Aalborg Universitet

Object oriented roads in modelmaps

Kjems, Erik; Kolar, Jan

Published in:
CUPUM 2007

Publication date:
2008

Document Version
Publisher's PDF, also known as Version of record

Link to publication from Aalborg University

Citation for published version (APA):
Kjems, E., & Kolar, J. (2008). Object oriented roads in modelmaps. In A. Nélson Rodriques da Silva, & L. C. Lucas de Souza (Eds.), CUPUM 2007: Book of abstracts University of São Paulo.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

? Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
? You may not further distribute the material or use it for any profit-making activity or commercial gain
? You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from vbn.aau.dk on: May 01, 2017

OBJECT ORIENTED ROADS IN MODELMAPS

Erik KJEMS
Associated Professor
Centre for 3D GeoInformation
Aalborg University
Niels Jernes Vej 14
DK-9220 Aalborg
Denmark
Tel: +45 96358079
Fax: +45 98156541
http://www.vrmedialab.dk/~kjems
kjems@3dgi.dk

Jan KOLAR
Assistant Professor
Centre for 3D GeoInformation
Aalborg University
Niels Jernes Vej 14
DK-9220 Aalborg
Denmark
Tel: +45 96359799
Fax: +45 98156541
http://www.3dgi.dk
kolda@3dgi.dk

Abstract: Modelmaps consist of features which can be observed or perceived in our daily life (Kjems and Kolar 2005). One of the major features, which apparently still are difficult to represent in an adequate way, are roads. Despite their visual significance and spatial extent, it is difficult to represent roads as objects in databases and to identify them as such in modelmaps. In most 3D systems roads are represented through textures like ortho-photos and not as feature-based geometry. This paper suggests a method to represent roads in 3D using an object oriented approach. 3D here means that road objects literally are cut into the terrain and represented as B-rep surface geometry. The data used for this approach are taken from the same data sets as used by car navigation systems. The methods described can be used in general, but parts of the approach are related to the GRIFINOR platform (http://www.grifinor.net) and developed particularly for that (Bodum, Kjems et al. 2005) (Kolar 2006).

Keywords: Roads, Terrain, Modelmap, Geo-information, Object-oriented

1. INTRODUCTION AND MOTIVATION

Modelmaps like we see them in Google Earth, Microsoft's Virtual Earth or NASA World Wind have a strong emphasis on the visual output and friendly user interface,. although the latter could be argued at least in terms of navigation mode. The visual representation is smooth and the user is given the ability to zoom in and out on particular geographical places around the world. More and more places are even represented with 3D models like the one in Figure 1 from the University of Aalborg, Denmark. More and more cities get a 3D model of the main part of their city. The motivation is given due to planning purposes, sales objectives, tourism or simply moving with the times. In Figure 1 an example of GRIFINOR is showing a change of perspective from world view to a detailed building view of the interior. Very often only landmarks and important buildings are modelled and presented and in many cases that is sufficiently.

Figure 1 Screen shot from GRIFINOR. From World view to the interior of the research facility at Aalborg University

But another general trend is clearly evolving; a trend towards improved model quality. This trend was expected and is also obvious. There is a lot of competition within the 3D modelling field. The result is visually appealing models with more details and spectacular light conditions. Once one has seen what is possible, the requirements and expectations for the next visualisation, for instance in connection with an urban planning project, have increased already. There are different kinds of projects categorised by the extent of the project. The smaller the project is the easier one can focus on a single building or house and work with details here while the surroundings are less important and can be handled with simple representations, such as bill boards. When the extent of the project is more like several building blocks the impact in the neighbourhood plays a bigger role, and the visualisation should handle these secondary parts with care. For example, a single storage of the building block can be modelled with more details to show the interior in order to demonstrate a future living situation. In modelmaps one will find all kinds of qualities in the modelling part depending on the purpose of the project, ability of the modeller and the performance of the system. The latter plays a big role in real time systems like those mentioned earlier and determines the amount of details that can be allowed to show in one scene. Visualised details depend primarily on the performance of the rendering system. Whether a real-time rendering or pre-rendering is convenient is then determined by the purpose of the visualisation. For instance, there is a huge difference on the visual output quality between sales material for a brochure and an on-line presentation. If one wants to carry out a visualisation for both purposes with the same modelmap a level-of-detail (LOD) solution must be deployed. Then maximum details with a long render time may be used for the brochure rendering and diminished detail controlled by the LOD algorithm may be used for the real time presentation.

