

BUHOM: A PROGRAM FOR ENHANCEMENT OF GEOMETRIC TOPOLOGIC CONSISTENCY OF BUILDING OBJECTS

I. O. Bildirici^{a,*}, D. Heidorn^b

^a Selcuk University, Faculty of Engineering & Architecture, Department of Geodesy & Photogrammetry, 42075 Kampus, Konya , Turkey – bildirici@selcuk.edu.tr

^b Institut für Kartographie und Geoinformatik, Universität Hannover, Appelstr. 9A, 30167 Hannover, Germany – Dieter.Heidorn@ikg.uni-hannover.de

Commission IV, WG IV/3

KEY WORDS: generalization, building generalization, digitizing errors, vector data

ABSTRACT:

Vector data is captured using various methods, e.g. manual digitizing, automatic vectorization etc. Errors or inconsistencies, e.g. self-cutting polygons, identical points etc, can not be avoided, even if digitizing is done very carefully. Vector data must be analyzed and corrected prior to any GIS application. This process is called "line-cleaning". In this paper we introduce a computer program that is capable of analyzing of building objects. It can correct digitizing errors and improve geometric quality of data, doing right angle adjustment, making buildings parallel to streets etc. The program was designed to process building objects before generalization with CHANGE generalization software package. However, it can also be used independently.

1. INTRODUCTION

Spatial data in vector format is captured with various methods, e.g. manual digitizing, automatic vectorization etc. Errors or inconsistencies, e.g. self-cutting polygons, identical points etc, can not be avoided, even if digitizing is done very carefully. Building objects are generally captured as polygons (area objects) in vector format. They cannot be considered as conventional area objects. The structure of their outer contours (object boundary) is quite different from land use polygons for example. Buildings are man-made objects, which have mostly right angles at corners, and are generally located parallel to streets.

Many GIS systems have tools to examine and correct the vector data, which are called line-cleaning tools. These tools are developed to help creating a correct network and/or polygon topology, but not suitable for building objects.

The software tool BУHOM has been designed to process 1:1000 scaled building data. It is developed with Fortran 90 programming language. The functionality of BУHOM can be summarized as follows: The objects are examined if there are self-cuts on contours. Identical points are sought, if exist they are eliminated. The angles on the corners are examined within certain pre-specified limitations; the sides are adjusted to have a right angle on that corner. The system makes sure, that the direction of the building (bearing angle of the longest side) stays unchanged during this process. If street data (center-lines or right and left sides of streets) is available, and if the building, within a threshold value, is nearly parallel to the nearest street, it is rotated so it becomes exactly parallel. The center of mass and the area of the building stay unchanged after processing.

BУHOM's functionality is actually considered as a pre-processing for the CHANGE software package, a tool for automatic building generalization. It can also be used independent from CHANGE. BУHOM has been tested on several data sets since 1998 at the IKG (Institute for Cartography and Geoinformatics, Hannover). It is mainly used before the generalization with CHANGE. Several tests and analysis on the results showed that the program delivers good results.

In this paper, the methods employed in the development of the program, and the application possibilities will be discussed in detail.

2. GEOMETRIC IMPROVEMENTS ON BUILDING OBJECTS

Our strategy for correction of digitizing errors and for improvement of geometry consists of following steps:

- Small area test. If the object area is smaller than a threshold value, it is deleted.
- Elimination of identical points within the object boundary.
- Test on self-cutting objects. If self-cuts exist, the object is deleted.
- Alignment or elimination of points that are approximately on the same straight line with prior and next points (insignificant points, see figure 4).
- Right angle adjustment. The building edges are sequentially examined. If the angle between adjacent edges approximates to a right angle within a threshold value, the right angle is reconstructed by rotating one of the edges.

* Corresponding author.

- Test on possible parallelism to the next street. If it is possible within a threshold value, parallelism is reconstructed by rotating the building round the center of mass.
- If the object is significantly changed after the tests above, original area, center of mass and direction (bearing of the longest edge) are reconstructed by rotating, scaling and moving.

The threshold values, which are entered by user, for the tests above play an important role for the performance of the program. These certainly depend on the scale. Program default values are determined at the 1:1000 scale.

Three properties of buildings should stay unchanged after processing.

- Area
- Center of mass
- Direction of the building (bearing of the longest edge, figure 1)

These parameters are saved before processing.

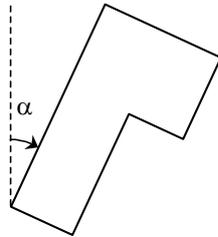


Figure 1. Direction of the building

In figure 2, the flow chart of the program is shown, where the processes and strategy of BUHOM are depicted.

