Scale-Dependent Simplification of 3D Building Models Based on Cell Decomposition and Primitive Instancing

Martin Kada

Institute for Photogrammetry, Universität Stuttgart,
Geschwister-Scholl-Str. 24D, 70174 Stuttgart, Germany
martin.kada@ifp.uni-stuttgart.de

Abstract. The paper proposes a novel approach for a scale-dependent geometric simplification of 3D building models that are an integral part of virtual cities. In contrast to real-time photorealistic visualisations, map-like presentations emphasize the specific cartographic properties of objects. For buildings objects, such properties are e.g. the parallel and right-angled arrangements of facade walls and the symmetries of the roof structure. To a map, a clear visual perception of the spatial situation is more important than a detailed reflection of reality. Therefore, the simplification of a 3D building model must be the transformation of the object into its global shape. We present a two-stage algorithm for such an object-specific simplification, which combines primitive instancing and cell decomposition to recreate a basic building model that best fits the objects original shape.

Keywords: 3D city models, generalization, levels of detail, algorithm

1 Introduction

The acquisition and presentation of 3D city models has been a topic of intensive research for more than 15 years. In general, such data sets include digital representations of the landscape, the buildings and more frequently also of the vegetation and the street furniture. A number of commercial software products and service companies exist nowadays for the reconstruction of buildings. For an efficient data collection of large areas, the objects are measured from aerial images or laser data. Therefore there is no façade information available in the source data which results in building models where the ground plan is simply extruded and intersected with the interpreted roof structure.

Besides the traditional analysis applications of 3D city models, which are e.g. the planning of mobile antennas, alignment of solar installations and noise propagation, the presentation of urban areas gains in importance. Real-time and web-based visualisation systems offer nowadays graphics of near photorealistic quality. To limit the amount of data that needs to be transferred over the network and to increase rendering performance, objects are represented in different levels of detail depending on their distance to the viewer. For 3D city models, the following classification of building objects in three discrete levels of detail is very common: block models with flat roofs and no facade structure, models with roof structures and architectural models with detailed roofs and facades. So far, cities have mostly collected data in the second level of detail with only a few selected landmarks being of higher detail. Because of the high costs involved in the acquisition, there are efforts to facilitate the exchange and interoperability between data and application providers. The Special Interest Group 3D (SIG 3D) of the initiative Geodata Infrastructure North-Rhine Westphalia (GDI NRW), Germany, e.g., proposed the application schema CityGML to the Open Geospatial Consortium (OGC) for standardisation [1]. Therein, the properties of five levels of detail are defined to support a broad variety of applications like car navigation systems, flight, driving and nautical simulators, tourism information systems, etc. A preliminary survey lists applications of 3D city models and their specific levels of detail requirements [2].

A photorealistic visualisation is not always the most adequate tool to communicate spatial information. Architects and designers often produce sketch like hardcopy outputs to make their objects appear more alive or to express the preliminary status of their designs. Recent works on interactive visualisations of 3D city models (e.g. [3]) explore non-photorealistic rendering techniques that imitate this style so that spatial situations are easier to perceive and comprehend. Such techniques, however, rely on information about the characteristic edges that best reflect the global shape of a building. This is basically what results from a cartographic simplification.

Another field of application for 3D city models are location based services or context-aware applications. Their users rely heavily on a location- or situation-dependent presentation of the information that is most relevant to their current task. To be useful anywhere at all times, such systems run on mobile devices like digital personal assistants (PDA) or mobile phones. As their screen size and resolution will always be a limiting factor, a geometric simplification of 3D objects is necessary to guarantee the graphical minimum feature size required by maps or map-like presentations. Otherwise the high line density makes it impossible to recognize important aspects of the building object.

Because it is not reasonable to collect and store data for all requested levels of detail, an automatic process is necessary that transforms 3D building models towards more simplified shapes. Object features that are under a minimum size, which can be determined from the scale parameters of the map projection, should be removed without disturbing the global shape. Properties that are specific for the object itself as well as the object type, however, must be preserved. In the case of 3D building models, these are the parallel and right-angled arrangements of facade walls and the symmetries of the roof structures. Object specific features are especially important for landmarks. The simplified model of a church or cathedral, e.g., must not miss its towers after generalisation as otherwise the object is hardly recognisable anymore.

A simplification of solitary objects under these spatial constraints is one of the elemental operators of cartographic generalisation. In cartography, both the spatial objects themselves as well as their arrangement are transformed with the goal to create maps or map-like presentations that help to communicate a spatial situation. Other generalisation operators omit or emphasise objects depending on their importance, aggregate semantically similar objects, replace a number of objects by fewer entities or displace them to relax the spatial density in areas with many objects. The generation of a situation- and context-dependent abstraction level of the spatial data is therefore possible to help viewers apprehend the presented spatial information.

2 Previous Work

The automatic generalisation of building models has been a research topic ever since Staufenbiel [4] proposed a set of generalisation actions for the iterative simplification of 2D ground plans. Several algorithms have been developed that remove line segments under a pre-defined length by extending and crossing their neighbour segments and by introducing constraints about their angles and minimum distances (e.g. [5, 6, 7, 8, 9]). Other approaches use vector templates [10, 11], morphological operators like opening and closing [12, 13], least-squares adjustment [14] or techniques from scale space theory [15].

A few algorithms also exist by now for the generalisation of 3D data. Forberg [16] adapts the morphology and curvature space operators of the scale space approach to work on 3D building models. Thiemann and Sester [17] do a segmentation of the building's boundary surface with the purpose of generating a hierarchical generalisation tree. After a semantic interpretation of the tree's elements, they can selectively be removed or reorganized to implement the elemental generalisation operators for simplification, emphasis, aggregation and typification. Another aggregation approach is proposed by Anders [18]. It works for linearly arranged building groups. Their 2D silhouettes, which are the results of three projections from orthogonal directions, are simplified, extruded and then intersected to form the generalised 3D model. With a strong focus on the emphasis of landmarks do Thiemann and Sester [19] present adaptive 3D templates. They categorise building models into a limited number of classes with characteristic shapes. A building model is then replaced by the most similar 3D template that is best fit to the real object. Because the semantics of the template is known, the object itself or specific features of the model can be emphasised at will.

