curtin
University of Technology
Perth Western Australia

School of Computing

DEPARTMENT OF GEOGRAPHIC INFORMATION SYSTEMS

Topology - Object - Relation - Graph

(TORG)

by
M A Humphrey Boogaerdt
student 935927k
November1996

Graduate Diploma in GIS

GIS Project 594

10290-7-94

Acknowledgments

This work would not have been possible without the patience and encouragement of my wife Nicoletta Ciffolilli.

I would like also to thank Mr Mark Gahegan, lecturer at the GIS Department of Curtin Uni of Technology; without his help I would not have been able to complete this project.

The work was sponsored by Omni-Scientific Services of Perth.

CONTENTS

List of Figures

Project Aims

This project comprises : The theoretical reasoning for Pattern Recognition by a topology - object - relation - graph.

This project aims to :

1. Set out the theoretical framework of using topology, objects, relations and graphs to recognise patterns in a geological map. In the pilot project the data set will be simple and the type of pattern to be recognised fixed. Once it has been established that it can be done further work can be carried out to make it usable for real life situations such as finding areas of potential mineralisation.

2. Some ideas that are mentioned in the project need further development, they are not the objective of the project but could help further research being carried out.

1. INTRODUCTION[1]

The term Pattern Recognition in this project is used in relation to existing maps or networks. It does not deal with recognising patterns in images, except when mentioned in the general discussion.

Humans recognise patterns by a combination of object-orientation and topology. In the 1960's there was a picture in a Dutch newspaper of a UFO. This UFO was photographed by someone in the UK. Sometime later it wasdiscovered that this spotting of the UFO was a fraud. The fraud was discovered by someone who worked in a brewery and recognised the 'UFO" as a cap or top of a beer barrel. This person recognised this cap, which is part of the very large object-class of caps, because he was in a brewing related industry and happened to be familiar with. If someone was to find this cap somewhere away from its usual surroundings that person would probably recognise that the item is a cap of some kind, but without further knowledge would not exactly know what kind of cap. That person would recognise the item as a cap because of its topology. To be able to recognise an item one needs a combination of topological and object-oriented knowledge.

[1] A lot of the ideas and information in this project comes from all sort of books and articles which are impossible all to quote. For that reason I have included a bibliography. Quoted material is referred to in the list of references.

Humans recognise items because they see patterns and an object. The two are linked. To explain this I will use the example of a simple in one piece moulded plastic chair with a hole in the back and four legs (in this paper when referring to a chair I will always mean this simple one as shown in fig 1).

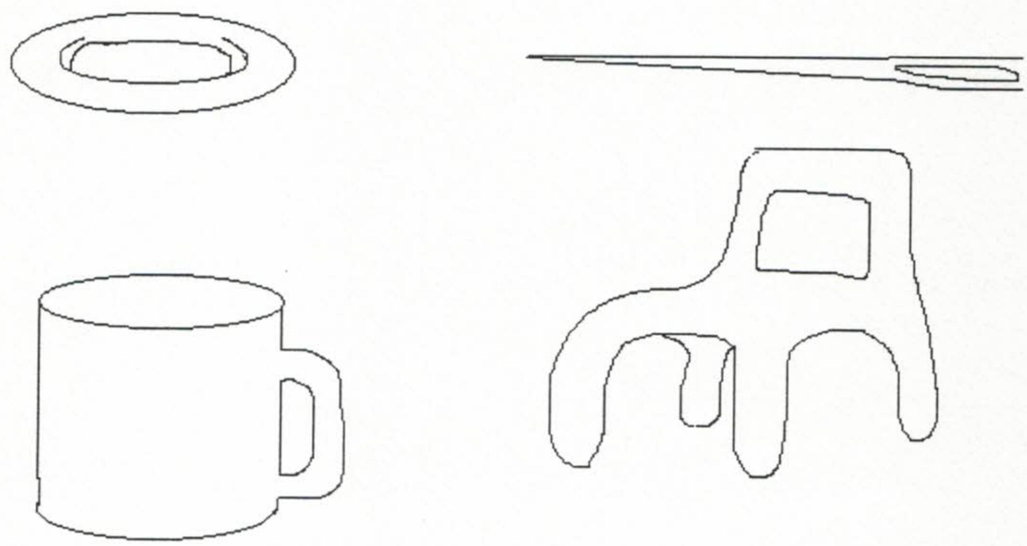


Fig 1 : The doughnut, mug, needle and chair are topological the same , namely the are a torus. They are said to be homeotopic