2.1 Right Angle Adjustment

The right angle adjustment process is relatively simple. Building objects are processed independently from each other. Starting from the first edge of the object, angles between edges are calculated and determined if these may be right angles. In figure 3, for instance, if the angle α approximates to a right angle within a threshold value, the edge 2-3 is rotated around the middle point of the edge. So the edges 1-2 and 2-3 are perpendicular to each other. The examination of the edges continues until the edge 8-1.

Another approach would be the rotation of the edge around the starting point (point 2). It is not preferred in the program development because we assumed that such rotations cause more change in the location of building.

If two edges perpendicular to each other, following condition must be fulfilled.

$$m_{12} = \frac{(y_2 - y_1)}{(x_2 - x_1)} \Rightarrow m_{23} = \frac{(y_3 - y_2)}{(x_3 - x_2)} \Rightarrow m_{12} + \frac{1}{m_{23}} = d = 0 \quad (1)$$

If $d \neq 0$, the deviation from the right angle is following.

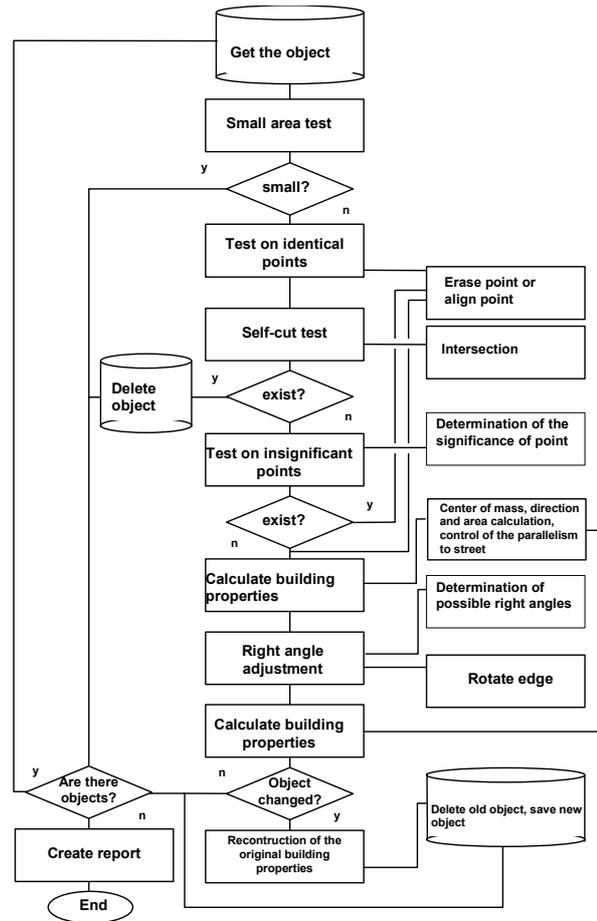


Figure 2. Flow chart of the program BUHOM

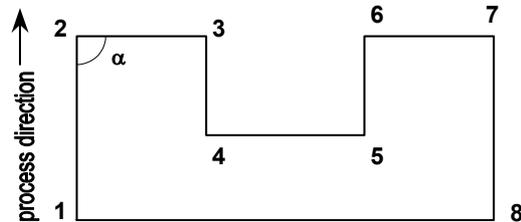


Figure 3. Processing of edges at right angle adjustment

$$d\hat{\alpha} = \tan d\alpha = \frac{1 + m_{12}m_{23}}{m_{12} - m_{23}} \quad (2)$$

If $d\hat{\alpha}$ is smaller than the threshold value (for angles), it is allowed to rotate the edge 2-3. The threshold value selected by the user determines the degree of the shape changing of the whole process.

2.2 Insignificant Points on the Object Boundary

On the boundary (contour) of building objects there are points that lie approximately on the same straight line with previous and next points. In figure 4, for instance, point 2 is such a point. Such points can be identified examining the angle $d\hat{\alpha}$ (figure 4). If it approximates to zero, point 2 can be considered as

insignificant. A threshold value is needed to decide. In other words, if $d\alpha$ is smaller than a threshold value determined by user, the examined point (point 2 in figure 4) is regarded as insignificant. It can be aligned or eliminated according to the choice of the user.

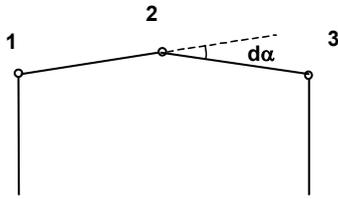


Figure 4. Insignificant points

2.3 Building-Street Relationships

Buildings are often located parallel to the next street. If street data is available, the existence of such relationships can be examined. If possible, parallelism can exactly be reconstructed. For this purpose street boundaries or centerlines can be used.

