International Journal of Geographical Information Science
Vol. 19, No. 8–9, September–October 2005, 975–997

Research Article

Scale-spaces for generalization of 3D buildings

H. MAYER*
Institute for Photogrammetry and Cartography, Bundeswehr University Munich, D-85577 Neubiberg, Germany

This paper presents a means for automatic generation of a level of detail (LOD) representation of three-dimensional (3D) building data, with the formally well-defined scale-spaces as the underlying theory. More specifically, we propose an approach that employs vector-based mathematical morphology and discrete curvature-space to generalize two-dimensional (2D) building outlines as well as 3D building surfaces. Scale-space events that are related to the semantics of objects are the major triggers of the generalization. The implemented approach preserves right angles. Additionally, more heuristic means to square buildings are presented. The test results have validated the approach.

Keywords: Generalization; Scale-space; Level of detail; 3D building data

1. Introduction

Developments such as the VRL '97 (the virtual reality modelling language, ISO/IEC 14772-1:1997) standard allow an easy visualization of complex three-dimensional (3D) data such as buildings. However, acquisition of building data is quite labour-intensive. For applications such as visualization or analysis in a geographical information system (GIS), the aggregation level of the acquired data is often unnecessarily low. That is, the data are too complex, or at least one would like to have additionally a less detailed representation. A typical example using such less detailed representations is the level of detail (LOD) concept from computer graphics. This has been employed to reduce the effort for visualization of objects at a larger viewing distance.

Many approaches that serve the purpose of data simplification already exist in computer graphics, computational geometry, and computer vision. They often yield multi-resolution surfaces by means of surface simplification. Common to all these approaches is that they are generally applicable, not regarding specific properties of the objects. Yet to obtain a high quality, it is desirable to develop a simplification procedure adapted to the specific object, in this case the building. Generalization research in cartography and GIS has so far been mostly focused on two dimensions (2D), with the main objective to transform data into the right level of abstraction for a particular analysis task.

In this paper, we propose basing the geometric generalization of building outlines in 2D and 3D on the formally well-defined scale-space theory from image analysis and image processing initiated by Witkin (1983) and Koenderink (1984). Scale-space theory deals with the formal definition of the concept 'scale' in terms of signals or

*Corresponding author. Email: helmut.mayer@unibw.de

International Journal of Geographical Information Science
ISSN 1365-8816 print/ISSN 1362-3087 online © 2005 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/13658810500161286

images with the scale-parameter describing the current level of scale. The image is smoothed, for example with a 2D Gaussian kernel, but the original resolution is retained. The smoothing can result in the split and merge of features in the image causing the elimination of details, or information with high frequency. Scale-space theory defines mathematical constraints for these split and merge events. We call the latter 'scale-space events' and propose employing them as a basis for generalization.

Mathematical morphology (Serra 1982) and discrete curvature-space derived from the reaction-diffusion space of Kimia *et al.* (1995) are the main scale-spaces used in this paper. Mathematical morphology has been used for the generalization of raster data, for example by Li (1996), and proved to be suitable for the generalization of buildings by Su *et al.* (1997) and Cámara and López (2000). We extend these works as follows:

- By widening the scope from mathematical morphology to general scale-space theory, we extend not only the theoretical background but also the set of operators available for generalization.
- As a vector representation with polygons in 2D and polyhedra in 3D is adopted, the generalization becomes independent of the orientation of the objects. Moreover, vector data make it easy to preserve right angles or enforce them.
- The generalization of buildings is extended to 3D.

The scale-spaces are generated incrementally: the outline is shifted in small steps, and the scale-parameter is determined as step-size times the number of steps. An important reason for an incremental processing is that scale-space events occur, for example, when parts of the building such as an annex are eliminated, when a building is split into parts, when an inner courtyard arises, or when two buildings are aggregated. The events are classified into internal and external events.

The most important findings of generalization based on scale-spaces are that:

- only relatively few parameters are required that have a precisely defined semantics, namely the step-size, the number of steps for opening, the number of steps for closing, and the number of steps for discrete curvature-space as well as a flag defining if the building should be squared, that is, if right angles should be enforced.
- by using incremental processing, the emerging scale-space events can be handled (at least in 2D) by simple basic operators.

Section 2 gives an overview, beginning with an introduction to scale-space theory, as it is not assumed to be well known in GIS. The relation between scale-space events in an image or a geometrical representation and the abstraction in an object representation is addressed. Then, the state of the art of surface simplification and generalization is outlined. Section 3 presents mathematical morphology for 2D building outlines in the form of opening and closing along with a description of internal and external events. After the definition of the discrete curvature-space, the combination of opening, closing, and discrete curvature-space is presented together with results, and more heuristic means to square building outlines are shown. Section 4 comprises the main part of our work. Following a description of our goals, the methods for generating LOD representations in 3D, data structures and input of data are discussed. Opening and closing as well as discrete curvature-space are presented for the 3D case including various examples. Finally, the combination of the different operators with squaring is shown. In section 5, the approach is discussed, and section 6 draws conclusions.

2. State of the art

2.1 *Scale-spaces*

Scale-space theory deals with the formal definition of the concept 'scale' (Witkin 1983, Koenderink 1984, Lindeberg 1994) in terms of signals or images. The basic idea is to generate multiscale-representations from a one-parameter family of derived signals. Data are systematically simplified, and details, or information with high frequency, are eliminated. The scale-parameter, often named $\sigma \in R_+$, is intended to describe the current level of scale.

A significant prerequisite for a particular scale-space is the so-called 'causality' according to Koenderink (1984): any feature in a coarse scale (large scale-parameter), be it point, line, or region, needs to have a (not necessarily unique) cause in a fine scale (small scale-parameter). More intuitively, this means that every feature at a certain scale is constructed of possibly weaker features at finer scales. It also means that features are not allowed to pop up at places where nothing has been before at finer scales.

The semantics of the scale-parameter additionally depends on the definition of a particular scale-space. If isotropy (independence of direction) and homogeneity (independence of location) are combined with causality and a continuous scale-parameter, the scale-space family necessarily satisfies the diffusion equation (Koenderink 1984) utilized for instance in physics to describe heat transfer. The convolution of the image with the 2D Gaussian Kernel is the solution of the diffusion equation for an infinite domain. This results in the so-called 'linear scale-space' of Lindeberg (1994).

Another way to define a scale-space is based on mathematical morphology (Serra 1982). Mathematical morphology can be defined for grey-scale imagery, but here we narrow our scope to binary images. Li (1996) has introduced mathematical morphology for the generalization of raster data. Erosion and dilation are the basic operators of mathematical morphology. For binary images, the morphological scale-space is generated by 'filtering' the image with often circular or square-shaped structuring elements whose size varies with the scale-parameter. Intuitively, the erosion operator means that the structuring element is shifted over the whole image and is compared with the corresponding part of the image. At all locations where all set (black) pixels of the structuring element coincide with black pixels in the given image, the reference point of the structuring element, usually its centre, is written as a black pixel into a new blank image. As this image consists of only the centres, regions in the image are thinned, or in other words eroded. Contrary to this, the dilation operator writes out the black pixels of the structuring element translated by the reference point into a new blank image at all locations of black pixels in the given image. This leads to an expansion, or a dilation of regions. Opening and closing are the two important operators constructed from erosion and dilation using the same structuring element for both operators. Opening consists of erosion followed by dilation, and closing is realized with dilation followed by erosion. While opening is used to eliminate small isolated groups of pixels and narrow parts as well as to broaden rifts, closing is employed to smooth outlines as well as to eliminate small holes and rifts.

The so-called 'reaction-diffusion' space of Kimia *et al.* (1995) combines linear and morphological scale-space. It comprises an inhomogeneous diffusion as well as a morphological component and is defined on closed outlines. Basically, the outline of a region is shifted in the normal direction controlled by two parameters. The first parameter is multiplied with the curvature and describes the strength of the curvature-dependent smoothing (diffusion). The second parameter controls the strength of the reaction. Kimia *et al.* (1995) show that the reaction is equivalent to erosion or dilation, with circles as structuring functions. The diffusion part is termed curvature flow by Malladi and Sethian (1996) and 'curvature-space' in this paper. To avoid problems at points with high curvature, the outline is sampled with a small step-size, normals and curvatures are computed for all points on this outline, and the shift is realized in small incremental steps. In figure 1, the characteristics of reaction and diffusion are demonstrated for an elongated object. The object is eroded in the reaction (morphological) part with the same speed from all sides, finally leading to the medial axis of the object. Opposed to this, by inhomogeneous diffusion, the parts of the outline are moved with a different speed depending on the curvature. This results in the tendency to preserve elongated objects, for they are only eroded from their high-curvature ends.