2. RECOGNITION & TOPOLOGY

2.1 Topology

Even though the example used below is a 3-D object it will illustrate what I am trying to achieve in the project which will deal only with 2-D problems represented in 2D.

Topologically (manifold wise) the chair is a torus because the surface of the chair can be modified in such a way , without rupturing the surface, to a torus by amongst other things shrinking its legs. This torus could also be moulded into a mug or needle (fig 1) . So, to use manifold topology in this way does not work. A simple chair without armrests is topologically a torus, ie the same as a cup. The hole between the chair's back and the seat is the hole in the torus. The legs are just pulled out extensions of the seat. We would not be able to recognise anything because everything can be classified into spheres or single or multiple torusses.

What is needed to recognise objects is a framework for fixed relationships between topological entities. This is where the object-oriented approach comes into play. Coming back to the chair how would we describe a chair. A chair is an object on which one can sit, i.e. it has a seat, a back and four legs (in our example). It does not matter how thick the seat is or how long the legs are or how high the back is, the topological relation ship of these eight items stays the same. The chair can be classified as an object with super and sub classes, describing the different type of chairs, which I will not discuss here. The above mentioned eight building blocks of the chair can be modelled as objects with a topology. The chair forms a topological superspace build up from six topological entities. (figs 2,3,4,5)

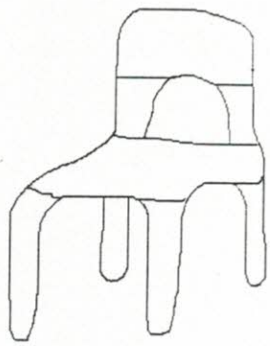


Fig 2 : The simple chair of Fig 1 showing the 8 topological building blocks that make up the chair , meeting adjacent blocks

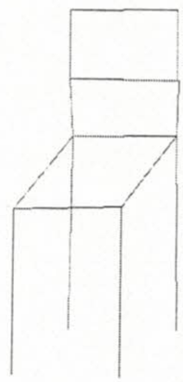
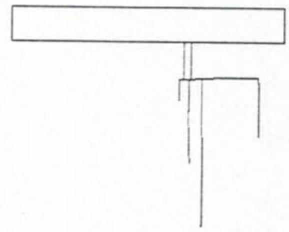


Fig 3 :A caricature of the same simple chair



Another caricature of the same simple chair

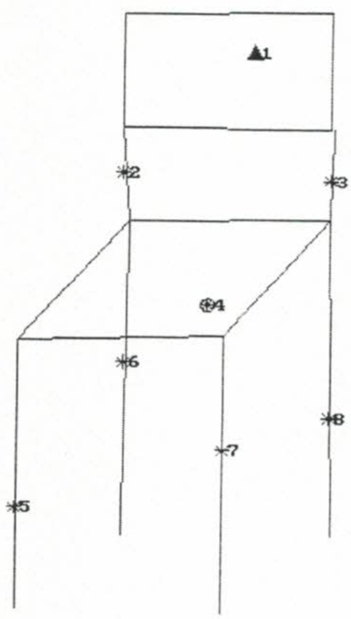


Fig 4

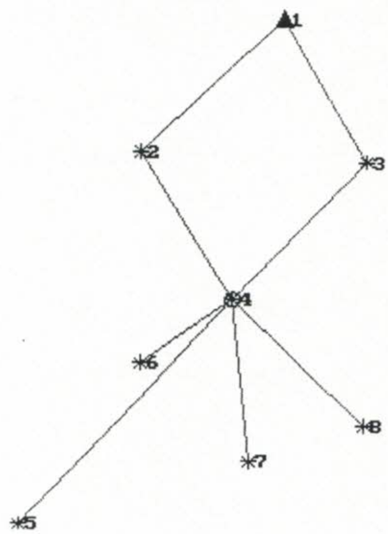


Fig 5 : Graph representing the simple chair

combo_seq_no	vertex_id	tier	
primary key	primary key		
	4	0	0 = anchor
	2	1	
	3	1	
	5	1	
	6	1	
	7	1	
	8	1	
	1	2	

