PERGAMON

Computers, Environment and Urban Systems
27 (2003) 265–281

Computers, Environment and Urban Systems

www.elsevier.com/locate/compenvurbsys

Empowering a GIS with inductive learning capabilities: the case of INGENS☆

Donato Malerba*, Floriana Esposito, Antonietta Lanza, Francesca A. Lisi, Annalisa Appice

Dipartimento di Informatica, Universitá degli Studi—via Orabona 4, 70126 Bari, Italy

Abstract

Information given in topographic map captions or in GIS models is often insufficient to recognize interesting geographical patterns. Some prototypes of GIS have already been extended with a knowledge-base and some reasoning capabilities to support sophisticated map interpretation processes. Nevertheless, the acquisition of the necessary knowledge is still a demanding task for which machine learning techniques can be of great help. This paper presents INGENS, a prototypical GIS which integrates machine learning tools to assist users in the task of topographic map interpretation. The system can be trained to learn operational definitions of geographical objects that are not explicitly modeled in the database. INGENS has been applied to the task of Apulian map interpretation in order to discover geographic knowledge of interest to town planners.

Keywords: GIS; Map interpretation; Machine learning; Inductive learning

1. Introduction

Data capturing is deemed one of the main impediments to the development of decision support systems based on geographical data. Indeed, data of interest to an application are often reported only on paper maps, whose raster representation is inadequate for subsequent analysis processes. Furthermore, obtaining vector data from a paper map is a very expensive and slow process, since it often requires manual intervention. Data capturing can be difficult even when maps are already available in vector format, since the lack of standards in the coding criteria, adopted

☆ A first version of this paper appeared in the *Proceedings of the International Workshop on Emerging Technologies for Geo-Based Applications*, Ascona, Switzerland, 22–25 May 2000.

* Corresponding author. Tel./fax: +39-80-5443269.
E-mail address: malerba@di.uniba.it (D. Malerba).

0198-9715/02/$ - see front matter
PII: S0198-9715(02)00024-8

by different organizations and private companies, involves writing conversion programs.

While supporting the data acquisition process is important, it is equally useful and challenging to automate the *interpretation* of a map in order to locate some geographical objects and their relations. Unfortunately, information given by the map captions or given as the basis of GIS models is often insufficient to recognize geographical objects of interest in a given application. For instance, a study of the drawing instruction of Bavarian cadastral maps (scale 1:5000) showed that symbols for road, pavement, roadside, garden and so on were defined neither in the caption nor in the GIS-model of the map (Mayer, 1994). These objects require a process of map interpretation, which can be quite complex in some cases. The detection of morphologies characterizing the landscape described in a topographic map, the selection of the important environmental elements, both natural and artificial, and the recognition of forms of territorial organization require abstraction processes and deep domain knowledge that only human experts have. Although these are the patterns that geographers, geologists and town planners are interested in while interpreting a map or analyzing data in a GIS, they are never explicitly represented in topographic maps or in a GIS-model. For instance, in a previous work in cooperation with researchers from the Town Planning Department of the Polytechnic of Bari, an environmental planning expert system was developed for administrators responsible for urban planning (Barbanente et al., 1992). The system was able to provide them with appropriate suggestions but presumed that they had good skills in reading topographic maps to detect some important ground morphology elements, such as system of cliffs, ravines, and so on. These are some examples of morphological concepts that are very important in many civil and military applications, but which are never explicitly represented in topographic maps or in a GIS-model.

Most research in GIS technology has focused on the aspects of data collection, storage and visualization (Laurini & Thompson, 1992). However, the range of GIS applications can be greatly extended by adding interpretation capabilities on georeferenced data. Some GIS prototypes have already been extended with a knowledge-base and some reasoning capabilities, in order to support sophisticated map interpretation processes (Smith, Donna, Sudhakar, & Pankaj, 1997). Nevertheless, these systems have a limited range of applicability for a variety of reasons.

Firstly, providing the system with operational definitions of some environmental concepts is not a trivial task. Often only declarative and abstract definitions, which are difficult to compile into database queries, are available.

Secondly, the operational definitions of some geographical objects are strongly dependent on the data model adopted for the GIS. Finding relationships between density of vegetation and climate is easier with a *raster data model*, while determining the usual orientation of some morphological elements is simpler in a *topological data model* (Frank, 1992).

Thirdly, different applications of a GIS will require the recognition of different geographical elements in a map. Providing the system in advance with all the knowledge required for its various application domains is simply illusory, especially in the case of wide-ranging projects like those set up by governmental agencies.