The introduced scale-spaces have different characteristics. While the linear scale-space smoothes the image function more or less continuously, the morphological scale-space eliminates regions with a spatial extent that is too small. The reaction-diffusion space integrates the above two scale-spaces. Yet, compared with linear scale-space, its inhomogeneous diffusion component (curvature-space) has the interesting property of preserving elongated regions. This is due to the fact that the curvature-space takes into account the curvature of the outline.

2.2 *Scale-space events and abstraction*

For this paper, scale-spaces are of special interest in combination with an object representation. The latter means that not only is an object, such as a building, visible (for a human) in an image, but there is a representation linked for instance to labels of the image pixels making up the building. Two components are considered for the link of scale-space and the object representation.

On the one hand, features such as edges can be tracked over different scales, and features in different scales can be linked. For instance, a line in a coarse scale can be matched to two bounding edges in a fine scale. Kosslyn (1994) elucidates the usefulness of this linkage from a psychological point of view: the behaviour of features when scale changes is important for the recognition of their corresponding objects. Bobick and Bolles (1992) show the representation of an object approaching a sensor. The object becomes larger and larger while the features which can be

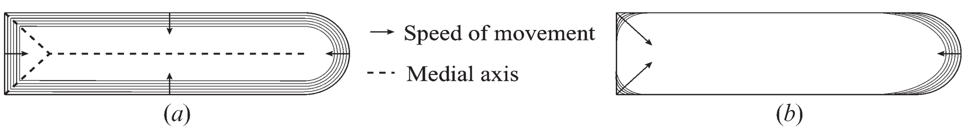


Figure 1. Elongated object under reaction and diffusion. (*a*) Reaction (morphology). Object is eroded with the same speed from all sides. Finally, the medial axis is obtained. (*b*) Inhomogeneous diffusion. The outline is moved with the speed depending on the curvature, which is zero for the straight parts. (Colour versions of all figures, where appropriate, are available online.)

extracted change. When, for instance, a leafless bush is approached, the features change from a dark point through a dark region to several dark lines representing the branches.

On the other hand, objects merge at certain scales according to their spatial proximity or similar properties, thus generating a more compact representation or making the analysis easier. When using spatial proximity, this is called part or aggregation hierarchy (Timpf 1999). When similar properties are employed, it is termed class hierarchy (Molenaar 1996) or generalization hierarchy (Timpf 1999). Both hierarchies describe the behaviour of objects with a spatial extent dependent on how they can be represented and analysed. Here, an implicit scale is inherent. This scale becomes explicit as soon as the object is visualized for example in a map.

The connection of scale-spaces and object representations was the focus of Mayer (1996) and Mayer and Steger (1998), according to which essentially two things happen simultaneously, when an image comprising characteristic object features is transformed in scale-space from a finer to a coarser scale:

- The information density of the image is reduced by means of scale-space events by eliminating points, edges, lines, and regions. Not only noise, but also meaningful information, is removed.
- The elimination of meaningful information can be considered equivalent to the elimination of parts of objects. In addition, it results in a simplification, that is an abstraction, and therefore an improved performance of object extraction.

The simplification makes it possible to link an object representation based on an aggregation hierarchy with the outcome of a scale change in the image. In many cases, also the geometrical representation of an object simultaneously changes with the elimination of parts. A road is typically represented as a region in a fine scale. Yet, in a coarse scale, it is more reasonable to represent it as a line. The term abstraction and its link to different hierarchies are discussed in detail by Timpf (1999), though without presenting practical means for the abstraction.

Scale-space events can be classified into annihilation, merge, split, and creation. Figure 2 presents an example of split and merge. Linear scale-space, i.e. Gaussian kernels with a different value for the scale-parameter, was used. While the narrow connection between the two squares is a detail that can be omitted at an intermediate scale, a more compact region is a reasonable approximation at a coarse scale. Note that figure 2(*b*) is only a 'split' when looking at a series of different scales.

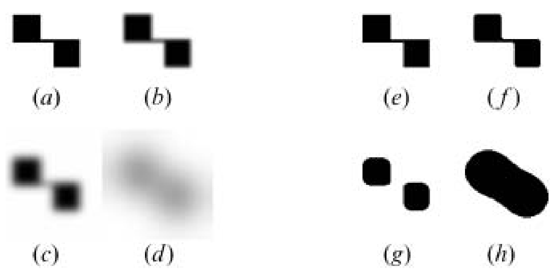


Figure 2. Split and merge for smoothing in linear scale-space (2D Gaussian kernel). σ of the Gaussian is given in pixels with the size of the images 256×256 pixels and one building block consisting of 64×64 pixels: (*a*) to (*d*) images; (*e*)–(*h*) images after thresholding; (*a*) and (*e*) input image; (*b*) and (*f*) σ=2—smoothing of the outline; (*c*) and (*g*) σ=5—split into two regions; (*d*) and (*h*). σ=20—merge into one region.

Here, it was directly obtained by means of the particular Gaussian kernel. The 'merge' can be seen as an instance of the change in geometry type presented by Schoppmeyer and Heisser (1995). A region becomes a line or a point during generalization. The new geometry has to be delineated, and a treatment of the neighbourhood might be necessary. Annihilation and creation are analogous to the appearance and disappearance of objects when zooming in or out or, in other words, when the scale is changed, as conceptually analysed by Timpf and Frank (1997). Timpf (1999) discusses the general relationship between abstraction, different hierarchies, and levels of detail.

The abstraction capability of scale-spaces and results as demonstrated in figure 2 suggest that scale-spaces be employed for generalization. Their feasibility has already been proved by Li (1996) for the morphological scale-space.

2.3 *Surface simplification*

There are many approaches from computer graphics, computational geometry, computer vision, and cartography dealing with the problem of surface simplification, which can be regarded as a part of the generalization of 3D buildings represented, for instance, by polyhedra. A survey on surface simplification is given by Heckbert and Garland (1997). Typically, the approaches focus on general objects made up of triangles. Many of these approaches have the goal of constructing a LOD representation, which can consist of different levels but also of a sequence of refinement transformations (Popović and Hoppe 1997). Garland and Heckbert (1997) present an approach to simplify surfaces and to generate a LOD representation which tries to balance mathematical soundness, speed, and quality. Short edges are contracted, and nearby points are merged. The so-called 'ϵ-offset surfaces' (parallel surfaces with distance ϵ) by Varshney *et al.* (1995) guarantee that the approximation is closer to the original than ϵ. Schmalstieg (1996) presents an approach to produce a LOD representation for VRML 1, the predecessor of VRML '97, based on octree-clustering. A study that is most relevant to our approach is described by Ribelles *et al.* (2001). This treats the problem of finding and removing features from polyhedra. The goal is to coarsen computer-aided design (CAD) models for finite element analysis. Common to all the approaches presented in this section is that they try to be general, not using the specific nature of objects such as buildings.

2.4 *Generalization*

Generalization aims at scale transition either for maps or for topographical or geographical information. The latter can also be interpreted as the generation of a LOD representation. In the context of generalization the term 'scale' is usually defined as the ratio of the length between two points in the analogous representation such as the map and the length between the corresponding points in reality.

For decades, relentless efforts have been made towards automation in generalization. This paper is devoted to the simplification of geometry, topology, and (by these means also indirectly) the semantics of objects, rather than their graphical representation. Our endeavour results in a modelling of geographic information on different levels of abstraction, which facilitates data analysis as it can be done on the level at which the spatial processes are understood best (Müller *et al.* 1995). The information could be stored in a multiple representation database where links between the levels make an incremental update possible as proposed by Kilpeläinen (2000).

The main topic of this paper is the automatic generalization of buildings. It has a long tradition at least in 2D. A classic approach that works on building outlines was developed by Staufenbiel (1973). First, threshold values for area, length, and spacing are derived from graphical minimum dimensions on maps. A fixed processing scheme is then determined to remove edges below the threshold, which is realized by simplifying or eliminating corners and by eliminating or emphasizing small outward pointing annexes or inward pointing notches. Neighbouring buildings are merged if their distance falls below the threshold. The smaller of two buildings would be displaced and aligned with the larger building. The overall approach is limited in that it works mainly locally. Neither the global shape of the building nor the context in terms of the characteristics of neighbouring buildings is considered. Meyer (1989) goes one step further by classifying the building shapes. Classes besides the rectangle are for instance L-, T-, U-, and Z-shaped regions. The classification is based on the correlation of the rasterized buildings with a template. If a classification is not possible, a simplification according to Staufenbiel (1973) is employed. Though the classification is important for suburban areas with relatively simple buildings, for more complex areas the same deficiencies exist as for the work of Staufenbiel (1973).