More details in the scene also give the possibility of increasing the amount of features in the scene and this way, in case of an object-oriented solution, also to attach more information to the modelmap. The object-oriented design of GRIFINOR has a very strong emphasis on information and objects are supposed to carry any kind of information which is relevant to the feature they represent. Since features in GRIFINOR are coded as Java objects arbitrary data, information or action can be contained in a feature. In Figure 2 the relationship between information density with regard to available objects and the rendering time is presented. The shown correlation is not given at all times and is very dependent on the used rendering technology and the efficiency of the used LOD algorithm. The figure shows more likely that there is a correlation and that one should be aware of it when requiring products like visualisations and modelmaps. This figure tries to place road details in relation to the render time and the amount of information you want to be able to store and access in a modelmap concerning roads. The development clearly goes towards real time rendering with all information available. And it looks very promising. Cutting the road into the terrain and dividing it into sub-parts results in a higher number of objects, and more precise handling of information about the road. Please adopt this figure as an indicative figure and not as a scientific measure.

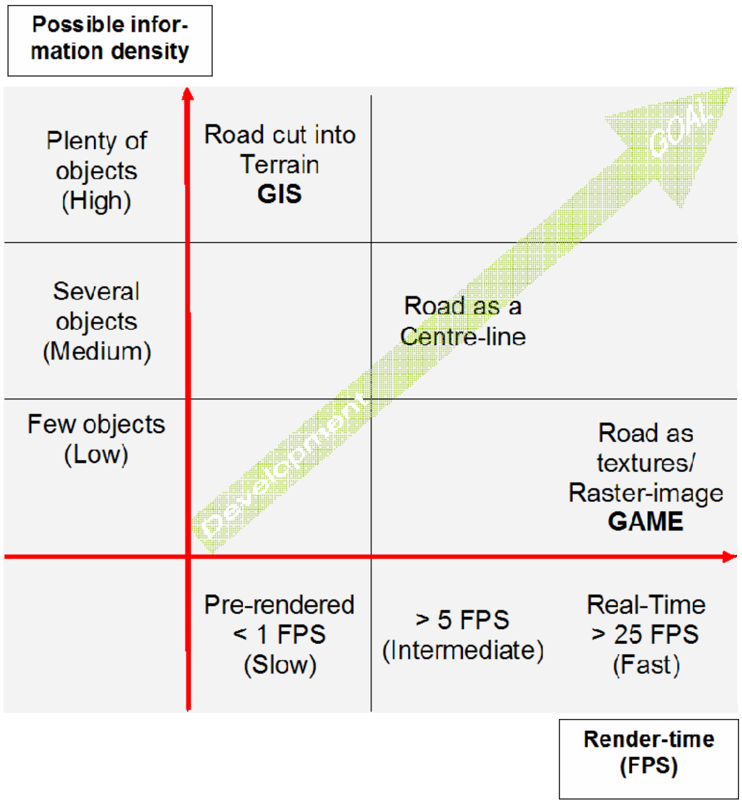

Figure 2 Relation between rendering time and possible information density

Since details play a bigger and bigger role in urban modelling the next pressing question must be what details one should focus on; where to spend the sparse resources. With reference to Figure 2 it very much depends on the application. Focus on representation of roads is an important part for an urban and rural area modelling for at least three reasons: being unable to navigate close to the ground in for instance Google Earth; being tired of low-res aerial textures; and finally a professional interest in precise representation of roads in modelmaps. The approach for a representation of roads presented in this paper is developed for the GRIFINOR platform. The GRIFINOR platform has been described on several occasions; please visit http://www.grifinor.net for further information.