A solution to these difficulties can be found in machine learning, a branch of artificial intelligence that investigates, among other things, how machines can be trained to recognize some concepts from a given set of examples (Mitchell, 1997). In this paper we present INGENS (INductive GEographic iNformation System), a prototypical GIS extended with a training facility and an inductive learning capability. In INGENS, each time a user wants to formulate queries concerning geographical objects not explicitly modeled in the database, he/she can prospectively train the system to recognize such objects within a special user view. Training is based on a set of examples and counterexamples of geographic concepts of interest to the user (e.g. ravine or steep slopes). Such (counter-) examples are provided by the user who detects them on stored maps by applying browsing, querying and displaying functions of the GIS interface. The symbolic representation of the training examples is automatically extracted from maps, although it is still controlled by the user who can select a suitable level of abstraction and/or aggregation of data. The INGENS learning module implements one or more inductive learning algorithms that can generate geographical object models from the chosen representations of training examples.

The INGENS logical architecture and data model are described in the next section. In Section 3, the map description and learning processes are outlined with reference to a particular application, namely the detection of important environmental and morphological concepts in topographic maps of the Apulia region to support town planning. Conclusions and ideas for future work are reported in Section 4.

2. INGENS software architecture and object data model

The software architecture of INGENS is illustrated in Fig. 1. The interface layer implements a graphical user interface (GUI), which allows the system to be accessed by the following four categories of users:

- Administrators, who are responsible for GIS management.
- Map maintenance users, whose main task is updating the Map Repository.
- Sophisticated end users, who can ask the system to learn operational definitions of geographical objects not explicitly modeled in the database.
- Casual end users, who occasionally access the database and may need different information each time. Casual users cannot train INGENS.
- The GUI is an applet and can be run in any Java-enabled Web browser (Fig. 2).

The layer of the application enablers makes several facilities available to the four categories of INGENS users. In particular, the *Map Descriptor* is the application enabler responsible for the automated generation of first-order logic descriptions of some geographical objects. Descriptors generated by a Map Descriptor are called *operational*. The *Data Mining Server* provides a suite of data mining systems that can be run concurrently by multiple users to discover previously unknown and

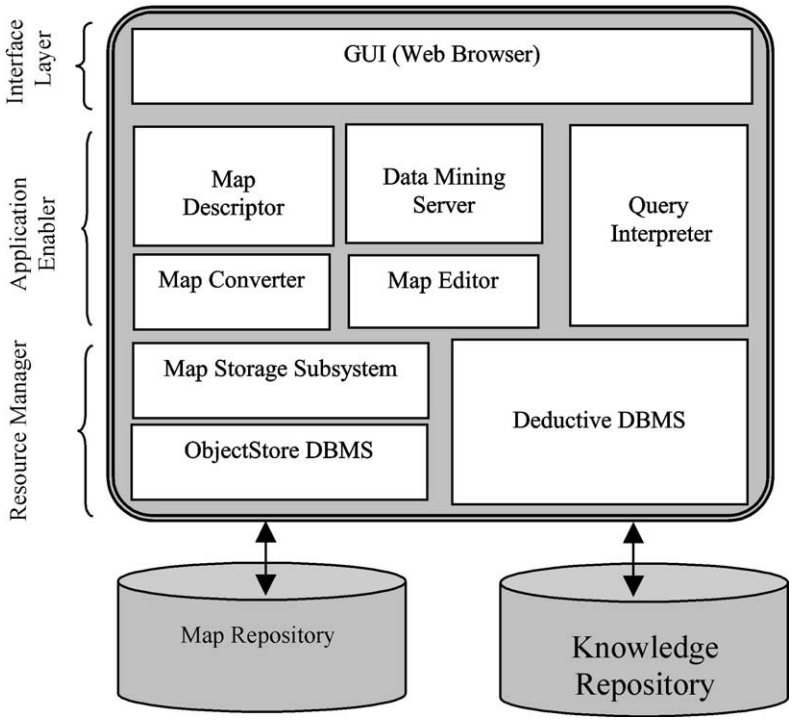


Fig. 1. INGENS three-layered software architecture.

useful patterns in geographic data. In particular, the Data Mining Server provides sophisticated users with an inductive learning system, named ATRE (Malerba, Esposito, & Lisi, 1998), which can generate models of geographical objects from a set of training examples and counter-examples. ATRE will be briefly introduced in Section 3. The *Query Interpreter* allows any user to formulate queries in SDMOQL, an extension of OQL (www.odmg.org) that supports data mining queries (Malerba, Appice, & Vacca, in press). The query can refer to a specific map and can contain both predefined predicates and new predicates, whose operational definition has already been learned. Therefore, it is the Query Interpreter's responsibility to select the involved objects from the Map Repository, to ask the Map Descriptor to generate their logical descriptions and to invoke the inference engine of the Deductive Database, in order to check conditions expressed by both predefined and new predicates. The *Map Converter* is a suite of tools which supports the acquisition of maps from external sources. Currently, INGENS can export maps in Drawing Interchange Format (DXF) by Autodesk Inc. (www.autodesk.com) and can automatically acquire information from vectorized maps in the MAP87 format, defined by the Italian Military Geographic Institute (IGMI) (www.nettuno.it/fiera/igmi/igmit.htm). Since IGMI's maps contain static information on orographic, hydrographic and administrative boundaries alone, a *Map Editor* is required to integrate and/or modify this information (see Fig. 3).