The simplification of 3D models has been a major topic in the field of computer graphics. See e.g. the survey of Luebke et al. [20] for an up-to-date summary of the most important work. However, these algorithms are designed for general models that approximate smooth surfaces and therefore typically do not perform well on 3D building models. The main reason is that building models consist of considerably fewer planar faces, but many sharp edges. Coors [21], Rau et al. [22] and Kada [23] show that the simplification operators and metrics can be modified so that the characteristic properties of the building models can be preserved during their simplification.

Despite the number of available 3D generalisation approaches, a continuous difficulty seems to be the simplification of the roof structure. Most algorithms avoid this problem by simply generating flat or pent roofs or assume that the roof type is already available as the result of a preceding interpretation. In this paper, we describe a generalisation approach for 3D building models and concentrate on a new procedural method to generate reasonable roof geometries.

3 Generalisation of 3D Building Models

We propose a two-stage generalisation algorithm for the geometric simplification of solitary 3D building models. As can be seen from the intermediate results of the example in Fig. 1, the two stages consist in a total of five steps. The first stage generates

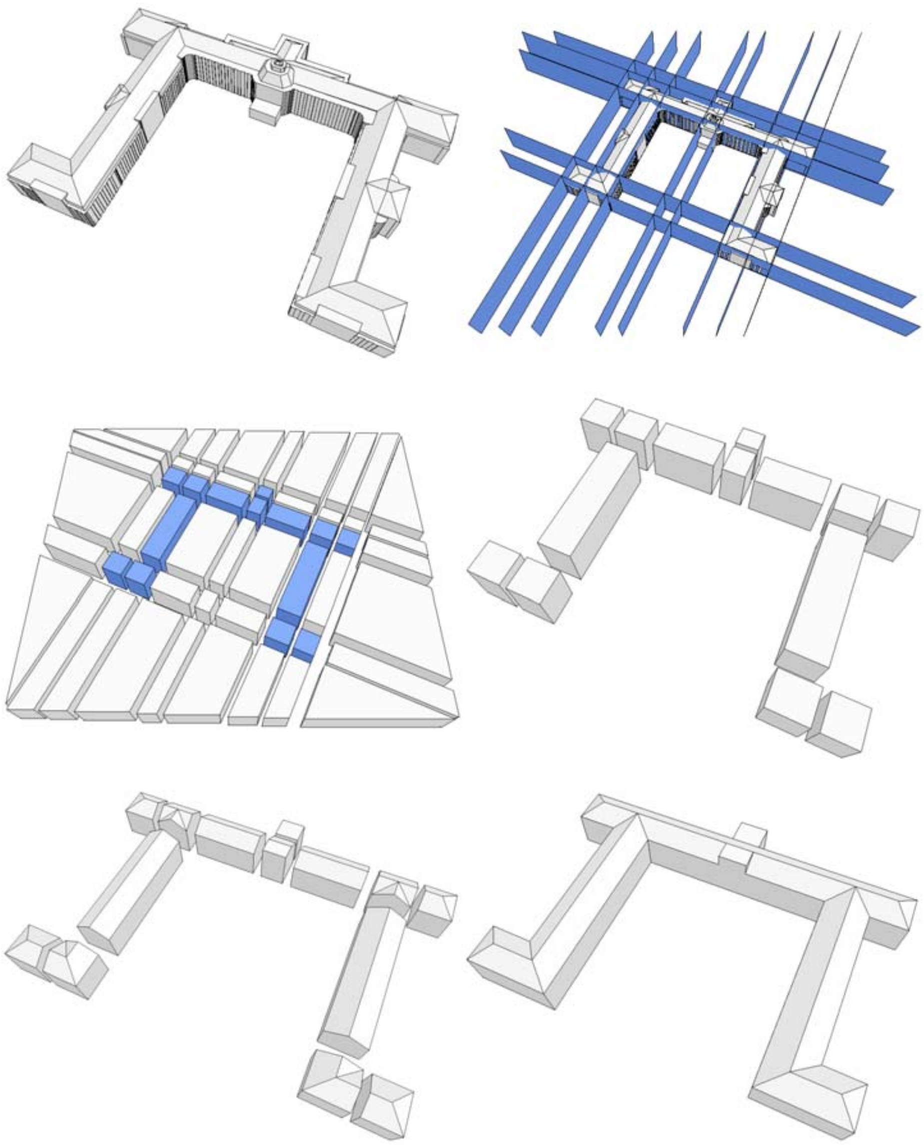


Fig. 1. Original 3D building model (top left) and the five generalisation steps.

a 2D decomposition of space that approximates the ground plan polygon by a disjoint set of quadrilateral primitives. We accomplish this by deriving plane equations from the major façade walls (1), subdividing the infinite space along these planes (2) and identifying the resulting cells that feature a high percentage of overlap with the original ground plan polygon (3). The second stage reconstructs the simplified geometry of the roof. Here, a primitive instancing approach is shown where the roof parameters are determined individually for each cell so that they best fit the original model under distinct adjacency constraints (4). By altering those parameters, the simplification of the roof can be properly adjusted. A union operation of the resulting primitives composes the final 3D building model and concludes the generalisation (5).

4 Ground Plan Cell Decomposition

Cell decomposition is a form of solid modelling in which objects are represented as a collection of arbitrarily shaped 3D primitives that are topologically equivalent to a sphere. The individual cells are usually created as instances from a pre-defined set of parameterized cell types that may even have curved boundary surfaces. Complex solids are then modelled in a bottom-up fashion by "gluing" the simple cells together. However, this operator restricts the cells to be nonintersecting, which means adjoining cells may touch each other but must not share any interior points [24].