Usually in manifold topology two touching entities (that "meet", Egenhofer & Franzosa,1991) can be merged into a new entity. In pattern recognition this can not be allowed because than the chair, as described above, becomes a torus again. As well the touching entities may be of a different class, e.g. a leg and a seat.

A chair is an object of let say the class "seating_furniture". It has legs which are a sub class of the class of "chairs". However legs are also used in the class of "tables". So legs are part of a multi-rooted class hierarchy (Blashek, 1994).

Topological relationships of an object with adjacent objects can be represented as a graph. Objects can be represented as vertices in a graph, with different properties associated.

Topologically there are no constraints on where the legs or back are joined to the seat, ie is they cannot be randomly positioned. But in reality one would define a chair to have its legs going in the "opposite direction" from the back in relation to the seat *. So there is a need for a "vectorial" component / coordinate system to the topological framework. This coordinate system cannot be the standard Cartesian, because it is internal to the object of "chair" and also a chair is still a chair independent of its position in xyz space. So there is a need for a reference point within the seat to which against the relative directions of the legs and back can be stated. Suggest to use as reference point the “anchor”. Two objects have the same vectorial direction if the angle between their average directions is below a certain limit. This limit is artificial and maybe defined on empirical grounds for an application. The coordinate system cannot be orthogonal it has to be flexible and relative. The units used along the *joists* are *boards* (new term), eg a certain polygon is three *boards* away from the *nail* means that there are two distinct topological objects in between this object and the *nail* and form that relationship. (fig 30).

* Forget possible pieces of art.

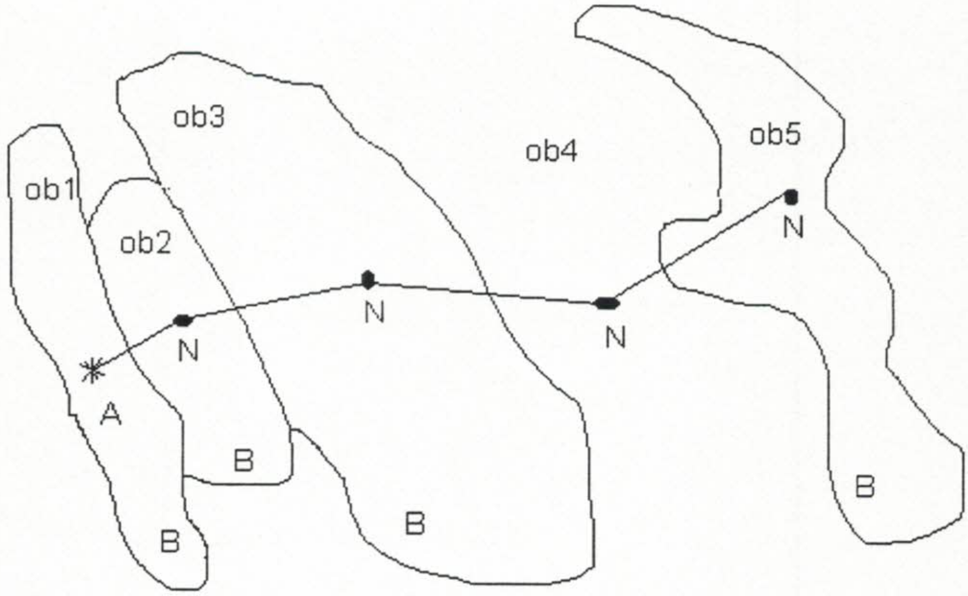


Fig 30 :

A = anchor ; B = board attached along joist by a nail

ob1 - ob4 = objects

ob4 = world, the nail for this vertex represents a null board

The sequence on this joist is always ob1-ob2-ob3-ob4-ob5

Eg ob1 never touches ob3, or ob5. It may touch ob4=world.