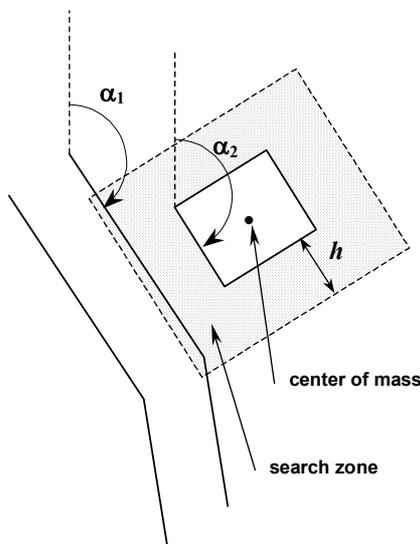


Figure 5. Building-street relationship

The nearest street boundary (or centerline) is searched by using a search zone (figure 5), which is defined by user (h parameter). After finding the nearest straight line segment of the street boundary, the nearest edge, which would be parallel to street, is sought. If such an edge exists, the difference of the bearings (α_1 and α_2) is examined. If it is smaller than a threshold value, the objects are considered parallel and the building is rotated around the center of mass.

If a building is rotated because of parallelism to street, the direction of the building, at the end of processing, is not reconstructed, but area and center of mass.

2.4 Topologic Inconsistencies Among Neighbor Buildings

After right angle adjustment topologic inconsistencies can occur, because the objects are rotated, scaled and moved. These inconsistencies can be overlaps and gaps between neighbor buildings (figure 6).

In order to analyze and correct the topologic inconsistencies, a hierarchical approach can be employed.

- All objects are sorted descending according to area.
- Beginning with the smallest one, small objects are compared with larger ones. Box test technique can speed up the comparison at this step.
- If any gap or overlap is found, small object is changed. If there is a gap, small object is extended towards the large object (object 1 in figure 6 - top). If any overlap is the case, small object is trimmed (object 2 in figure 6 - bottom). In both cases small object can be moved, but this can cause another inconsistencies with other objects around.

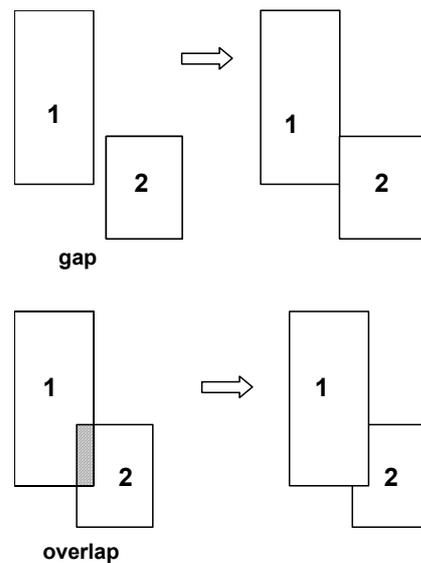


Figure 6. Topologic inconsistencies

After these corrections some objects can have small edges, which is not desired in the sense of generalization. However, the enhancements here should be considered as a preprocessing prior to generalization. In other words, these small edges are eliminated in the building generalization, which is supposed to be undertaken thereafter. Another point is the changes in size and orientation of the objects. Since the gaps and overlaps are supposed to be small in size, the changes in size can be considered to be small amounts. The width of gaps is entered by user, which should be determined within minimal dimensions on the map (e.g. 0.2 mm on the map). The overlaps normally occur due to digitizing errors. They must be removed even if the overlapping area is not a small amount. In this case object area may considerably be changed.

3. SOFTWARE DEVELOPMENT

The program BUHOM is developed with the programming language FORTRAN 90. It uses the DBB database in the background. The input files (building file and street file) and output file (processed buildings) are in the FDP file format. DBB format and FDP format are in-house file formats of Institute for Cartography and Geoinformatics of University of Hannover. For more information about file structures see IfK (1998), Bildirici (2000).

BUHOM uses a configuration file for user parameters. In this file (buhom.cfg) the following parameters are entered by user:

- Search radius for identical points
- Angular threshold value for right angle adjustment, for recognition of insignificant points, and for parallelism to street.
- Threshold value for small area
- Maximum distance from building to street (h in figure 5)
- Are small objects deleted or not? (y/n)
- Are identical points eliminated? (y/n)
- Self-cutting test? (y/n)
- Test on insignificant points (y/n)
- Insignificant points are to be eliminated or aligned? (y/n)
- Right angle adjustment (y/n)

After processing a report file is created, in which some useful information about last program session is to find.