Regnauld *et al.* (1999) show how a suitably defined sequence of three simple algorithms for simplification, squaring, and local enlargement can bring about desired results of building generalization. According to an empirical study, the determination of an appropriate sequence depends on the target scale and the type of building (large vs. small). Though this is a step forward, there is still much to do concerning the improvement of the operators, especially for the development of an automatic control. A production net consisting of hierarchically organized rules which construct a hierarchy of parts is used by Stilla *et al.* (1998) to generalize relatively simple building outlines from different information sources. In this context, productions refer to the elimination of annexes as well as inward pointing notches with the goal to compare and combine the information from different sources.

Ruas (1999) proposes a model based on autonomy, constraints, and a hierarchical presentation for the generalization of urban areas. It is shown that groups of objects, termed 'meso situations', are of particular importance: They render context to delimit the values of parameters and control the processing according to the regional characteristics. This work has provided a means for a more global control. Recently, least-squares adjustment was introduced by Sester (2000) as a technique for generalization. Observations are introduced in terms of competing constraints. Via least-squares adjustment a holistic solution is obtained. Even though this gives good results for squaring, aggregation, and displacement of building ground plans, there are still weaknesses in dense areas for which, for example, a preceding typification is proposed.

Li (1996) deals with scale-spaces, particularly binary mathematical morphology on raster data. Opening and closing based on simple structuring elements are employed for the generalization of regions with smooth outlines. That mathematical morphology can be used for conflict detection, cartographic displacement, and region to line collapse (skeletonization), is documented by Li and Su (1997), Su and Li (1997), and Su *et al.* (1998).

Most relevant to our approach are the works of Su *et al.* (1997) and Cámara and López (2000) which employ mathematical morphology on raster data for the generalization of buildings in 2D. Su *et al.* (1997) focus on aggregation and compute the size of the structuring element from a minimal distance between different regions given for a map and the ratio of source scale and target scale. The two-step procedure begins with the so-called 'natural combination' which is dilation followed by erosion. The same structuring element is used for both operators, in most cases leading to the closing operator. The second step is shape refinement where either the convex hull is computed or the so-called 'irregularity filtering' consisting of opening, convex hull computation, and the computation of the complementary sets is employed. While the latter is somehow more heuristic, the results are more realistic than for the convex hull which, in the definition used here for raster data, always leads to rectangles. As a whole, the paper confirms the basic feasibility of the generalization of buildings based on mathematical morphology on raster data. In a more production-oriented paper, Cámara and López (2000) employ five different 3×3 structuring elements and up to four erosion or dilation operations to generalize urban city blocks. To find the optimal combination of erosion and dilation and the structuring elements, they have computed results for all possible options and compared them with manually generated ground plans as ground data . The optimal combination depends not only on the data used but also strongly on the different indices used to compare the automatic and the manual result such as the percentage of agreement or entropy. As usually no ground data are available, developing convincing control algorithms is quite a challenge. Because the buildings are discretized in the raster data, the results for Su *et al.* (1997) and Cámara and López (2000) both depend on the angle of the buildings to the coordinate axis.

To this end, one would ask how well scale-spaces could fit into a generalization task based on expert knowledge presented at the beginning of this section. To answer the question, a basic experience from image analysis can be stressed: expert knowledge in the form of heuristics is not suitable for low-level methods. It is of fundamental importance to employ low-level methods which are mathematically sound and which have only a few, well-defined parameters, such as a scale-parameter constructed from step-size times the number of steps, directly corresponding to the shift of the outline.

In the remainder of this paper, we focus on the extension of the formal methods or, more precisely, the set of available operators for generalization, by widening the scope from mathematical morphology to general scale-space theory. The acquisition, representation, and use of expert knowledge are therefore beyond the goal of this paper, but still we acknowledge its importance for controlling the proposed operators for a practical application. The only expert knowledge implicitly employed here is that we model our objects to be made up mostly of straight edges and right angles, and that larger structures should be kept when using a small scale-parameter. The subsequent section shows how mathematical morphology can be employed on 2D vector data, how abstraction occurs by internal and external events, how a discrete curvature-space can be defined in 2D, and how buildings can be squared.

3. Generalization of 2D building outlines

This section focuses on the generalization of 2D building outlines or the generation of a 2D LOD representation. The generalization in 2D based on scale-spaces provides not only a means to implement 2D mathematical morphology on vector data but also a fundamental understanding of its extension in 3D.

3.1 *Opening, closing, internal events, and external events*

In image analysis, scale-space operators are mostly employed in a raster-based way. The rounding up of the right angles in figure 2 is not suitable for the generalization of man-made objects such as buildings. It is preferable to use a vector-based representation, where discontinuities, particularly right angles, can be represented easily. The scale-space operators to be utilized should have basic properties such as causality, which is the case for opening and closing. The vector outlines are shifted inward or outward as in the reaction part of the reaction-diffusion space (cf. figure 3(*b*)). It was shown by Kimia *et al.* (1995) that shifting the outlines inward or outward corresponds to erosion and dilation with a circle (ball for higher dimensions) as a structuring element. Taking into account our expert knowledge, we have modified these operators by intersecting straight lines, for the purpose of retaining corners. For buildings with right angles, our modification results in the equivalence of erosion or dilation of the building outline with a square structuring element with a side parallel to the building outlines.

The shift of the outline is made incrementally in small steps, as shown in figure 3(*b*) and (*c*). The scale-parameter is determined as the step-size times the number of the steps. The incremental shift is adopted in accordance with Kimia *et al.* (1995). While they also employ it to deal with highly curved portions of their general outline, the main point is the treatment of scale-space events or 'shocks' in the notation of Kimia *et al.* (1995). In figure 3(*b*) and (*c*), the scale-space event results in the elimination of the inward pointing notch in the middle of the upper part of the building. We classify scale-space events into internal and external.

Internal events occur in topologically local areas. They are synonymous with one or more unnecessary points or very short segments which are shorter than the half step-size. These points or very short segments emerge when incremental displacement is employed. The internal events can be further classified into U, Z, and L (cf. figure 4), for which simple basic operators exist: for U events, small inward or outward pointing structures are eliminated. Z events are solved by averaging straight lines. For L events, two or more very short segments are eliminated, and one point is generated by intersecting the longer segments neighbouring the very short segments. For opening and closing, only U events occur. In other words, only structures with inward going notches or outward going annexes can be treated.

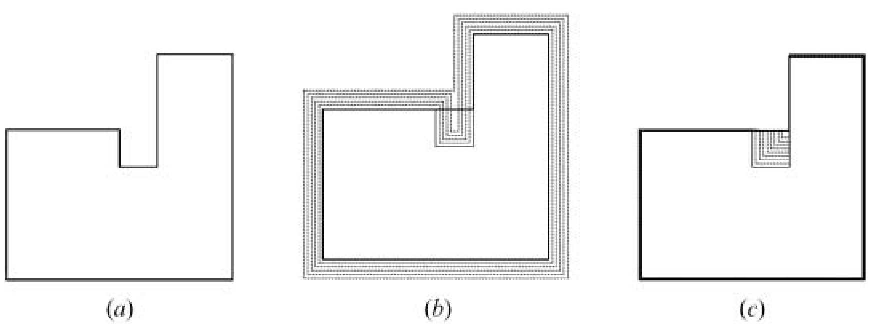


Figure 3. Result of the application of different scale-space operators based on vector data: (*a*) input data; (*b*) closing; (*c*) discrete curvature-space; (*b*) and (*c*) intermediate steps=dashed lines; result=thick lines; in (*b*), the segments are shifted first outward and then in one large step inward again.

Figure 4. Internal scale-space events: (*a*) U event; (*b*) Z event; (*c*) L event.

External events emerge when topologically non-local segments (segments which are not directly connected by a vertex or by one or many very short segments) of a building or any segments of different buildings touch or overlap. They lead to a significant change in topology. Since we use an incremental approach, events can be detected as follows: a vertex is less than half the step-size away from non-local segments of the building itself or any segments of another building. As all the vertices have to be checked for their distances to all segments, this has a relatively high complexity. However, one can reduce the computing intensity considerably by employing bounding boxes or efficient data structures. Typical examples for external events are the split of objects connected by narrow connections in the course of opening, and the elimination of small gaps such as a passage, when closing (cf. figure 5). Likewise, two or more buildings can be merged.