In our algorithm, the cell decomposition serves two purposes: First, it is build as an approximation of the building ground plan and is consequently per se also a generalization thereof. Second, it provides the basic building blocks for the reconstruction of the roof geometry. Since the input models are provided as 3D data, all computations are also performed in 3D, even though the dimension of the resulting cells is really 2D; or 2.5D if a height is applied like in the example of Fig. 2. For clarity reasons, however, the accompanying Fig. 2, 3 and 4 are given as 2D sketches.

The faces in a polyhedral building representation are always planar. If the real building facade features round or curved elements, then they must be approximated in the model by small polygons. We therefore generate the cell decomposition by subdividing a finite 3D subspace by a set of vertical planes. Fig. 2 e.g. shows a building and the cell decomposition which results from subdividing space along the facade segments.

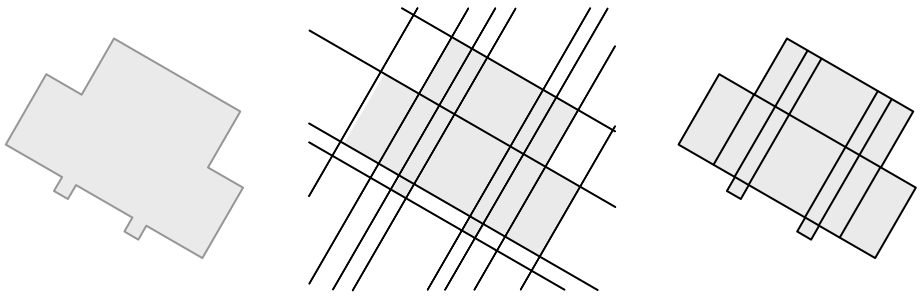


Fig. 2. Building ground plan (left), overlaid decomposition of space along its façade segments (middle) and resulting cell decomposition (right).

As it can be seen, the union of the cells is not yet a simplification of the original shape and the small cells complicate the reconstruction of the roof geometry. So instead of using each individual façade polygon, we cluster them together with a special buffer operation for the purpose of generating fewer planes that in turn produce a decomposition of fewer cells. However, these planes should correspond with the most important facade segments so that the decomposition reflects the characteristic shape of the object. The importance of a plane is measured as the surface area of all polygons that are included in the generating buffer and that are almost parallel to the created plane. Polygons with a different orientation are not counted.

4.1 Generation of Decomposition Planes

We implemented a greedy algorithm that generates the plane of highest importance from a set of input facade polygons. At this point, we ignore all roof polygons and only use polygons with a strict horizontal normal vector. By repeatedly calling the algorithm, new planes are added to the result set and all polygons inside the buffer are discarded from further processing. The generation of planes ends when no input polygons are left or when the importance of the created planes falls under a certain threshold value.

At the beginning of the algorithm, buffers are created from the input polygons (see Fig. 3. Each buffer is defined by two delimiting parallel planes that coincide with the position and normal direction of a generating polygon. These planes may move in opposite directions to increase the buffer area until a generalisation threshold is reached. The buffers are first sorted by their importance and then merged pair wise to create larger buffers. Starting with the buffer of highest importance, the buffers of lower importance are tested for their inclusion in this buffer. If all polygons of a buffer can be included into the one of higher importance without increasing the distance between their delimiting planes above the generalisation value, then the merge is valid and is executed. The algorithm stops when no more buffers can be merged and the averaged plane equation of the polygons of the buffer of highest importance is returned.

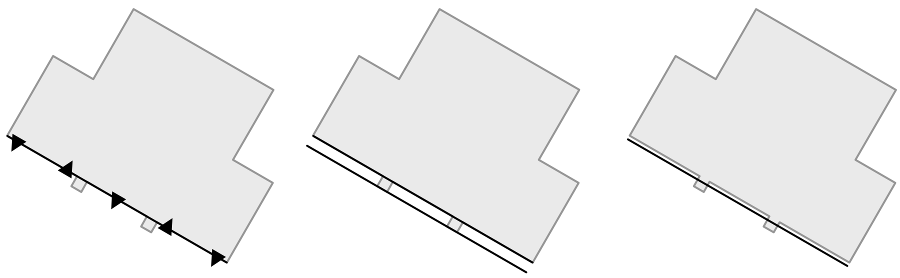


Fig. 3. Initial buffer from facade segments (left), delimiting planes of the maximised buffer (middle) and resulting averaged plane (right).

In order to enforce parallelism and to support right angles of the facade segments, the resulting planes are analysed in a last step. If the angle of the normal vectors from two or more planes is found to be below a certain threshold, these planes are made parallel or rectangular. If the deviation is only a small angle, this can be done by changing the normal vector of the plane equation and adjusting the distance value. For larger values, a rotation of the planes around their weighed centroids of the polygons is chosen.

For our computations, we use four threshold values. The most important one is the generalisation distance that the buffer planes may move apart. As this value also determines the distance of the planes used for the decomposition, it is also approximately the smallest ground plan feature length of the resulting set of cells. Another threshold value determines the lowest importance of a plane that is still a valid result. Here, the square of the generalisation distance is used. Buffers below that value probably do not contain polygons with a side length of the generalisation distance and are therefore not important. The last two threshold values are angles. As it is important for the roof construction that the cells are parallelograms, the angle for enforcing parallelism is rather large. We chose 30° for parallelism and 10° for right angles.

See Fig. 4 for the set of buffers that result in a simplified cell decomposition.

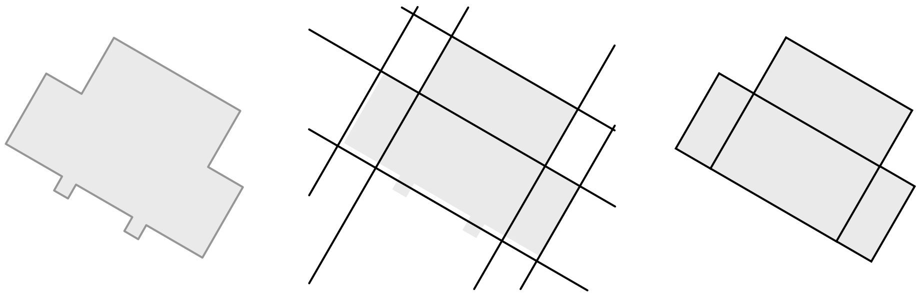


Fig. 4. Building ground plan (left), overlaid simplified decomposition of space along its façade segments (middle) and resulting cell decomposition (right).