"Anchor" = a central point / vertex (the vertex in the graph of the highest degree, but in principle this choice may be arbitrarily) in the graph around one builds up a topological relationship for a control-object . E.g. it makes more sense logically to have the anchor in the seat of the chair than in one of its legs. This is because a chair has to have a seat, but not necessarily the leg the anchor was placed in. An anchor could be the first *nail* of one or more *joists* in a control-object. For searching efficiency it is best to the vertex with the highest degree he anchor.

They cannot use exact directions because "*nail*" (new term) positions are arbitrarily in each object and only represent the schematic graphical output of topological relationships (fig 30). Or may be the vectorial directions of a combo can be recorded as "the same" or "not the same". Or may be a series of fuzzy sets for general directions of "*joists*". For example

≈ opposite 180 ° ± 45 °

≈ right angle 90 ° ± 45 °

≈ same 0 ° ± 45 °

≈ or another classification.

A *"combo"* (= combination object) is build up of two or more objects which have topological relationships whit each other. When these topological relationships are represented as a graph they can form a network. A combination of topological units that corresponds with the search criteria is called the *"control-object"* (c-o). So any chair found during a search that is homeotopic to the reference control-object. To make searches most effective a control object must have a "central" anchor object which is first searched. Then if these objects are found with the correct relationship index then the relevant sub-vertices are checked for their relationships . Can use setup index for composite/complex/hyper objects.

A combo can itself be built up of combos. They could be scale dependent. A combo could be of a certain type of object class. The indication that a certain object is a combo means that one set up n-dimensional system of drilldown. Self similarity- fractal.

The seat of the chair forms the central "*o-space*" (= object-space (new term)). The "*anchor*" forms the center of the topological object against which the vectorial directions are measured along *joists* (new term), which are analogous to axes of a coordinate system. In the case of the chair the legs will have a negative direction and the back a positive one. This is the way the chair is defined in the "object library".

joist =

- string of topological units to represent a certain topological sequence.
- non-linear topological entity, i.e. does not have to form a straight line
- relationship that may form a super object.
- layercake (infers exclusivity of layers) topological relationship could form a super-object
- the length is 2 < n < N ; N set of integers; n = the n-th *nail* the last in this case along the joist.
- joist -- only for a finite use of integers ; built up of topological units that TOUCH or MEET (see later for explanation of these terms).

Joist have *boards* and *nail*s

A board = topological unit namely an object along joist.

Boards are *nail*ed to a joist. *Boards* are cold butted next to each other, they touch (0-cell intersection) or meet (1-cell intersection) or are disjoint. If one board is removed the space becomes part of the (empty) world but when counting *nail*s along the joist the position still exists. Or the *joist* becomes one *nail* shorter.

Boards meet along a joist. The joist is the invisible string that joins the *boards* via their touch and meet points. The position of the *nail* in each board is unimportant. A board has only one *nail* per joist. *Boards* have no specified width. A board meets at most with two other *boards*. A *boards* i (the i-th *nail* along the joist) is at most met by *boards* that have the positions i-1 or i+1. *Boards*, belonging to a certain joist, are in no other way allowed to meet. Where this *nail* is located within the board is arbitrary, except when there is a cluster of data points in the object, while the rest of the space in objects is interpreted.

The joist coordinate system is a non-linear, non-orthogonal multidimensional system. Distances can be measured in integer values only. This is because only the relative distance between objects are measured. Because the objects are topological the size can vary and so one can only count number of objects some object is away (see fig 31).

