can be applied, e.g. aggregation of adjoining windows of one floor, enlargement of doors, removal of balconies.

6 Conclusions / Outlook

In the paper, the segmentation of a building has been presented. An existing approach for the segmentation of polyhedra has been extended: firstly, heuristics have been introduced in order to reduce the time complexity of the processing, secondly, the segmentation is described by Boolean operations in a CSG tree. Based on this segmentation, a generalisation hierarchy can be established.
The CSG tree resulting from the segmentation is usually very unbalanced. Small parts are close to the root - as they are segmented early in the process - and the biggest part is in the deepest leaf. In the future work, it will be examined if a reorganisation of the tree is useful, so that bigger parts get nearer to the roof and the tree gets more balance.

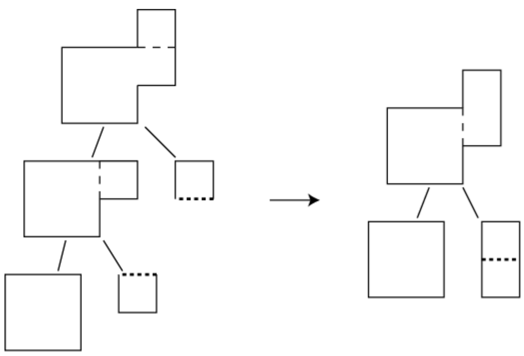

Figure 8) Touching features (dotted lines) can be merged afterwards to rearrange the CSG-tree.

In some cases (for example see Figure 8) a feature is generated that forms a bigger convex part with a neighboured feature. Here is to study, in which cases these features have to be merged or not.

For the generalisation operations symbolisation, aggregation and typification, semantic information is required. As described in the previous section, this semantic information will be derived from the geometric and topologic characteristics. To this end, object models will be set up, that allow for an automatic determination of those parts. This will also be subject to future work.

7 References

ALBERT, J.; BACHMANN, M.; HELLMEYER, A.: Zielgruppen und Anwendungen für digitale Stadtmodelle und digitale Geländemodelle. Collection in the context of the working group "applications and target groups" in the SIG3D of the initiative GDI North-Rhine/Westphalia, April 2003. http://www.ikg.uni-bonn.de/sig3d.

BRENNER, C.: Building Reconstruction from Laser Scanning and Images. In: Proc. ITC Workshop on Data Quality in Earth Observation Techniques, Enschede, The Netherlands, November 2003.

CORNEY, J.; LIM, T.: 3D Modelling with ACIS. Saxe-Coburg Publications 2001.

FORBERG, A., MAYER, H.: Generalization of 3D Building Data Based on Scale-Spaces. In: "Geospatial Theory, Processing and Applications", IAPRS Vol. 34, Part 4, Ottawa, Canada, 2002.

GRÖGER, G. et al.: Das interoperable 3D-Stadtmodell der SIG 3D der GDI NRW. Handout of the working group “modelling” in the SIG3D in the GDI NRW, May 2004, http://www.ikg.uni-bonn.de/sig3d.

HECKBERT, P.; GARLAND, M.: Survey of polygonal surface simplification algorithms. Technical report, CS Dept., Carnegie Mellon University 1997. http://www.cs.cmu.edu/~ph.

KADA, M.: Automatic generalisation of 3D building models. In: "Geospatial Theory, Processing and Applications", IAPRS Vol. 34, Part 4, Ottawa, Canada, 2002.

KRUSCHWITZ, E.: Euler-Modellierung dreidimensionaler Körper. Shaker Verlag 1996.

RIBELLES, J.; HECKBERT P.; GARLAND, M. ; STAHOVICH, T.; SRIVASTAVA, V.: Finding and Removing Features from Polyhedra. In: American Association of Mechanical Engineers (ASME) Design Automation Conference, Pittsburgh PA, September 2001.

SESTER, M.; KLEIN, A.: Rule Based Generalization of Buildings for 3D-Visualization. In: Proceedings of the International Cartographic Conference (ICA), Ottawa, Canada 1999.

THIEMANN, F.: Generalization of 3D Building Data. In: "Geospatial Theory, Processing and Applications", IAPRS Vol. 34, Part 4, Ottawa, Canada, 2002.