4.2 Cell Decomposition

Once the planes have been determined, they are then used to generate the cell decomposition of the building model. Theoretically, an infinite 3D space should be subdivided brute force by the planes. However, as an infinite space is unpractical, a solid two times the size of the building's bounding box is used. Because the plane equations were averaged from facade segments and therefore have no horizontal component, the space is only divided in two dimensions. The resulting cells are therefore 2D polygons extruded into the third dimension.

The decomposition consists of building and non-building cells. Only the building cells are of interest for further processing. The other cells should be discarded. However, these cells can not directly be identified from the decomposition process. Therefore, a further step is necessary.

For that reason, a percentage value is calculated that denotes the overlap of the cell with the original building ground plan. Cells that result in a high overlap value are considered building cells whereas the other cells are considered as non-building cells. A precise value can be computed by intersecting the cell with the ground plan polygon and dividing the resulting area by the area of the cell. As the cells are rather big, an overlap threshold of 50% is able to correctly distinguish between building and non-building cells.

5 Roof Simplification by Cell Decomposition

The roof structure for general 3D building models can be very complex. We therefore present two methods for their simplification. Both recreate a simplified version of the original roof structure for the previously generated ground plan cell decomposition. The first method extends the cell decomposition approach to the third dimension. It is general enough to recreate all roof shapes. As it will be shown in section 6, however, limiting the possible 3D shapes of the cells to a subset of common roof types can lead to a more suitable roof structure for a subset of common buildings.

So far, the roof polygons have been neglected. Now they are used to determine the decomposition planes of arbitrary orientation in order to generate 3D cell decompositions from the ground plan cells. Although the decomposition is done per cell, the planes are determined globally from all roof polygons to ensure that neighbouring cells fit well against each other. We use the buffer approach as previously described. The subdivision process is then done with the subset of planes that has polygons in their buffer that intersected the respective cells. This avoids a heavy fragmentation of the cells.

The resulting cells are now real 3D solids, so the classification in building and non-building cells has to be done in 3D space. Consequently, a percentage value that denotes the volume of the original building model inside each respective cell is computed. Fig. 5 shows the decomposition of the example building of Fig. 1 by the roof planes and the resulting building cells after their identification.

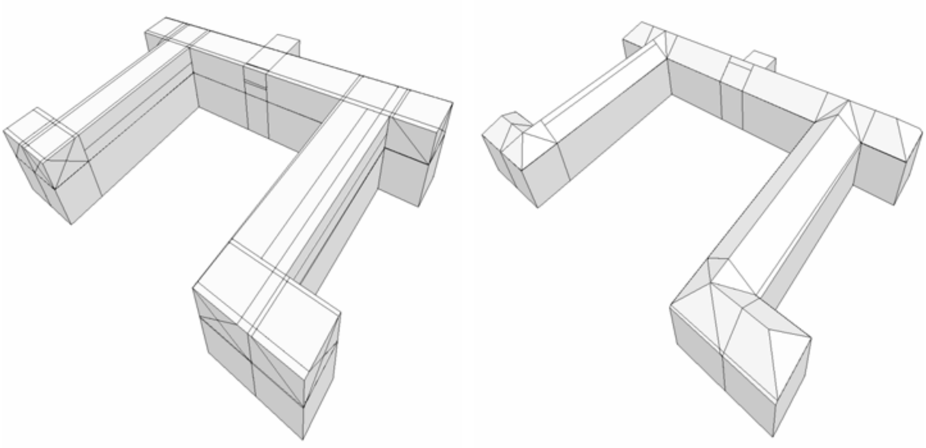


Fig. 5. Decomposition of the roof before (left) and after (right) identification of building cells.

As can be seen in Fig. 5, there are some inaccuracies in the resulting model. These are caused by planes that do not cut the 2.5D cells at exactly the same location in space. We remove these inaccuracies by a vertex contraction process that pulls the roof vertices to the closest ground cell corner point, edge or cell centre if they are within close distance. Fig. 6 and 7 show results of the generalisation algorithm for simple example models as well as rather complex landmarks.