Since roads are missing when one moves closer (zooms in) to the ground especially in urban areas, and since roads can be determined with acceptable results from available data, they have been chosen right after buildings for a closer investigation. The following presentation on how to model roads has not been implemented yet but will be as soon as some time is available.

2. PREREQUISITES OF A ROAD ALIGNMENT

There are many ways to build or design a road. The design will depend on a huge number of parameters, which vary with for instance the country we are in, the present road and safety level, the details necessary, and the laws that apply. In reality all these things are more or less given and simple to follow, but when one wants to model a road virtually there are even more circumstances one must take into account; both from a data availability point of view but also from a data model point of view. This chapter tries to bring some light into both areas.

2.1 Data models suitable for road

Normally one will distinguish between two different kinds of data representations suitable for roads. These are either data stored and handled as raster images or data stored and handled as points containing x,y,z information about the real world road structure either existing or planned. Choosing either raster or points or both depends very much on the purpose. It is possible to view this issue from many perspectives. Let’s numerate a few.

1. One can choose to see this issue as a computer science challenge, how to store, present and interact with a road model, not focusing on roads as such but merely on geometry which happens to be a road.
2. One can choose to look at the data available and try to make the best road model with the highest possible quality of details.
3. One can choose to present all roads as images entirely and simplifying the whole issue a lot though also limiting the possibilities within a presentation, cf. the introduction and Google Earth.
4. One can choose to take an origin within a limited usage and focus entirely on for instance screen size and view angles. This means that one lets the context and user interaction be the depending parameters for the development. Cf. car navigation as an example.
5. One can look at an application and one’s customers’ needs and try to gather the necessary data and develop the application especially for that purpose. Cf.

address and map search on the Internet.

Today it seems to be almost impossible to satisfy all the perspectives at the same time. The same issue has been dealt with within the area of cartography and is therefore not new (Kraak and Brown 2000). The main issue is perhaps to face it and accept that the challenges and constraints we have in cartography are inherited into the computer age. And even if we have the possibility of describing complex data sets, taking most situations into account, and being able to present dynamic maps changing their layout whenever required, one still has to take into consideration what the viewer might want to see, needs to see or is able to see. The latter is contingent upon the chosen technology like the screen size, the context, the vision ability of the viewer, light conditions, colours and so on.

These conditions all together differ from situation to situation and are not subject of this paper. Though, it is important to understand that a tremendous amount of different parameters will influence the final outcome and are crucial trying to optimize the visualization for a certain purpose.

2.2 Data available for the modelling

Cutting the road into the terrain means dealing with the road as surface geometry and not only as for instance a centre line. One will need a complete description of the road design to comply the demands for an acceptable level of model quality. Nevertheless, at this state of data availability and data understanding in general within the road sector, it is unlikely that a road for a modelmap will benefit from these data, which obviously are available due to modern road design using CAD systems, but not stored in a sufficient way in database Since we decided to implement roads as full geometry we have to do a lot of assumptions and take restraints into account. This also means that this attempt first of all is an experiment and not an implementation of a proven concept for roads. Not yet at least.

3. ROAD GEOMETRY

A newly initiated international cooperation within the Industry Alliance for Interoperability (IAI) society focuses on roads. This means that roads are going to be embodied into the International Foundation Classes (IFC) standard. A standard developed within the construction sector and widely adopted as international exchange standard for object oriented design. The work of describing new objects within roads started in the autumn of 2006 and does still not exist. It would not really have changed that much for the ongoing work in GRIFINOR because the attempt presented in this paper uses available data from the car navigation area and not CAD data. Therefore it is not possible to obtain sufficient data to describe the road surface in any case. Instead it is necessary to make assumptions upon the road and derive some templates from common road design. Unfortunately, roads are designed as differently as there are countries in the world, and even worse. Each county or municipality also tends to do things their own way. This means one will have to derive some general assumptions upon the road design which can be observed in reality.

Those observations were limited to Denmark, since the data available first of all were applying to this country. Roads can be categorized in different ways. But the main focus for GRIFINOR is on the geometry site; therefore the roads were divided into geometrically manageable groups derived from the data set.