3.2 *Discrete curvature-space*

After introducing mathematical morphology in the form of opening and closing in the preceding section, this section is dedicated to another scale-space operator, namely discrete curvature-space. This is obtained when only segments under a certain minimal length are shifted. This leads to results like in those in figure 3(*c*). The procedure for the discrete curvature-space is as follows:

1. For one short segment that is perpendicularly located between two longer segments, we identify two cases:
   - If the two longer segments lie to the same side of the short one, thus forming an inward or outward pointing U-shape (cf. figure 6(*a*)), we shift the short segment outward or inward, respectively.
   - If the short segment is flanked by two longer ones (Z-shaped structure; cf. figure 6(*b*)), we shift the two long segments in such a way that the short segment is getting even shorter.
2. For more than one short segment, we shift two segments which are locally L-shaped, inward or outward. The direction depends on the angle in the building interior connecting the short segment to the longer neighbouring segments or another L-shape. If this angle is 90° (quarter circle; cf. figure 6(*c*)), the short segments are shifted outward; if it is 270° (three-quarter circle; cf. figure 6(*d*)), they are shifted inward. When the calculation of the shift-values is done in one direction of the building outline (for figure 6(*c*) and (*d*), it is assumed that the

Figure 5. External scale-space events. (*a*) Opening: the thin horizontal connection is eliminated after the inward going segments meet. (*b*) Closing: the vertical gap is closed after the outward going segments meet; thick lines in (*a*) and (*b*): external scale-space event.

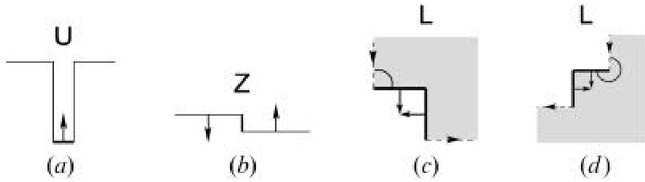


Figure 6. Procedure for discrete curvature-space: (*a*) U-shape, 1 short segment; (*b*) Z-shape, 1 short segment; (*c*) L-shape, 2 and more short segments 90°; (*d*) L-shape, 2 and more short segments 270°; (*c*) and (*d*) quarter or three-quarter circle indicates the 90° or 270° angle, respectively, in the building interior represented in grey connecting the pair of short segments (thick lines) to the long neighbouring segment or another L-shape. For the building outline, it is assumed that the direction of the building outline goes downwards as indicated by arrows.

direction of the building outline goes downward as indicated by an arrow), it is guaranteed that also for more than one L shape, a meaningful result is obtained.

This procedure is similar to curvature-space (cf. section 2.1). The problem is that for straight lines, the curvature is zero, and for the corners, the curvature can be seen to be either undefined or infinite. Here, again expert knowledge is used. Looking at a building outline, the intuition is that when approximating it with a smooth contour, parts with shorter segments would be approximated with higher curvature. Therefore, we take as the curvature radius half the length of the shorter segment connected to a vertex. Examples can be seen in figure 7(*a*) for the U-shape and (*b*) for the Z-shape. In the reaction-diffusion space of Kimia *et al.* (1995), points are shifted according to the curvature. As described in section 2.1, the outline is sampled with a small step-size, resulting in a rounded outcome such as in figure 1. Instead of this, here only short segments below a certain length, or in other words points with high curvature, are shifted. Thus, the minimum curvature radius is eventually enlarged to half of the minimum length of the segment. This is the reason why we term this procedure 'discrete curvature-space'.

3.3 *Combination of operators and results in 2D*

In the following, a sequence of opening, closing, and discrete curvature-space with 10 successive steps for each operator and for all four examples was employed. The step-size was 0.5 units, and typical smaller building parts measure 2–5 units. For discrete curvature-space, the minimum length was chosen to be 6 units. After the application of the scale-space operators, the buildings were re-scaled to their original size. Figure 8 presents our results for two reasonably complex buildings. Since we start with opening and closing, all U events (annexes and notches) are treated by opening or closing, and only the Z and L events (step- or stair-structures) are handled by the discrete curvature-space.

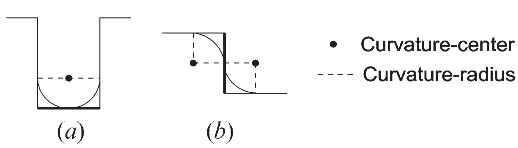
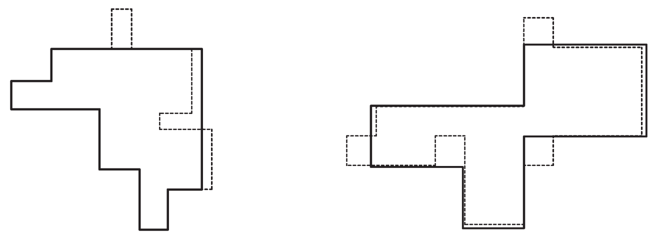


Figure 7. Discrete curvature-space. Curvature radius for (*a*) U-shape and (*b*) Z-shape.

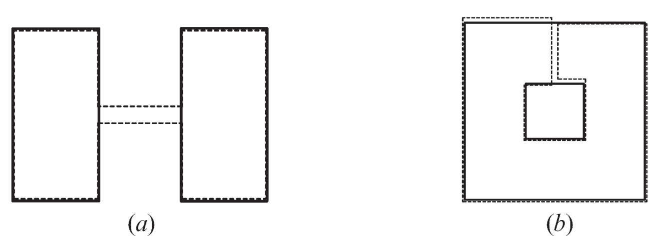


(*a*) (*b*)

Figure 8. (*a*) and (*b*) Results for two reasonably complex buildings. A sequence of opening, closing, and discrete curvature-space with a step-size of 0.5 units, 10 steps, and a minimum length of 6 units were used for all three procedures. Typical small building parts measure 2–5 units.

Figure 9 presents two examples of external events: segments which are not directly connected are merged when the distance becomes smaller than the half step-size. Figure 9(*a*) shows the external scale-space event in which a building with a narrow connection is split into two separate buildings. In figure 9(*b*), an inner courtyard arises after eliminating a narrow passage. The external scale-space events only occur in morphological scale-space, as the discrete curvature-space with its shifting of short segments only causes internal events. In figure 9(*a*), the boundaries of the narrow connection meet during closing, while in figure 9(*b*), the boundaries of the narrow passage collide during opening (cf. also figure 5). The deletion of the corresponding segments leads to a change in topology (split and generation of hole), which is characteristic for external events.

Practically, we start with shifting the segments inward step by step for the opening. At each step, we check if a segment becomes shorter than the half step-size or if the distance of a vertex to a non-local part of the outline or another building becomes smaller than the half step-size. We treat the corresponding topology changes induced by the corresponding internal or external events. After the given number of steps, we reset by shifting the remaining outline back in one large step of step-size times number of steps. Then, we start going outward for the closing, again step by step and checking for short segments and distances. We again reset after the given number of steps. For the discrete curvature-space, which is the last part, we only shift segments shorter than the minimal length. We only check for very short segments and not for distances as, in discrete curvature-space, no external events arise. Finally, we rescale the resulting outline so that the original area is preserved. We do so by connecting the vertices to the centre of gravity, then scaling the



(*a*) (*b*)

Figure 9. Examples for external events: (*a*) narrow connection between buildings⇒two buildings; (*b*) narrow passage⇒Inner courtyard; for (*a*) and (*b*), the same sequence of operators and same parameters as for figure 8 were used.

resulting vectors to obtain the original area. Additionally, we shift the centre of gravity of the resulting object to the centre of gravity of the original object. This re-scaling explains the fact that in all the above figures, most notably in figure 8(*b*), the outline of the building is slightly shifted.

Without shifting, the elimination of parts can give the impression that free space exists where there actually have been parts of the building. Shifting moves the building to an average position which, in many instances, suitably represents the building. For re-scaling, it is less clear whether the original area (or volume in 3D) or the area after scale-space-based generalization is more suitable, though we prefer the original area.

3.4 *Squaring of non-orthogonal structures in 2D*

In many cases, the angles between segments in the given input data are close to 90° or 180°. Based on the incremental processing necessary for the treatment of the scale-space events, a stepwise squaring of angles in the intervals [45°, 90°) and (90°, 135°] or an elimination of angles in the intervals (0°, 45°) and (135°, 180°) can be achieved.

We use a semi-circle or a straight line, respectively. Connecting the end-points of a semicircle with an arbitrary point on the semi-circle always results in a right angle at this arbitrary point. We make use of this fact as follows (cf. figure 10 a)): The preceding point P_{i-1} and the subsequent point P_{i+1} of a point P_i with the non-right angle are connected $\left(\overline{P_{i-1}P_{i+1}}\right)$. The centre point PM of $\overline{P_{i-1}P_{i+1}}$ is connected to the point to be shifted, thus obtaining a vector $\overrightarrow{P_M P_i}$. P_i is shifted incrementally with a given step-size in the direction $\overrightarrow{P_M P_i}$ such that the distance to P_M approaches $|P_{i-1}P_{i+1}|/2$. Figure 10(*b*) illustrates how non-orthogonal angles are incrementally squared.