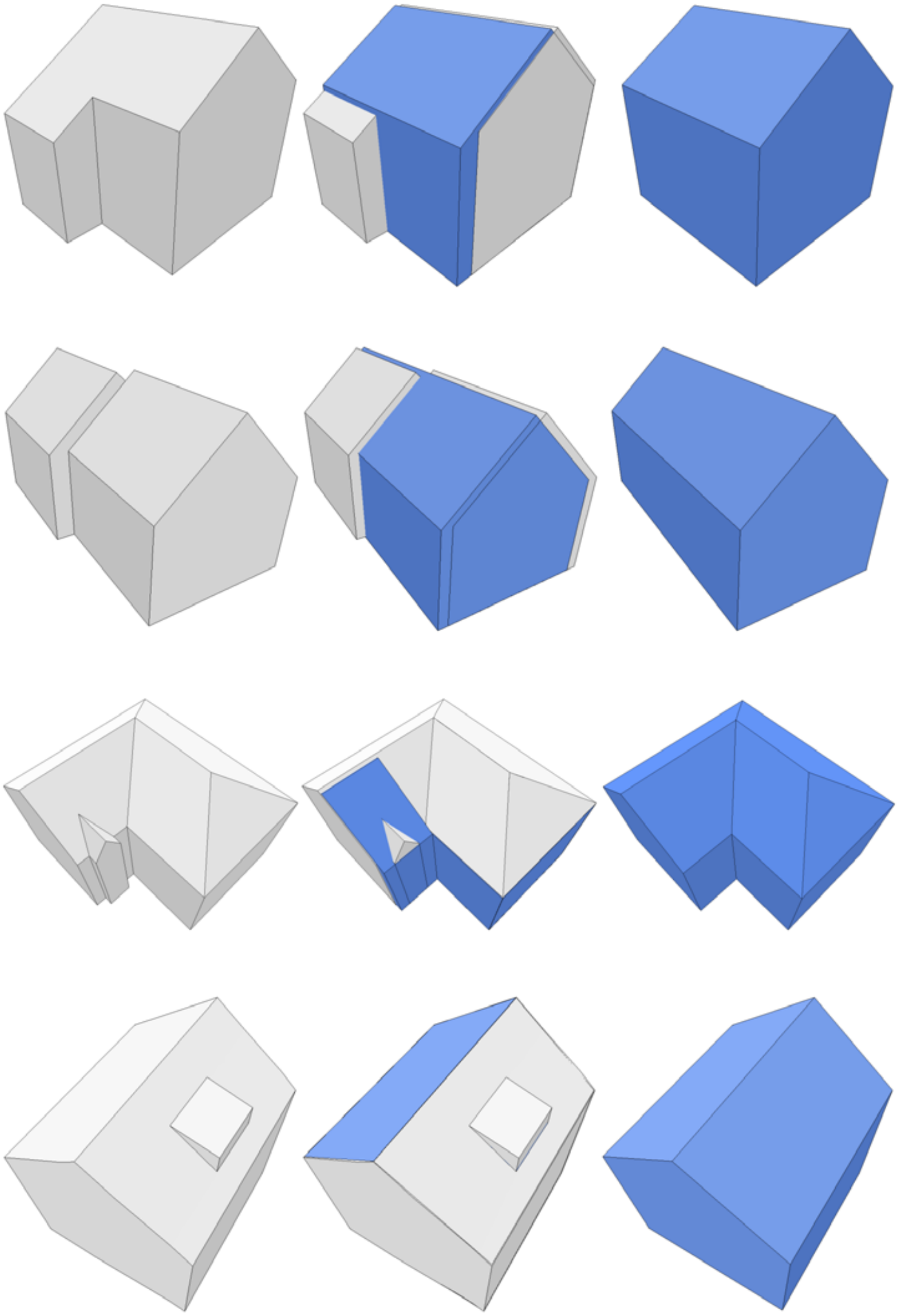


Fig. 6. Original (left) and generalised (right) 3D building models and their overlays (middle).

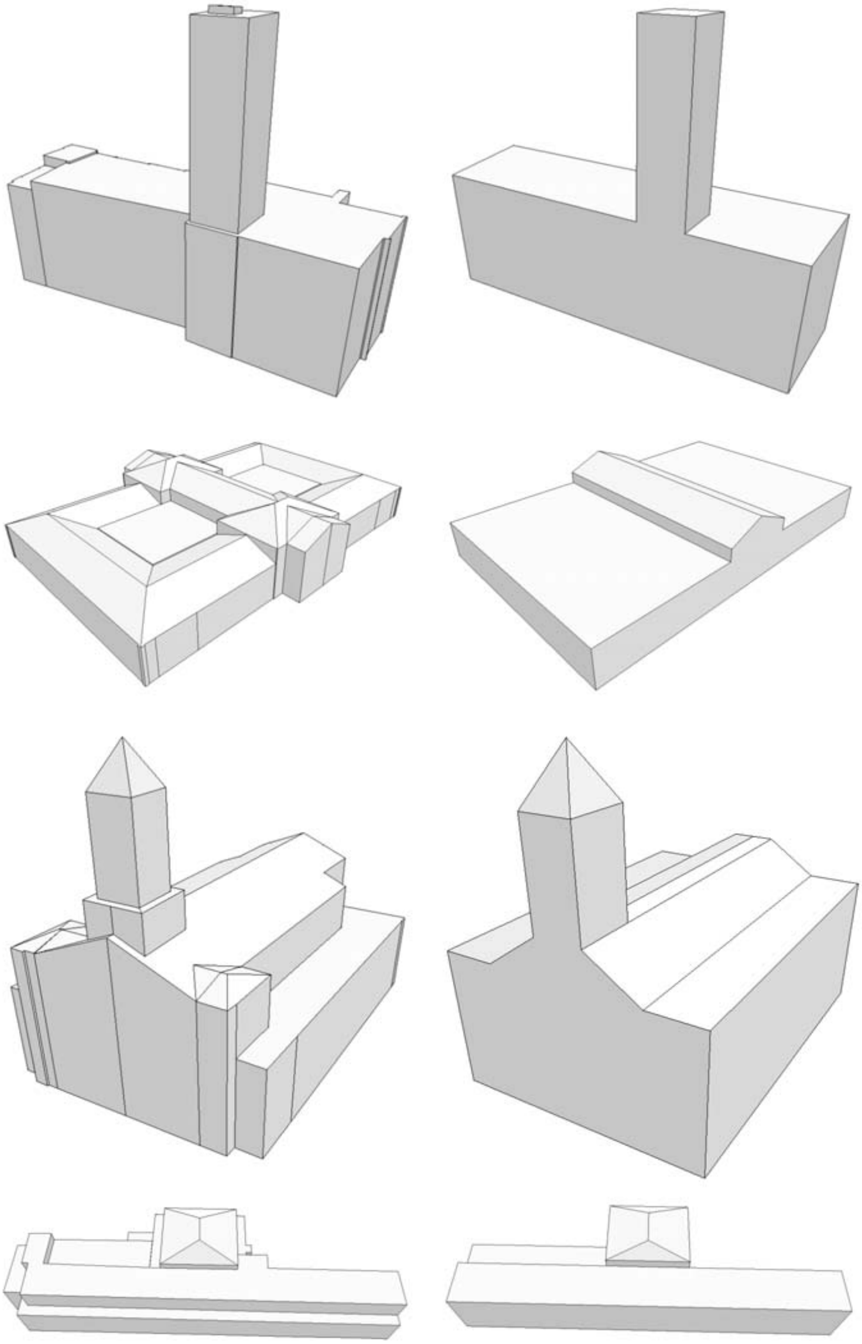


Fig. 7. Original (left) and generalised (right) 3D landmarks.