There is a need for a relationship like a joist because certain relationships are not random / accidental, e.g. a suit of rocks that were deposited sequentially in time. So besides the spatial adjacency relationship for every object index should have a possibility to hook into a joist Or is it more efficient to have a separate index of *joists* which only will be accessed when required

If only two items are attached to each other via a "meet" (Egenhofer & Franzosa, 1991) then there is in general no need for a "*nail*".

Iff [1]ABC on a joist then

A connects to B

B connects to C

A does not connect to C

B can be an empty object along the joist and only exists to have a topological relationship between A & C which are not adjacent and at least one board apart.

May be there is a need for hierarchical system of naming boards along a joist. Maybe the nail id is a combination of the joist_id and the number along the joist or may be a combination of the joist_id and the object_id.

In topological relationships it is to be possible to delete a board or join two *boards* to a new one for modelling purposes. This resembles serial reduction in network graph theory. In this case the number of *nail*s along the joist is reduced by the "number of *boards* minus one" merged. And the reduction is only carried out for modelling.

[1] iff - if and only if

2.2 Recognition

The time it takes you to recognise your parents faces in photographs does not change from ages 5 to 40 to 70 (Kosko, 1994) . Why is it that we recognise people during their lives even when they change. One can only conclude that to recognise something one only needs global essential features i.e. basically features sketched down by a caricaturist, details that changes with age are left out.

In general a picture built up of lines and polygons. In the case of *joists* the vertices are an anchor or *nail*s. Edge enhance a picture in way that only caricature is left possibly using techniques like the medial axis transformation by Ogniewicz & Ilg (1990) , Toussaint (1980), or convert into b&w picture of lines Blum's Medial Axis Transforms (Gold, 1992).

In order to recognise patterns a picture / image has to be reduced to a caricature. Build tables to record the vectorial component of each line. Index this information to graph representation. The Macquarie Dictionary (1981) describes caricature as *"a picture, description etc, ludicrously exaggerating the peculiarities or defects of persons or things".*

From another point of view when we learn we learn something totally new we can only see & understand the "caricature", often called the "basics". Later we can learn/ assoc/see/recognise more detail. This fits with the fact even when we are adults when we are introduced to a new subject, we are first taught the gross outlines and concept, so to say the caricature of the subject. Before able to learn details one has to be at least conscious competent , preferable unconscious competent in knowing the caricature. After mastering that one can study the subject in more detail. If we do not learn in this way we often cannot see the wood from the trees.

A caricature could be seen as a minimal set of topological relations between features. So if a picture can be broken down into a set of minimal topological units, a caricature can be produced of the picture.

Maybe the way we recognise things is by caricatures[1] One could say a caricature is topological homeomorphic to the geometric reality. This is a bit like photographing a simple chair from one corner close-up with a fish-eye lens. The topological relationships stay the same but the geometry is distorted, and it has become a caricature of the geometric reality. Earlier this century D'Arcy Thomson describes in his book "On growth and Form" that he discovered when drawing one species of fish on a rectangular grid and deformed the grid (smooth coordinate transformation) he could get remarkable resemblance of different but related species (Casti, 1994),

To recognise someone need to use various fuzzy sets that can change over time and are needed for recognition ,like :

color hair (multi dimensional) light - dark brown, black red

[1] Naturally a caricature for recognition is often built up of more than the visual aspect, it also builds a caricature of smell, sound, etc.

amount hair

thick - thin

full - bald

straight - curly (different hair cut)

But on the other hand a caricature is enough to recognise someone, cf caricatures of politicians.

Another example of a caricature are mud maps of the real map only highlighting the important parts they are topologically correct. One could state that every map is a caricature of reality.

So in search for orebodies one first makes a caricature of "orebody" environment to see what minimum points are that make the entity recognised.

Babies recognise (not taking in account smells and voice) parents faces regardless whether they are positioned upright, sideways, laying or upside down. This phenomena continues with toddlers that start drawing. They draw , e.g. houses upside down, which could be topologically correct but is always geometrically incorrect. The way they draw (not looking at the technique how to draw straight lines) is in general representing the caricature of the house.

Given the fact that recognition capabilities by humans are enormous, they appear to be based on the topology of caricatures and the context.