To make the modelling possible it is inevitable to simplify the geometry of a road and ignore small details and for instance take care of the drainage of the road. Before the road is split up into sub-parts we will take a closer look on the road and derive it into objects.

3.1 Roads as objects

Geographic features are usually characterized by its geometry. But what about a road? Where does it start and where does it end. This information is needed for a definition of a road object. So first of all a road must be split into defined objects before any modelling makes sense.

Unfortunately, this is not easy. A natural way of doing it would be to look at existing databases held by official road authorities or road data provided by for instance car navigation companies. Road authorities do in fact have a lot of information about roads. In Denmark for instance the cross-section of the road is described with a road identifier, mileage and record each time a section changes. Also the horizontal and partly the vertical alignments are represented in the database. It would be possible to split up a road in all these sections, defining them as objects and sample together as a road. This approach would be a bottom-up approach since one would take its starting point at the smallest piece of road registered. A top-down approach would be to look at roads registered with identifiers and known by that name in general. Like highway “1” or “101”. These objects would be rather large in terms geometric data and very inefficient to handle as road features in a modelmap. The new IFC data format will try to come up with a way to describe roads built up by objects. These objects will focus on the “as built” situation, since the data standard is meant as a support during the whole life cycle of a road. This especially implies the data exchange during the construction phase and not only during the maintenance period for which road databases are developed today. The maintenance period is by far the longest and a strong emphasis must be put on that. The IFC will probably cover every detail of a road; therefore extractions of that data set can go in every possible direction.

This being said, we will here present our very own way to develop a way to present and represent roads as cut in geometry in GRIFINOR. This paper only presents roads as horizontal and vertical alignments. Cross-roads in especially one plane, but also two or more levels will be dealt with at a later point. Also very complex geometry used in large cross-roads in bigger cities is not a subject of this paper. These designs are typically “one of a kind constructions” and not within reach at the moment. The main focus in this work is to establish as many km roads as possible in an automated manner. We want to cut the road into the terrain with some simple geometric rules using algorithm written for the available datasets covering the country of Denmark to begin with.

In Figure 3 outlines of simple road objects are shown. The road is split up in subsections. These subsections are defined through points in the centre line ($P_i \dots P_{i+n}$) every time a geometry changes. In the figure these sections are equally long, but this is not the case necessarily. When a road has long straights in the alignment only few points are necessary to describe the road, but if the stretch is very much curved a lot more points are necessary for the description. The figure shows that the runway (RW) is a nice and smooth plane. The width will certainly vary a lot and will be defined by the category of the road. The width is set to 3.5 metres as a standard width if no information is available. Each road category will get its own cross-section defined. This example is rather simple solely to present the idea. But since the surface of the road is not really a big problem any kind of cross-section can be used. The runway surface actually has a slope of 2.5% down each site from the centre line. Taking this into account is actually a luxury and in many cases not really required, but since these planes have to be calculated anyway why not do it realistically. The roof-formed shape of straight lines will be replaced by a single-sided slope in curves. The shoulders (SH) will likewise vary a lot from road to road like the runway. Again this will depend on the road category, but the standard width is chosen to be two metres. There will be no ditch in this simplified cross-section. The next step is therefore to define the cutting plane to the terrain (TC). For that the road is elevated one metre in all cases apart from urban areas. If the elevation model of the landscape is made with a laser-scanning, the road should already be present in the points cloud (Hatger and Brenner 2003). In that case the TC area is presumably small. With an old and more inaccurate elevation model the TC area will vary a lot but nevertheless ensure a nice finish towards the terrain.

The road segment derived from the centre line will therefore be represented by six objects, three from each side of the road. This approach uses x,y,z points along the centre line between nodes indicating a cross-road. So each piece of road between nodes will contain a number of segments depending on the curvature. For each segment six objects will be generated.

