

Algorithm for constructing three-dimensional Barcodes to represent nD spatial objects in GIS

D E Andrianov¹, S V Eremeev¹ and Y A Kovalev¹

¹Vladimir State University named after Alexander Grigorievich and Nikolai Grigorievich Stoletovs, Gorky street, 87, Vladimir, Russia, 600000

e-mail: yurko02@mail.ru

Abstract. The article describes the algorithm for creating three-dimensional Barcodes to represent nD features. The algorithm is based on computer topology methods using the 3D sweep hull algorithm for computing convex hulls and Delaunay triangulation. The result of the algorithm are 3D Barcodes of features. 3D Barcode graphs were built that reflect their time differences.. The algorithm for constructing 3D Barcodes will allow analyzing spatial nD objects at different time intervals.

1. Introduction

Currently, geographic information systems mainly work with vector 2D maps. But now, the information on 2D maps is not enough for a more detailed analysis of the terrain, so GIS is increasingly beginning to include the ability to process 3D maps.

The relevance of the work lies in the fact that the existing algorithms for processing and storing 3D map data work exclusively with coordinates, which significantly increases the processing time of such objects. It is proposed to apply computer topology methods using the 3D sweep hull algorithm to develop an algorithm for constructing 3D Barcodes that will allow storing and quickly processing data on spatial objects in GIS [1, 2, 3].

There are various algorithms for handling nD features. In [4], an algorithm is presented that allows processing nD objects. The meaning of the algorithm is to simulate n-dimensional characteristics (time and scale) as additional geometric dimensions perpendicular to the spatial objects, creating a higher order model in the use of intervals, on the basis of which 2D models are raised to the following dimensions. These intervals are obtained from the cell complex (topological space) and they are divided into smaller ones. Extrusion is a widely used method in GIS for creating simple 3D models. Starting from the planar partition of the polygons and the height interval associated with each of them, it generates a set of spatially-decomposing rectangular polyhedra, assuming that each polygon exists over its entire segment. For example, a set of building trails and related heights are squeezed into a set of simple prismatic buildings [5, 6, 7].

The authors of [8] were builded the foundation for the integration of five dimensions into one formal presentation of data. The formal definition of geographic data in the 5D conceptual continuum

will most effectively manage and query geographic data using one integrated approach. In addition, it will ensure consistency in scope and time.

This approach led to a new theory and method for geodata, as well as technologies that implement multidimensional partitioning. Integration of multidimensional geodata concepts allows to use of common geometry and embedded topological, temporal and scale structures through full 3D + time + scale splitting.

But these algorithms are based on the use of coordinates of objects, which significantly slows down their processing.

In this work, three-dimensional Barcodes will be built on the basis of data from spatial objects from a 3D map. The resulting Barcodes will be compared with 2D Barcodes for vector maps. The advantages of the developed algorithm can be considered as the speed of work, the ability to work with three-dimensional maps, as well as to store object changes over time.

2. Algorithm for constructing 3D Barcodes for nD features in GIS

The algorithm of building 3D Barcodes is based on the 3D sweep hull algorithm [9]. This algorithm is also called the Newton apple shell, and it functions as follows:

1. Sort a set of points $\{x, y, z\}$ in the sequence $z \rightarrow x \rightarrow y$.
2. Starting with the first of the sorted points, the points are connected, until a triangle of the area is formed to form an array. The process of connection occurs by building a ball around each point.
3. New points are sequentially added to the array. The edges of the array are triangles, which are represented as a list with information about adjacency. The process of adding a new point to the array includes determining which triangular faces are visible to the new point and replacing them with new triangles made using the new point and closing the edges in terms of the new point.
4. Next, a non-intersecting triangulation of the set of triangles is created.
5. Adjacent pairs of triangles of this triangulation must be "inverted" in order to create Delaunay triangulation from the original non-overlapping triangulation.

The algorithm generates Delaunay triangulation together with a three-dimensional convex hull for a set of points.

The construction of the Barcode itself takes place by analogy with 2D Barcodes as described in [10]. The object barcode $B = \{(x_i, l_i)\}$, where, $i = 1, 2, \dots, n$ is the hole number, n is the number of holes, x_i is the coordinate of the beginning of the object's Barcode hole, l_i is the length of the object's Barcode hole.

A distinctive feature is the addition of the third component in the Barcode which is a time t . Taking into account the added time, Barcodes for different periods of time are combined on each radius into a single component.

Accordingly, 3D Barcodes, built on the basis of data on nD objects, will look like this: $B = \{(x_i, l_i, t_i)\}$, where t_i is a certain period of time in which the object was changed [11, 12].

Also 3D Barcode can have another third component line, for example, scale s .

Then the Barcode will look like: $B = \{(x_i, l_i, s_i)\}$, where s_i is the scale on which the feature is displayed.