4. Generalization of building outlines in 3D

In this paper, we assume that 3D buildings are polyhedra composed of orientable planar surfaces. Additionally, we expect that buildings are often orthogonal. This means that when structures of a building are not rectangular at the beginning of the generalization, they have to be squared (cf. section 4.4). Though this is a restricted model for buildings, it is still general enough to reveal the potential of scale-space-based generalization.

Modelling the geometry and especially the topology of polyhedra is a highly intricate task. Therefore, a computational geometry tool box, particularly the Computational Geometry Algorithm Library (CGAL; www.cgal.org), was used. For the representation of input and output data, the VRML '97 standard and the

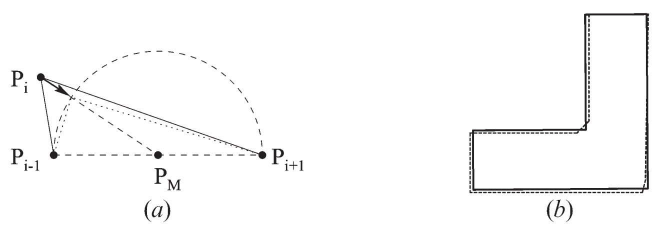


Figure 10. (*a*) Squaring of angles based on a semi-circle; (*b*) result for squaring a building with non-right angles.

vrwave viewer were used. This makes it possible to present not only the resulting images but also the complete 3D building data on the Internet (serv.photo.verm.tu-muenchen.de/b3d/).

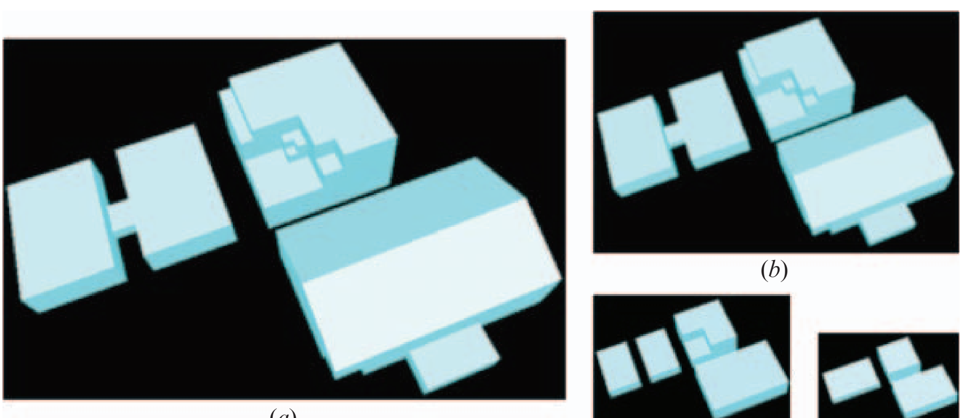
4.1 *Generation of a 3D LOD representation*

The ultimate goal of our generalization in 3D is the automatic generation of a LOD representation for the visualization of buildings. An example is given in figure 11. A narrow connection between two parts of a building is eliminated at a medium scale, and the two buildings are merged into one building at a coarse scale (left building). Different step- or stair-structures as well as inward and outward pointing box-structures are eliminated (centre building). Finally, the non-rectangular roof of a building is squared while eliminating an inward pointing box and an annex (right building).

The CGAL structure for polyhedra relies on a half-edge data structure. It is based on vertices, halfedges, and facets. On the other hand, the VRML '97 data structure 'IndexedFaceSet' consists of 3D points and facets defined by lists of connected points. Therefore, the input is straightforward. Only the orientation of the facets might be inconsistent: some might point inward and some outward. As the halfedges by definition have to be oriented in opposite directions, a contradicting orientation of a facet is immediately discovered, and the facet is reversed.

One of the most important features of the half-edge data structure in CGAL is that it provides Euler operators which enable incremental topological modifications of an object when scale-space events occur. Important Euler operators are, for instance, split of a facet into two new facets, join of two facets, split of a vertex, join of two vertices, split of a loop, and join of two loops. The first four operators are mainly useful to deal with internal events, while the latter two are very important to handle the external events. The split of a loop makes it possible, for instance, to split a polyhedron along a cycle of edges into two new polyhedra. Last but not least, there are also Euler operators that can deal with holes such as to create and fill them.

In the following sections, a uniform step-size of 0.5 units was used for all examples. The sequence of operators for the examples on mathematical morphology



(*a*) (*b*) (*c*) (*d*)

Figure 11. 3D level of detail (LOD) representation: (*a*) input data; (*b*)–(*d*) first to third level.

was always opening followed by closing, and the same number of steps (=4) was used when not stated otherwise. For the discrete curvature-space presented from figure 16 onwards, a larger number of steps, namely 12, was used, and the minimal length was set to 5 units. Here, analogously to 2D (cf. section 3.3), the volumes of buildings are re-scaled at the end.

4.2 *Opening and closing*

Transferring opening and closing from 2D to 3D is straightforward. Opening or closing in 3D is synonymous with starting to move the facets in the direction of the normals inward or outward, respectively. As all the normals are oriented outward, this is fairly easy as long as a building is orthogonal. For problems with non-orthogonal buildings, see section 4.4. An example is given in figure 12. In figure 12(*c*), the input is presented as a white wire frame, and the four steps of incremental inward-going opening and outward-going closing are drawn as red wire frames. The final result is shown as a green solid (cf. also figure 12(*b*)) where the gap in the centre is closed. The scale-space event that occurs when the facets collide while closing is marked by the blue facet in figure 12(*c*).

Examples for events while opening are presented in figure 13 as well as figure 14(*b*) and (*c*). Figure 13 shows how the facets of the annex collide, and therefore the latter is abandoned. In figure 14, the connecting part is removed, resulting in two separate buildings in figure 14(*b*) (external event).

Figures 14(*d*) and (*e*) show, in addition to the split, which also happens in this case, another external event which occurs when closing the building with the doubled number of steps, which also means a doubled scale-parameter compared with figure 14(*b*) and (*c*). While the height of the connection was 2 units, which corresponds to 4 steps of 0.5 units, the distance between the buildings is 4 units, that is 8 steps times 0.5 units. The event in figures 14(*d*) and (*e*) is similar to what happens with the building in Figure 15. However, for the latter, the external event results in a hole in the building (‘Doughnut’), while for the former, the two buildings are merged.

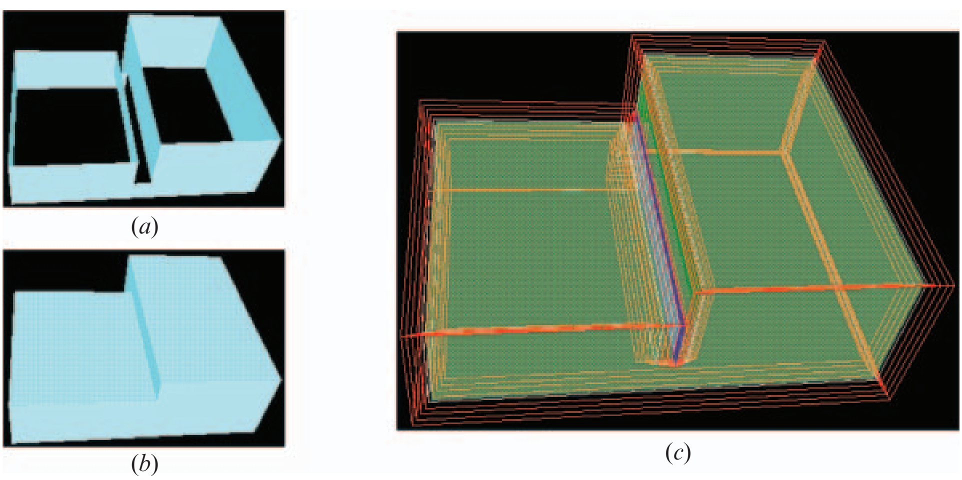


Figure 12. Event while closing: (*a*) input data; (*b*) result; (*c*) four steps of incremental processing. Input: white; opening and closing: red; scale-space event: blue facet; result: green solid.

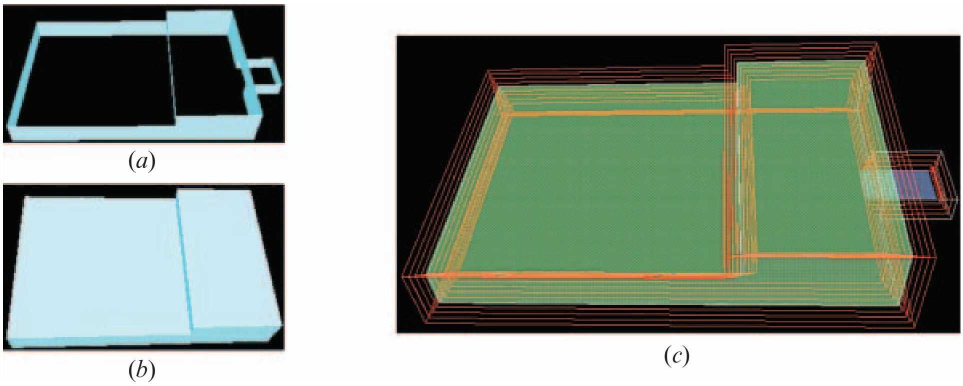


Figure 13. Event while opening: Elimination of the annex on the right side: (*a*) input data; (*b*) result; (*c*) four steps of incremental processing; explanation: cf. figure 12.