6 Roof Simplification by Primitive Instancing

The roof simplification via cell decomposition does sometimes not lead to good looking models. This is the consequence of the universal approach where no interpretation of the original roof structure is performed. We present three of the most common shortcomings.

For very flat roof structures, there is only one buffer generated which results in one decomposition plane. As this plane gets the slope of one dominant roof polygon, a shed roof is created (see Fig. 8). A better generalisation would be a gabled roof.

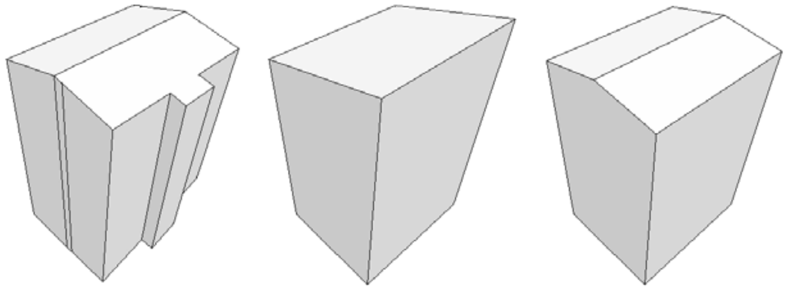


Fig. 8. Original 3D building model (left) and its generalisations via cell decomposition (middle) and primitive instancing (right).

Because the slopes of the decomposition planes are not aligned, the generalisation of hipped roofs often results in an asymmetric roof structure (see Fig. 9). However, a symmetric hipped roof would in most cases be preferred.

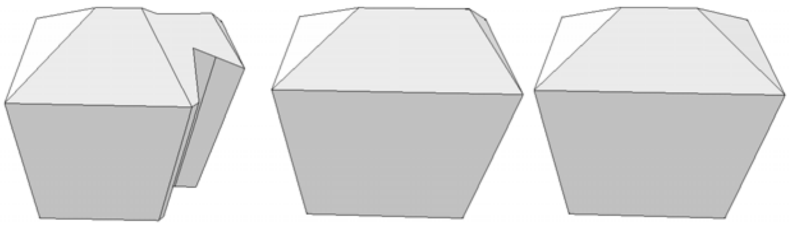


Fig. 9. Original 3D building model (left) and its generalizations via cell decomposition (middle) and primitive instancing (right).

Due to different ridge heights, some roof cells may not have a high enough percentage value that is necessary to classify it reliably as a building cell. This happens especially at the valley where two buildings meet (see Fig. 10). The missing cell disturbs the appearance of the generalised building model as such a roof shape is likely to be wrong.

In all three situations, an interpretation of the roof structure is required to create a simplified roof that best resembles the original model, is symmetric and has a realistic shape. Because the height discontinuities of the roof structure have already been incorporated into the cell decomposition, the interpretation can be done per cell.

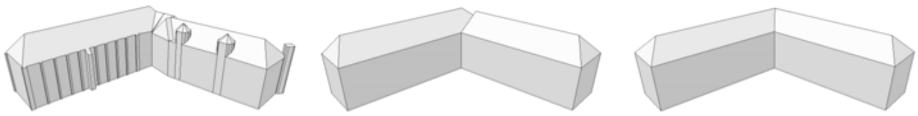


Fig. 10. Original 3D building model (left) and its generalizations via cell decomposition (middle) and primitive instancing (right).

The interpretation of the roof type is performed via a cell based primitive instancing approach. Here, every cell is tested against all possible primitive types that are parameterised in terms of the roof properties. So far, we support the eight roof types that are shown in Fig. 11. These are flat, shed, gabled, hipped roof and some connecting elements. The gabled and hipped roof elements need a ground plan in the shape of a parallelogram. Other shapes are not possible and must result in a flat or shed roof. However, the cell decomposition for most buildings with a gabled or hipped roof will usually provide an adequate set of cells. This prerequisite of the cells can be ensured during the generation of the ground plan decomposition by using only approximating planes parallel and rectangular to the general orientation of the building. Otherwise if the prerequisites can not be met, the primitives must be shaped as flat or shed roofs. Alternatively, the roof can be generalised via the cell decomposition approach.

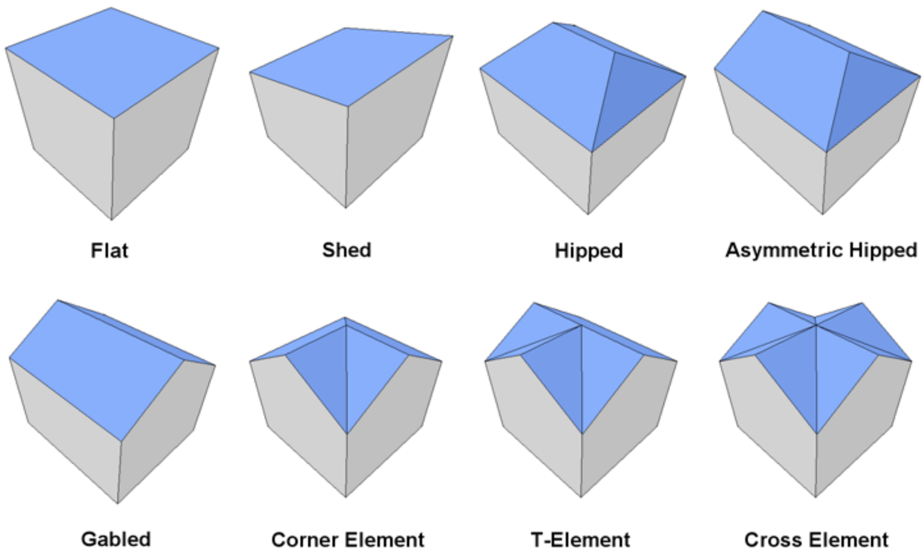


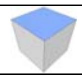
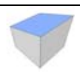
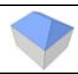
Fig. 11. The eight primitive types supported by the roof simplification.