3D Barcode can be built not only for one object, but also for their group. In this case, the coordinates of the entire group of objects will be perceived as a single object.

Consider the construction of 3D Barcode on an example. Figure 1 shows a 3D feature. Figure 2 shows the 3D Barcode for the object shown in Figure 1.

The graph of Figure 2 shows the Barcode length on the axis x , the Barcode radius on the axis y , and the year of change on the axis z .

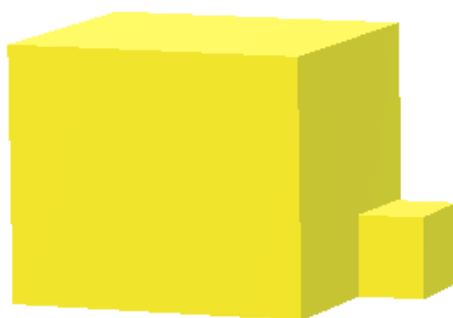


Figure 1. Example of a 3D object.

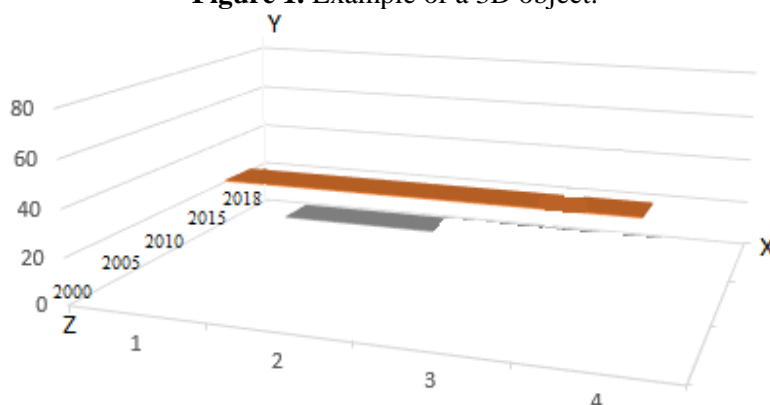


Figure 2. 3D Barcode.

3. The results of the algorithm

To carry out the experiment of building 3D Barcode, a tablet of the 1990s was taken, reflected in Figure 3 (a) with the image of a map of the city's terrain. Some of the objects were built in 3D, and Barcode was built for them. Then the plot from this area was taken from Yandex maps for 2018. This area is reflected in Figure 3 (b). For these objects was also built 3D Barcode shown in Figure 3 (c).

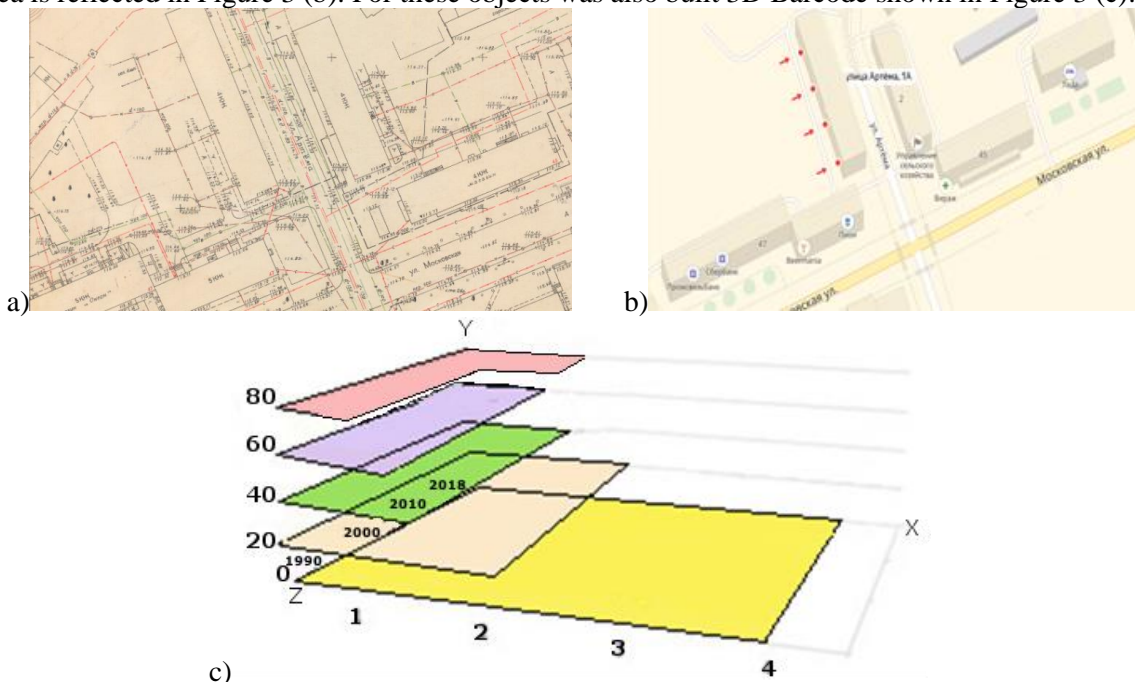


Figure 3. A group of objects and their 3D Barcodes: a-b) Objects that change over time; c) 3D Barcodes of objects in different time intervals.