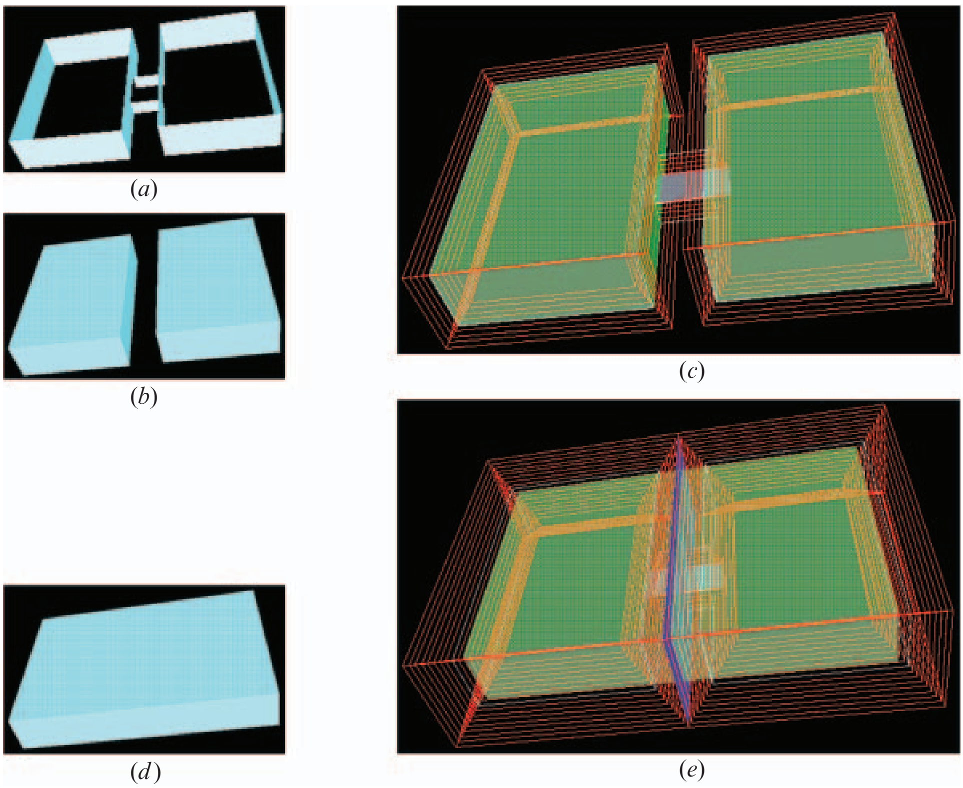


Figure 14. External events while opening when using four steps and while closing when using eight steps, that is doubled scale-parameter: split and merge. (*a*) Input data; (*b*) result split; (*c*) four steps of incremental processing for split; (*d*) final result (merge); (*e*) eight steps of incremental processing for split and merge; explanation for (*c*) and (*e*): cf. figure 12.

4.3 *Discrete curvature-space*

The basic consideration for discrete curvature-space in 3D is how to shift the facets. Figure 16 can be seen as the result of figure 13. Like in 2D, a Z-shaped structure, which we call 'stair', has to be eliminated. This is done by averaging the lower part

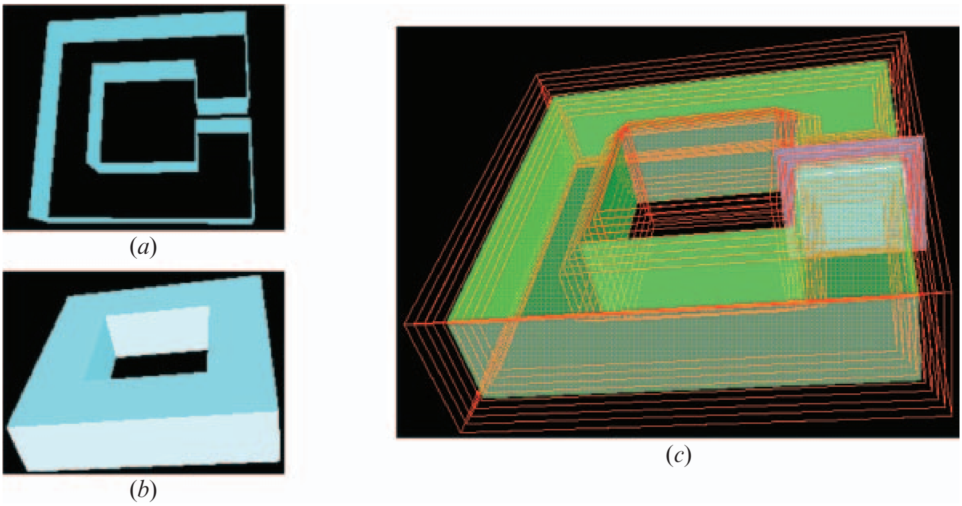


Figure 15. External event while closing: doughnut (*a*) input data; (*b*) result; (*c*) four steps of incremental processing; explanation: cf. figure 12.

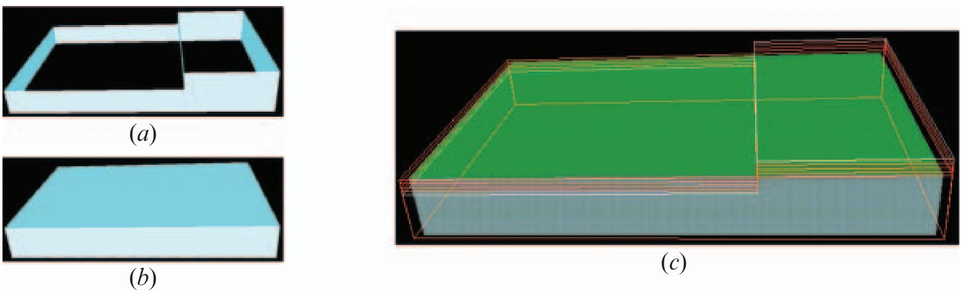


Figure 16. Event in discrete curvature-space: stair (*a*) input data; (*b*) result; (*c*) 12 steps of incremental processing of which only four have been effective because then the short concave segments cease to exist; explanation: cf. figure 12.

on the left side and the higher part on the right side. Figure 17(*a*) illustrates a closer view of the stair. In principle, the shift of the facets is controlled by the relations of their 3D normals. Here, these depend on the 2D polygon defining the neighbouring facet (cf. figure 17(*b*)).

Moreover, as can be seen from figure 18(*a*), the detection of a stair can be based on concave segments. These can be determined by comparing the orientation of the triangle constructed from three successive points in the ordered list of polygon

Figure 17. (*a*) Normals of stair in 3D (red cylinders) directly depend on (*b*) the polygon defining the neighbouring facet.

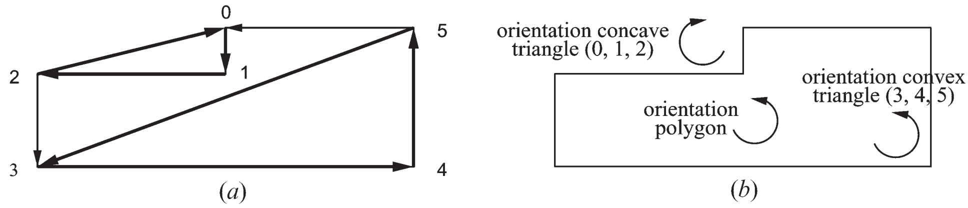


Figure 18. Determination of concave segments: the orientation of the triangle in (*a*) constructed from the successive points 0, 1, 2 defining two concave segments is clockwise as indicated in (*b*). Opposed to this, the orientation of the triangle constructed from the successive points 3, 4, 5 defining two convex segments is counterclockwise, which is the same as the orientation of the whole polygon.

points with the orientation of the whole polygon (cf. figure 18(*b*)): if the orientations are different from each other, the three points define two concave segments, which indicates a stair existing in 3D.