BUHOM do not correct gaps and overlaps that can occur after right angle adjustment, because such inconsistencies are removed in the following generalization process with CHANGE. If generalization does not follow the enhancement, these inconsistencies can be corrected with another program (GQE), which is developed by Bildirici (2000). GQE (Geometric Quality Enhancer) is also developed with FORTRAN 90 programming language with the aim of correcting gaps and overlaps between buildings (figure 6). In this program the approach for topologic inconsistencies discussed in 2.4 is realized. In the future the authors are planning to integrate the functionality of GQE into BUHOM.

In order to give an idea about the functionality of BUHOM, an application is given in figures 7 and 8, in which original objects and processed objects can be seen. In this application all tests were performed, but no parallelism to street. In figure 8, a building with a complex outline is given. This shows the functionality of the program visually.

4. CONCLUSIONS

Vector data is prone to geometrical inconsistencies, which should be examined and corrected before GIS applications and generalization. This is called line cleaning, which is integrated in most GIS software. Line cleaning tools, in general, are developed to process lines objects rather than polygons. Since generalization requires error free data, especially for buildings, a preprocessing step is needed. Therefore the program BUHOM was developed. In addition to common line cleaning tasks, it is capable of doing right angle adjustment and reconstructing parallelism to streets. It has been tested at Istanbul Technical University, Istanbul, and at the Institute for Cartography and Geoinformatics, Hannover (Bildirici, 2000; Bildirici, 2003).

BUHOM can be used both as a preprocessing tool before generalization and as an independent tool for geometric/topologic improvement.

ACKNOWLEDGEMENT

The authors thank Prof.Dr. Monika Sester for her encouragement and valuable suggestions.

REFERENCES

Bildirici, I.O., 2000. *Generalization of buildings and roads in scale range 1:1000-1:25000*. Dissertation, Graduate School of Istanbul Technical University, Istanbul, Turkey. (in Turkish).

Bildirici, I.O., 2004. Building and road generalization with the CHANGE generalization software on Turkish topographic base map data. *Cartography and Geographic Information Science*, 31(1), pp.43-51.

Kruse, I., Powitz, B.M., 1990. Die netzwerkartige Datenbank des IfK, Institut für Kartographie, Hannover (unpublished document).

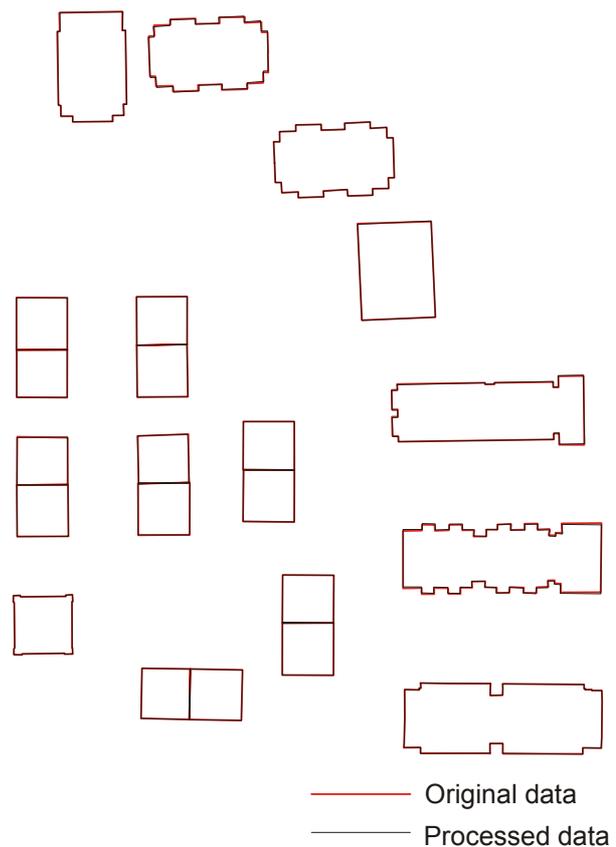


Figure 7. Original and processed data (Original scale 1:1000, reduced to 1:2 000)

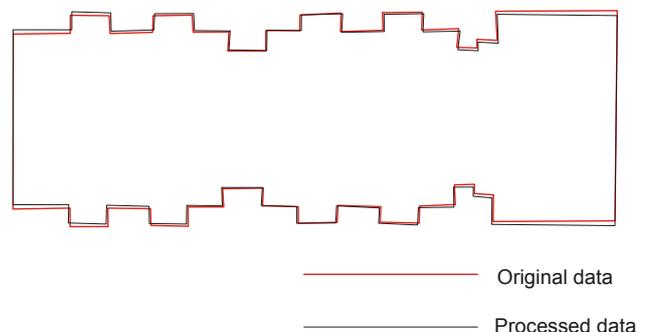


Figure 8. One of the processed buildings (enlarged)