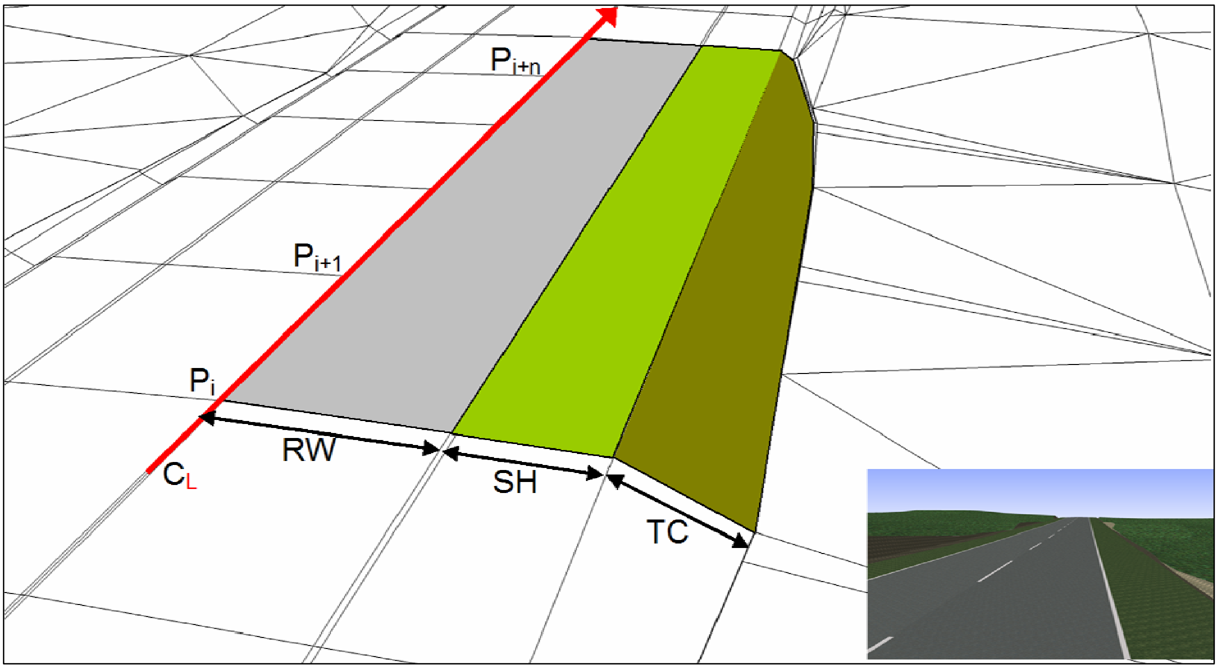

Figure 3 A fragment of road is presented showing the objects

3.2 Outlook to the algorithm

All segments and objects will be generated on the basis of points; and then replace the terrain beneath the road. In Figure 4 a cross-section is shown with the existing terrain beneath the road. The challenge is to calculate the cutting lines on the terrain where the new road segments are supposed to fit in. So the road generation goes as follows:

1. Read data from data set with road segments, nodes and road category.
2. Find the height i.e. the points of the vertical alignment along the segments. If these are not given, find them in the terrain. Generate x,y,z points along the alignment. The road will be elevated one meter above terrain if necessary.
3. Generate accompanying points between RW and SH and between SH and TC.
4. Calculate the points on the terrain surface where TC "hits" the terrain.
5. Remove the terrain between the lines on each side of the road.
6. Generate the road using "constraint triangulation" from the points available (Watson 1981) (Bowyer 1981).

This will probably work for most of the roads which are simple in their geometry, but will give strange results where the road is deviating from the rules used to generate the surface area. This will happen at all cross-roads and where roads change their cross-section design considerably, for instance increasing the number of lanes from four to six.