In figure 19, three different kinds of structures, which can only be eliminated by discrete curvature-space, are shown: (1) the small stair at the upper left side; (2) the inward pointing box at the right side bounded from three sides; and (3) an outward pointing box in the centre. On this occasion, we had to introduce the first modification compared with 2D: in figure 19(*c*), there is a 'short concave segment' indicated by an arrow. As this segment is shorter than the minimum length, the corresponding facet, which is the facet marked as 'ground plane' of the inward pointing box, will be shifted according to the definition of discrete curvature-space. Yet we know from expert knowledge that it is desirable to preserve large structures with a small value of the scale-parameter, that is a small number of steps. In order to keep the shift small for the ground plane, the shift is weighted with the inverse of the

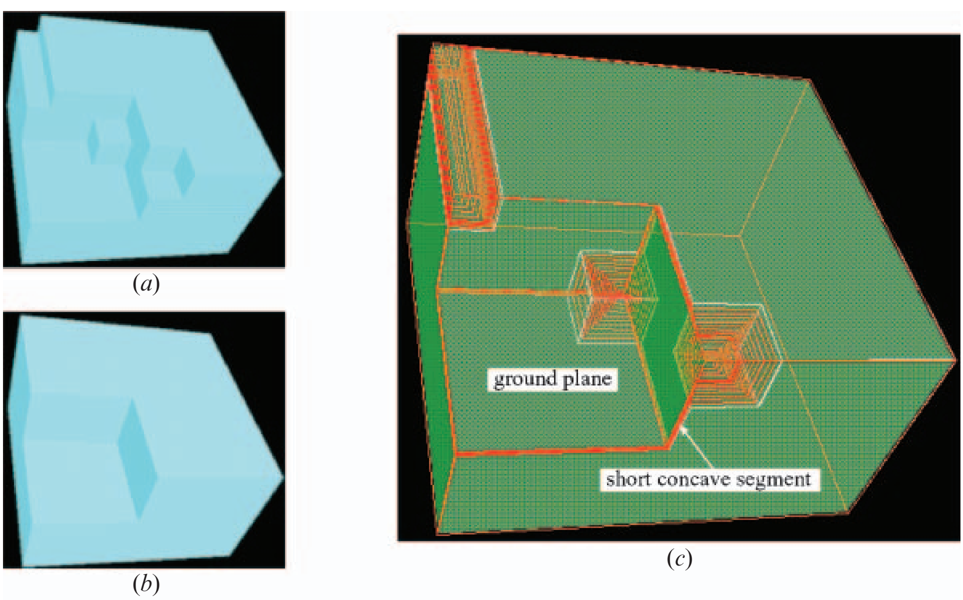


Figure 19. Events in discrete curvature-space: one stair and two boxes: (*a*) input data; (*b*) result; (*c*) 12 processing steps of discrete curvature-space; explanation: cf. figure 12.

area of the facet. Thus, small facets move fast, while large facets move little or not at all. Because the weighting affects the effective step-size, the number of steps for discrete curvature-space is higher than for opening and closing in 3D.

4.4 *Combination of opening, closing, discrete curvature-space, and squaring*

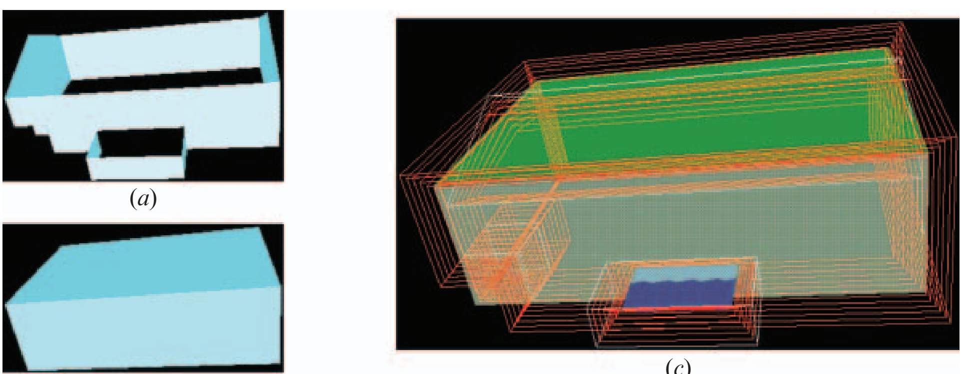
We use 2D heuristic squaring to treat non-rectangular corners or obtuse angles between facets in 3D. Unfortunately, if opposite facets to be squared are not mirror symmetrical, points are shifted differently, and the whole object becomes sheared. This is due to the different side lengths of the triangle inscribed to the semi-circle used for squaring in section 3.4, leading to different directions of movement. This problem can be partially alleviated by moving the points in the direction of the bisector of the enclosing segments.

To avoid manual selection, it is very useful to keep the same order of operations. For this it should be noted that in the examples for discrete curvature-space (cf. section 4.3) the same results are obtained by a combination of opening, closing, and discrete curvature-space in exactly that order. This order of operations has also been kept for the next example and for all the examples presented in figure 11.

In figure 20, a combination of the proposed scale-spaces with squaring is presented. The squaring is done in parallel with opening and closing. The annex in the centre and the inward pointing box of the entrance at the left side are eliminated by opening and discrete curvature space, respectively. In parallel with opening and closing, the sloping roof is squared.

To give an idea about the dependence of the results on parameter values, table 1 presents the parameters and their values used for the generation of the LOD representation shown in figure 11. One can see that the minimal length is proportional to the number of steps used for discrete curvature-space. While for figure 11(*b*) the effects of generalization are small, with increasing scale-parameter, or number of steps, the buildings are considerably simplified in 11(*c*) and especially 11(*d*), where the number of steps for all operators is again doubled.

Finally, also the complexity of the algorithm is of interest. Here, we restrict ourselves to time complexity. Basically, the complexity of shifting varies linearly with the number of segments in 2D and facets in 3D. The most critical part in terms



(*a*) (*b*) (*c*)

Figure 20. Squaring of the roof, elimination of the annex and the inward pointing box of the entrance: (*a*) input data; (*b*) result; (*c*) four steps for opening and closing and 12 steps for discrete curvature-space; explanation: cf. figure 12.

Table 1. Parameters used to generate figure 11 (step-size in all cases: 0.5 units).

Part	Steps opening and closing	Steps discrete curvature-space	Minimal length
b	2	6	2.5
c	4	12	5
d	8	24	10

of complexity is the check for distances of vertices to non-local segments or facets of a building or distances between vertices of a building to another building needed for the determination of external events. The complexity of this task grows exponentially, though it can be reduced considerably by using elaborate data structures. The complexity of handling the events varies linearly with the involved segments or facets, the number of which is usually small.

5. Discussion

This paper demonstrates a variety of techniques which can be used to generalize different types of 2D and 3D vector building data. Some ideas have been developed that can overcome known weaknesses of the proposed approach.

We found that a fixed sequence of opening, closing, and discrete curvature-space with fixed parameters leads to acceptable results for a variety of buildings. The most important reason for this is that mathematical morphology and discrete curvature-space are complementary: while discrete curvature-space is the only means to eliminate stair- or (inward-pointing) box structures, it cannot be used to aggregate or split objects which is the strength of mathematical morphology. In spite of the mostly satisfying results, the modification of the sequence and the parameters (cf. section 4.4) in conjunction with a high-level, e.g. rule-based, control is of high importance to make our operators useful for practical applications.

Squaring is the geometrical problem least understood in this paper: When one makes one facet orthogonal in 2D, there is a strong tendency for the neighbouring facets to become sheared. A 3D version based on the idea of making the angles at the vertices orthogonal was tested. Yet we discarded it because of the same shortcomings as for the initial 2D solution. The outcome of the analysis mostly points to an all too local treatment of the problem. The final idea, which was not implemented, involves using the facets as the basic element of the squaring. Depending on their area, neighbouring facets can be incrementally rotated until they are finally parallel or orthogonal.

A major design criterion, but a weakness of the approach, is its focus on rectangular structures. This is useful for many buildings and their aggregations in Europe and North America. However, different models are needed for other parts of the world and for specific buildings; in particular, the scale-space operators have to be adapted. For instance, the generalization of steps of stairs into bigger steps, thereby maintaining the characteristics of a building, is appropriate for modern block-structures. For a Roman amphitheatre with its long stairs, however, a sloped surface is a better approximation. These kinds of problem stimulate us to acquire higher-level knowledge of the object semantics that should control the generalization.