For comparison, 2D Barcodes were constructed for the objects from Figure 3 (a, b). These Barcodes are shown in Figure 4 (a, b).

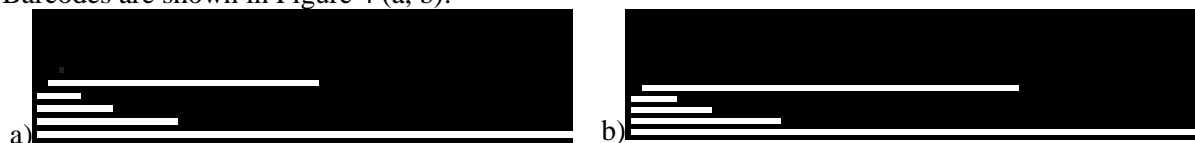


Figure 4. Graph of 2D Barcodes: a) Barcode for objects from figure 3 (a); b) Barcode for objects from figure 3 (b).

When comparing 2D and 3D Barcode graphs, we can conclude that 3D Barcodes are more informative, store more information, and, unlike 2D Barcodes, allow you to quickly access the desired nD object on the map. One of the advantages of the algorithm's performance is the use of Delaunay triangulation for constructing holes of objects, which significantly reduces the processing time of objects.

4. Conclusion

An algorithm for constructing 3D Barcode for nD features developed in the article. It can be applicable to all vector maps, including 3D maps.

Unlike 2D Barcode, this algorithm takes into account an additional characteristic which is a time. This allows to track changes in nD features over the years.

In the future, the algorithm will also be able to use the scale as an additional characteristic, which will significantly reduce labor costs when working with vector maps.

The developed algorithm is also useful in real estate. It will allow to search for the best terrain for building buildings, estimate the time of building objects, as well as analyze areas for missing buildings.

5. References

- [1] Ereemeev S V, Andrianov D E and Komkov V A 2013 Algorithms for the formation of a graph model of an urban area in a GIS *Geoinformatics* **4** 19-24
- [2] Zhilin Li and Qi Zhou 2012 Integration of linear and areal hierarchies for continuous multi-scale representation of road networks *Intern. J. of Geographical Information Science* **26** 855-880
- [3] Herbei and Radulov I 2015 Topology of spatial data *In 15th International Multidisciplinary Scientific GeoConference SGEM* **2(2)** 87-94
- [4] Arroyo Ohori K, Ledoux H and Stoter J 2015 A dimension independent extrusion algorithm using generalised maps *International Journal of Geographical Information Science* **29(7)** 1166-1186
- [5] Myasnikov E V 2017 Hyperspectral image segmentation using dimensionality reduction and classical segmentation approaches *Computer Optics* **41(4)** 564-572 DOI: 10.18287/2412-6179-2017-41-4-564-572
- [6] Afanasyev A A 2017 Hybrid methods of automated identification of changes in landscape cover according to the data of remote sensing of the Earth under noise conditions *Computer Optics* **41(3)** 431-440 DOI: 10.18287/2412-6179-2017-41-3-431-440
- [7] Pechenkin V V 2017 Optimization of placement of observation tools in a three-dimensional scene in order to minimize "blind zones" *Computer Optics* **41(2)** 245-253 DOI: 10.18287/2412-6179-2017-41-2-245-253
- [8] Peter van Oosterom and Jantien Stoter 2014 5D Data Modelling: Full Integration of 2D/3D Space, Time and Scale Dimensions *Technical University of Delft* **1** 2-16
- [9] Dr David A Sinclair 2016 A 3D Sweep Hull Algorithm for computing Convex Hulls and Delaunay Triangulation **1** p 26

- [10] Kovalev Y A, Ereemeev S V and Andrianov D E 2018 Algorithm for searching for differences in spatial objects that change over time, based on Barcode *International Conference on Soft Computing and Measurements* **1** 481-483
- [11] Boissonnat J-D, Srikanta K C and Tavenas S 2015 Building Efficient and Compact Data Structures for Simplicial Complexes *An extended abstract appeared in the proceedings of SoCG* **1** p 230
- [12] Edelsbrunner H and Mücke E P 1994 Three-dimensional alpha shapes *ACM Trans Comput Graphics* **13** 43-72

Acknowledgments

The reported study was funded by RFBR and Vladimir region according to the research project № 17-47-330387.