The selection of the roof type and the parameter estimation works by discretising the bounding box of the cell's ground plan. For example, if the 2D space is divided in ten times ten subspaces, then 100 samples are the result. For each subspace inside the cell area, a length and an angle are computed. The length denotes the distance of the original model to the instanced primitive and the angle is computed as the horizontal difference of their normal directions. The vertical components of the normal vectors are ignored, which makes the angle independent from the eaves and ridge heights.

In order to determine the roof type, all eight primitives are instanced and the angle values for all samples computed. An angle below 30° is considered a match. For the hipped roof type, changes in the ridge length results in a different number of matches. Therefore, several parameter values are used. The primitive type with the highest number of matches is taken as an intermediate result for the cell's roof type. Afterwards the ridge and eaves heights are initially set to the highest and lowest value of the original roof polygons inside that cell. The geometric error is determined as the sum of all squared length values. Both parameters are then individually altered until the error is at a minimum.

Once the roof types and their parameters have been determined, the type of each cell is validated against a preference table (see Table 1). Therefore, the cells with approximately the same eaves and ridge heights are first grouped together. For the validity check of a cell, only the cells in the same group are of interest. For each cell, the number of neighbour cells and their arrangement are considered and compared with the preference table. The roof types of the cells of a group are altered until the best overall match is found and the roof parameters that are shared among a group of cells are estimated for the whole group again. This concerns mainly the eaves and right heights, so that smooth roof polygons are created for neighbouring cells.

Table 1. Preference table for primitive roof types.

neighbour primitives								
0	+	+	++	–	++	–	–	–
1	+	+	○	++	++	–	–	–
2[1]	+	+	○	–	++	–	○	–
2[2]	+	+	–	–	–	++	–	–
3	+	+	–	–	–	–	++	–
4	+	+	–	–	–	–	–	++
– bad match		○ possible match		+ good match		++ perfect match		

[1] opposite [2] corner arrangement

In some circumstances, the roof simplification by primitive instancing creates shapes that do not conform to reality. Rather a valid appearance is preferred. This would be a problem in building reconstruction where similar techniques are applied. Here the aim is to generate a true-to-life representation of the building. In generalisation, as long as multiple cells have a high probability that results from the type and parameter estimation and are good matches in the preference table, then the roof shape is very likely also a good overall simplification. If the overall deviation in the length, angle and preference table is too high, then a fallback to the cell decomposition approach is always possible.

The interpretation of the roof structure, however, has some major advantages compared to the cell decomposition approach. First, symmetries in the roof structure can easily be maintained by adjusting the slopes of gabled and hipped roof elements. Second, a special simplification for uniform roof elements is also possible. For example, we experimented with parallel gabled and hipped roofs that are quite common for factory or shopping halls. Once the parameters of a building are known, which includes the number of the uniform elements, a typification of the roof structure is possible. Typification is an elementary generalisation operator that replaces n features by a lower number of features. The seven uniform hipped roof elements in the building of Fig. 12 are e.g. replaced by five elements. In this example, the parameterised rim can also either be retained or removed by the generalisation operator.

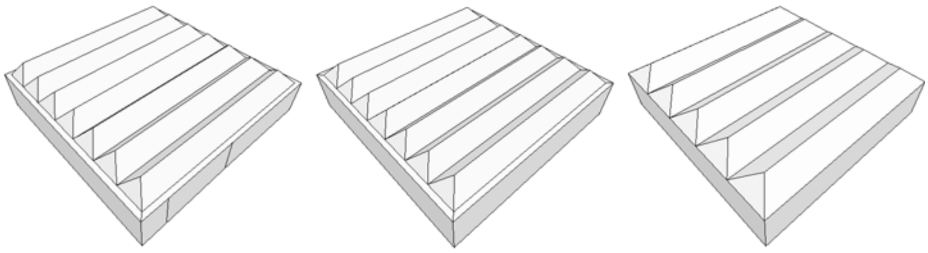


Fig. 12. 3D building model with uniform parallel hipped roof elements in its original shape (left) and before (middle) and after (right) typification.

Another example is the simplification of round and curved building elements. For the palace in Fig. 13, the three tower elements were first identified from the ground plan polygon by their circular arranged facade segments. After their parameters were determined, all tower polygons were removed and the simplification of the remaining building model was performed by the primitive instancing approach as described. Afterwards, the towers were added again to the final model as simplified versions. Without an interpretation of the towers, these elements would be eliminated by the simplification or might even interfere with the generalization process.

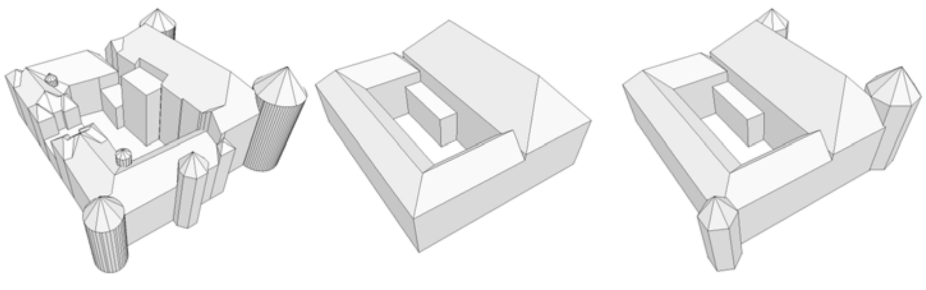


Fig. 13. 3D building model with circular tower elements in its original shape (left), after generalisation of the main building (middle) and with simplified towers (right).

7 Conclusion

Map and map-like presentations are essential to communicate spatial information. As 3D city models are becoming standard products of surveying offices, map-like 3D presentations are only a matter of time until they become available for a wide audience. Because maps need to be mobile, such applications will run on mobile devices with all their limitations. As 2D generalisation operators are already a common tool to prepare data to the scale of maps, such a scale-depending transformation of 3D data will require new operators.