In our recent research, we try to avoid the additional artificial parameter in the form of the threshold for discrete curvature-space. The goal is to generate a continuous version of curvature space on the basis of a simpler theory that can be implemented more easily. As no constraint for the shift of the facets is available, we let all facets move. The basic movement goes inward, so we add the morphological component, in this case erosion, as in the reaction-diffusion space of Kimia *et al.* (1995). In the same way as for the discrete curvature-space (cf. section 4.3) it is desirable to preserve large structures with a small scale-parameter. We found that we can control the shift of the facets fairly well by a weighting function depending on the length of the concave segments and the area of a facet (for instance $\text{length}^2 \times \text{area}$). The results are comparable with those for discrete curvature-space with an appropriately selected threshold, yet with the advantage that one less parameter value has to be specified. Finally, it is noteworthy that the continuous curvature-space can generate results at arbitrary scales without having to run it several times with different thresholds. This feature is useful for a smooth transition during visualization.

The work presented here was hindered considerably by the laborious effort needed to manipulate the 3D geometry based on the Euler operators of CGAL. Recently, we have started a project which will follow the lines of this paper based on the 3D modelling system ACIS (www.spatial.com). In ACIS and similar systems, the incremental modifications can be handled much more efficiently. To avoid the lessons we had to learn, this kind of system is therefore highly recommended for research in 3D generalization.

6. Conclusions

In summary, the mathematical morphology which was introduced for generalization by Li (1996), and has been applied for the generalization of building data most notably by Su *et al.* (1997) and Cámara and López (2000), was implemented for vector data in 2D as well as 3D.

We have also shown that from the point of view of scale-spaces, generalization takes place by means of internal and external scale-space events (for instance, the elimination of an annex and the split of a building in two parts) which can be handled by (at least in 2D) simple basic operators when using incremental processing.

We only use relatively few parameters with a precisely defined semantics, namely the step size, the number of steps for opening, closing, and discrete curvature-space as well as one flag defining if right angles should be enforced.

By and large, a major advantage of our approach to generalization is that it is based on mathematically sound scale-space theory. This is true with mathematical morphology for which the correspondence to raster-based mathematical morphology was shown by Kimia *et al.* (1995). For discrete curvature-space, the connection is weaker, as we have made some modifications there.

Above all, basing generalization on scale-spaces contributes to the detection and development of the 'missing' operators for generalization (Mackaness *et al.* 1997) particularly for 3D. Besides, generalization is at least an implicit step for the development of methods for the recognition of structures: objects are more comparable after simplification. Again, here we contribute operators for 3D.

Acknowledgements

This work was supported by Deutsche Forschungsgemeinschaft research grant MA 1651/5-1. We want to thank the editors and the anonymous reviewers for their helpful comments and we are grateful to Liqiu Meng for proofreading.

References

Bobick, A. and Bolles, R., 1992, The representation space paradigm of concurrent evolving object descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(2), pp. 146–156.

Cámara, M. and López, F., 2000, Mathematical morphology applied to raster generalization of urban city block maps. *Cartographica*, **37**(1), pp. 33–48.

Garland, M. and Heckbert, P., 1997, Surface simplification using quadric error metrics. In *SIGGRAPH '97*, pp. 209–216.

Heckbert, P. and Garland, M., 1997, survey of polygonal simplification algorithms. In *Multiresolution Surface Modeling Course SIGGRAPH '97*. Available online at: www.cs.cmu.edu/~ph (accessed 8 June 2005).

Kilpeläinen, T., 2000, Maintenance of multiple representation databases for topographic data. *The Cartographic Journal*, **37**(2), pp. 101–107.

Kimia, B., Tannenbaum, A. and Zucker, S., 1995, Shapes, shocks, and deformations I: the components of two-dimensional shape and the reaction-diffussion space. *International Journal of Computer Vision*, **15**(3), pp. 189–224.

Koenderink, J., 1984, The structure of images. *Biological Cybernetics*, **50**, pp. 363–370.

Kosslyn, S., 1994, *Image and Brain* (Cambridge, MA: MIT Press).

Li, Z., 1996, Transformation of spatial representation in scale dimension. *International Archives of Photogrammetry and Remote Sensing*, **31**(B3/III), pp. 453–458.

Li, Z. and Su, B., 1997, Some basic mathematical models for feature displacement in digital map generalization. In *International Cartographic Conference*, Vol. I, pp. 452–459.

Lindeberg, T., 1994, *Scale-Space Theory in Computer Vision* (Boston, MA: Kluwer Academic).

Mackaness, W., Weibel, R. and Buttenfield, B., 1997, Report of the ICA Workshop on Map Generalization. In *Publication of the ICA Working Group on Map Generalization* (Gävle, Sweden). Available online at: www.geo.unizh.ch/ICA/ (accessed 8 June 2005).

Malladi, R. and Sethian, J., 1996, Flows under min/max curvature flow and mean curvature: application in image processing. In *Fourth European Conference on Computer Vision*, Vol. I, pp. 251–262.

Mayer, H., 1996, Abstraction and scale-space events in image understanding. *International Archives of Photogrammetry and Remote Sensing*, **31**(B3/III), pp. 523–528.

Mayer, H. and Steger, C., 1998, Scale-space events and their link to abstraction for road extraction. *ISPRS Journal of Photogrammetry and Remote Sensing*, **53**, pp. 62–75.

Meyer, U., 1989, *Generalisierung der Siedlungsdarstellung in digitalen Situationsmodellen*, Vol. 159 (Hannover, Germany: Wissenschaftliche Arbeiten der Fachrichtung Vermessungswesen der Universität Hannover).

Molenaar, M., 1996, Multi-scale approaches for geodata. *International Archives of Photogrammetry and Remote Sensing*, **31**(B3/III), pp. 542–554.

Müller, J., Weibel, R., Lagrange, J. and Salgé, F., 1995, Generalization: state of the art and issues. In *GIS and Generalization—Methodology and Practice* (London: Taylor & Francis), pp. 3–17.

Popović, J. and Hoppe, H., 1997, Progressive simplicial complexes. In *SIGGRAPH '97*, pp. 217–224.

Regnauld, N., Edwardes, A. and Barrault, M., 1999, Strategies in building generalization: modelling the sequence, constraining the choice. In *Progress in Automated Map Generalization—ACI '99*.

Ribelles, J., Heckbert, P., Garland, M., Stahovich, T. and Srivastava, V., 2001, Finding and removing features from polyhedra. In *ASME Design Engineering Technical Conferences*.

Ruas, A., 1999, The role of meso level for urban generalization. In *Progress in Automated Map Generalization—ACI '99*.

SCHMALSTIEG, D., 1996, *LODESTAR: An Octree-Based Level of Detail Generator for VRML, Technical Report*. Institute of Computer Graphics, Visualization and Animation Group, Vienna University of Technology.

SCHOPPMEYER, J. and HEISSER, M., 1995, Behandlung von Geometrietypwechseln in GIS. *Nachrichten aus dem Karten- und Vermessungswesen*, **I**(113), pp. 209–224.

SERRA, J., 1982, *Image Analysis and Mathematical Morphology* (London: Academic Press).

SESTER, M., 2000, Generalization based on least squares adjustment. *International Archives of Photogrammetry and Remote Sensing*, **33**(B4/3), pp. 931–938.

STAUFENBIEL, W., 1973, *Zur Automation der Generalisierung topographischer Karten mit besonderer Berücksichtigung großmaßstäblicher Gebäudedarstellungen*, Vol. 51 (Hannover, Germany: Wissenschaftliche Arbeiten der Fachrichtung Vermessungswesen der Universität Hannover).

STILLA, U., MICHAELSEN, E. and JURKIEWICZ, K., 1998, Structural analysis of right-angled building contours. *International Archives of Photogrammetry and Remote Sensing*, **32**(3/1), pp. 379–386.

SU, B. and LI, Z., 1997, Morphological transformation for detecting spatial conflicts in digital generalization. In *International Cartographic Conference*, Vol. I, pp. 460–467.

SU, B., LI, Z. and LODWICK, G., 1998, Morphological models for the collapse for area features in digital map generalization. *GeoInformatica*, **2**(4), pp. 359–382.

SU, B., LI, Z., LODWICK, G. and MÜLLER, J.-C., 1997, Algebraic models for the aggregation of area features based upon morphological operators. *International Journal of Geographical Information Science*, **11**(3), pp. 233–246.

TIMPF, S., 1999, Abstraction, levels of detail, and hierarchies in map series. In *Spatial Information Theory—Cognitive and Computational Foundations of Geographic Information Science*, pp. 125–139 (Berlin: Springer).

TIMPF, S. and FRANK, A., 1997, Exploring the life of screen objects. In Auto-Carto 13ACSMASPRS, pp. 194–203.

VARSHNEY, A., AGARWAL, P., BROOKS, F., WRIGHT, W. and WEBER, H., 1995, *Generating Levels of Detail for Large-Scale Polygonal Models. Technical Report CS-1995-20*, Department of Computer Science, Duke University, USA.

WITKIN, A.P., 1983, Scale-space filtering. In *Eighth International Joint Conference on Artificial Intelligence*, pp. 1019–1022.