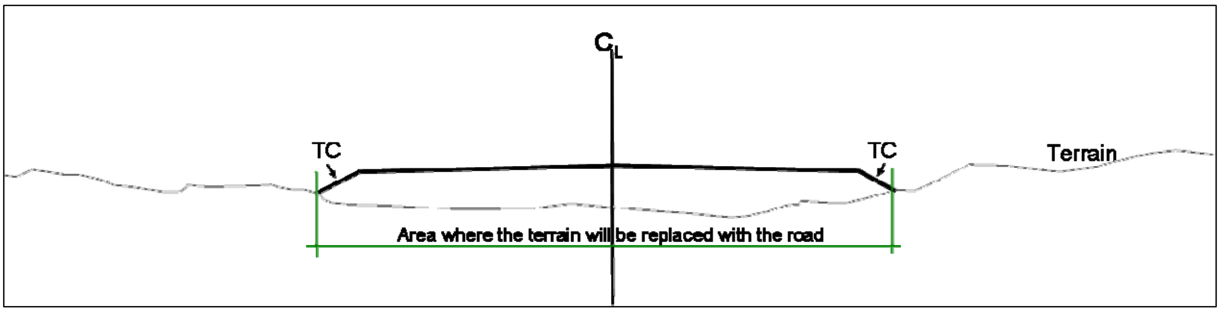

Figure 4 A cross-section showing how the new road surface will replace the existing terrain

3.3 More challenges

This paper only presented the beginning of a development towards road cuts in the terrain. There are a lot of challenges, first of all getting it right in urban areas. But also the curved geometry will have to be sorted out carefully, since curved geometry is very sensitive with respect to the visualisation and the way the eye perceives this kind of geometry. The whole virtual environment is constructed with triangles. Therefore curved surfaces need a lot of triangles to give the impression that the surface actually is a smooth curve and not something else. This has been an issue always, also in CAD systems where circles are drawn by a large number of small lines.

One of the main challenges will be to get the roads look right in the terms of rendering technology of the viewer. Car navigation companies have a lot experience already, but to this day they do not really use the terrain – yet. Another burning issue will be to address LOD support in the road data representation. This is a crucial requirement in order to achieve a solution that is scalable from local (municipality) with precise geometric description to national, continental or global road network coverage with simplified geometries.

4. CONCLUSION AND PERSPECTIVE

For the time being one can argue that modelmaps not really are ready to include roads as surface geometry. They are too complicated to handle and not worth the effort. Nevertheless regarding the GRIFINOR system we decided to go for roads anyway facing the challenge and be one of the first to describe some ways to handle roads together with terrain within modelmaps. It is important to realise that a modelmap is covering a large area modelled in three dimensions. In huge number of projects and visualisations roads have been modelled nicely and cut into the terrain. There are several software packages dealing with only that issue, but all together these models only are covering a limited area and not a whole country like we do in GRIFINOR, where we actually are covering the whole world in one data model.

The biggest challenge probably is to develop a consistent method to generate roads in a reliable and sustainable way. It is important that the methods developed will work with future datasets and has its focus on automation. Modelmaps are using large datasets where it will be impossible to incorporate special solutions for all kind of geometric appearances. One will have to settle with the second best and make it look as good as possible. The method described in this paper is about to be put to test. A future development will hopefully give some methods that can handle roads in a way that they appear as natural as for instance buildings and vegetation do in modelmaps today.

REFERENCES

Bodum, L., Kjems, E., et al. (2005). GRIFINOR: Integrated Object-Oriented Solution for Navigating Real-time 3D Virtual Environments. Geo-information for Disaster Management. P. v. Oosterom, S. Zlatanova and E. M. Fendel. Berlin, Springer Verlag.

Bowyer, A. (1981). "Computing Dirichlet Tesselations." The Computer Journal 24: 162-166.

Hatger, C. and Brenner, C. (2003). EXTRACTION OF ROAD GEOMETRY PARAMETERS FROM LASER SCANNING AND

Kjems, E. and Kolar, J. (2005). From Mapping to Virtual Geography. CUPUM '05 : Computers in Urban Planning and Urban Management, London.

Kolar, J. (2006). Global Indexing Of Topographic Surface For 3D Visualization And Analysis. Development and Planning. Aalborg, Aalborg University. Ph.D.

Kraak, M.-J. and Brown, A. (2000). Web Cartography. London, Taylor & Francis.

Watson, D. F. (1981). "Computing The N-dimensional Delaunay Tesselation with Application To Voronoi Polytopes." The Computer Journal 24: 167-172.