This paper proposes a new algorithm for the simplification of solitary 3D building models. It is based on cell decomposition and primitive instancing. Geometric properties that are specific to buildings like the coplanarity, parallelism and rectangularity of facade segments are preserved during simplification or can even be enforced if needed. The generalisation is solely controlled by an intuitive distance threshold value that specifies the minimum size of the building elements that are created.

The partition of the algorithm into two stages proved to be very effective as the cell decomposition of the building's ground plan simplifies the generalisation of the roof structure. We have shown two approaches for roof simplification. We think that the interpretation of the roof shape is necessary in order to execute more elaborate simplification operations.

Acknowledgements

The research described in this paper is founded by "Deutsche Forschungsgemeinschaft" (DFG – German Research Foundation). It takes place within the Collaborative Research Centre No. 627 "NEXUS – SPATIAL WORLD MODELS FOR MOBILE CONTEXT AWARE APPLICATIONS" at the Universität Stuttgart. The 3D building models are provided by Stadtmessungsamt Stuttgart.

References

1. Gröger, G., Kolbe, T.H., Plümer, L.: City Geographic Markup Language. Approved Discussion Paper of the Open Geospatial Consortium. (2006)
2. Albert, J., Bachmann, M., Hellmeier, A.: Zielgruppen und Anwendungen für Digitale Stadtmodelle und Digitale Geländemodelle – Erhebung im Rahmen der Arbeitsgruppe „Anwendungen und Zielgruppen" der SIG3D im Rahmen der Initiative GDI-NRW (2003).
3. Buchholz, H., Döllner, J., Nienhaus, M., Kirsch, F.: Real-Time Non-Photorealistic Rendering of 3D City Models. In: Proceedings of the 1st International Workshop on Next Generation 3D City Models, Bonn (2005).
4. Staufenbiel, W.: Zur Automation der Generalisierung topographischer Karten mit besonderer Berücksichtigung großmaßstäbiger Gebäudedarstellungen, Wissenschaftliche Arbeiten der Fachrichtung Vermessungswesen der Universität Hannover (51), Ph.D. Thesis (1973).
5. Powitz, B.-M.: Zur Automation der Kartographischen Generalisierung topographischer Daten in Geo-Informationssystemen, Wissenschaftliche Arbeiten der Fachrichtung Vermessungswesen der Universität Hannover (185), Ph.D. Thesis (1973).
6. Regnauld, N., Edwardes, A., Barrault, M.: Strategies in Building Generalization: Modelling the Sequence, Constraining the Choice. In: Progress in Automated Map Generalization – ACI (1999).
7. Van Kreveld, M.: Smooth Generalization for Continuous Zooming. In: Proceedings of the ICA, 4th Workshop on Progress in Automated Map Generalization, Peking, China (2001).

8. Harrie, L.E.: The Constraint Method for Solving Spatial Conflicts in Cartographic Generalisation. In: Cartography and Geographic Information Systems (1999).
9. Weibel, R.: A Typology of Constraints to Line Simplification. In: Proceedings of the 7th International Conference on Spatial Data Handling (1996) 533-546.
10. Meyer, U.: Generalisierung der Siedlungsdarstellung in digitalen Situationsmodellen. Wissenschaftliche Arbeiten der Fachrichtung Vermessungswesen der Universität Hannover (159), Ph.D. Thesis (1989).
11. Rainsford, D., Mackaness, W.A.: Template Matching in Support of Generalisation of Rural Buildings: In: D. Richardson und P.v. Oosterom (eds.): Advances in Spatial Data Handling, 10th International Symposium on Spatial Data Handling. Springer-Verlag, Berlin (2002) 137-152.
12. Camara, U., Antonio, M., Lopez, A., Javier, F.: Generalization Process for Urban City-Block Maps. In: Proceedings of the XXII International Cartographic Conference, La Coruna, Spain (2005).
13. Li, Z.: Transformation of Spatial Representations in Scale Dimension. In: International Archives of Photogrammetry and Remote Sensing. Vol. 31, Part B3/III (1996) 453-458.
14. Sester, M.: Generalization based on Least Squares Adjustement. In: International Archives of Photogrammetry and Remote Sensing, Vol. 33, Part B4/3, Amsterdam (2000) 931-938.
15. Mayer, H.: Scale-Space Events for the Generalization of 3D-Building Data. In: International Archives of Photogrammetry and Remote Sensing, Vol. 32, Part 3/1 (1998) 520-536.
16. Forberg, A.: Generalization of 3D Building Data based on a Scale-Space Approach. In: Proceedings of the XXth Congress of the IRPRS, Vol. 35, Part B, Istanbul, Turkey (2004).
17. Thiemann. F, Sester, M.: Segmentation of Buildings for 3D-Generalisation. In: Working Paper of the ICA Workshop on Generalisation and Multiple Representation, Leicester, UK (2004).
18. Anders, K.-H.: Level of Detail Generation of 3D Building Groups by Aggregation and Typification. In: Proceedings of the XXII International Cartographic Conference, La Coruna, Spain (2005).
19. Thiemann. F, Sester, M.: 3D-Symbolization using Adaptive Templates. In: Proceedings of the GICON, Wien, Austria (2006).
20. Luebke, D., Reddy, M., Cohen, J.D.: Level of Detail for 3D Graphics, Morgan Kaufmann, USA (2002).
21. Coors, V.: Feature-Preserving Simplification in Web-Based 3D-GIS. In: Proceedings of the 1st International Symposium on Smart Graphics. Hawthorne, NY, USA (2001) 22-28.
22. Rau, J.Y., Chen, L.C., Tsai, F., Hsiao, K.H., Hsu, W.C.: Automatic Generation of Pseudo Continuous LoDs for 3D Polyhedral Building Model. In: Innovations in 3D Geo Information Systems, Springer Verlag, Berlin (2006).
23. Kada, M.: Automatic Generalisation of 3D Building Models. In: Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications, Ottawa, Canada (2002).
24. Foley, J. van Dam, A., Feiner, S., Hughes, J.: Computer Graphics: Principles and Practice (2nd Edition), Addison-Wesley (1990).