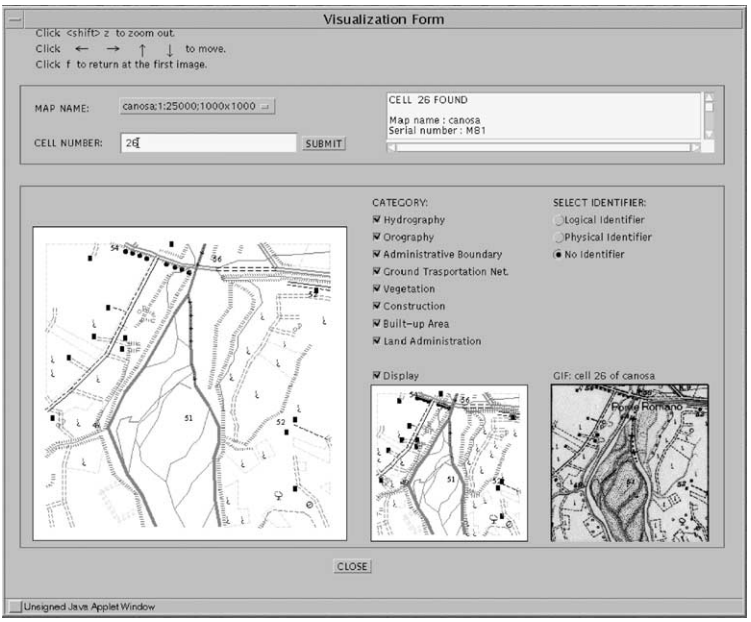


Fig. 2. INGENS graphical user interface displaying map data (upper left corner), cell data (upper right corner), zoomed area (left), geographical objects in the cell (middle), and original bitmap (right).

The lowest layer manages resources like the *Map Repository* and the *Knowledge Repository*. The former is the database instance that contains the actual collection of maps stored in the GIS. Geographic data are organized according to an object-oriented data model, which is described in the next subsection. The object-oriented DBMS used to store data is a commercial one (ObjectStore 5.0 by Object Design, Inc.), so that full use is made of a well-developed, technologically mature aspatial DBMS. Moreover, an object-oriented technology facilitates the extension of the DBMS to accommodate management of geographical objects. The *Map Storage Subsystem* is involved in storing, updating and retrieving items in and from the map collection. As a *resource manager*, it represents the only access path to the data contained in the Map Repository and accessed by multiple, concurrent clients. The Knowledge Repository contains the operational definitions of geographical objects induced by the Data Mining Server. In INGENS, different users can have different definitions of the same geographical object. Knowledge is expressed according to a relational representation paradigm and managed by an XSB-based deductive relational DBMS (Sagonas, Swift, & Warren, 1994).

2.1. The object-oriented data model

In INGENS, data are organized in topographic maps. An association among maps is established when they describe the same territory on different scales. Each map is stored according to a hybrid tessellation–topological model. At the *conceptual*

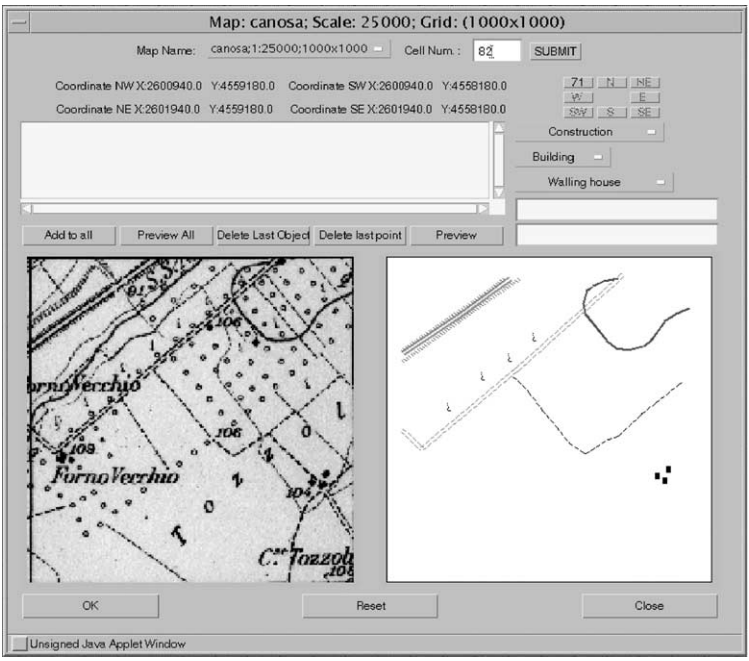


Fig. 3. The Map Editor interface is a Java applet that allows users to reproduce the content of a raster image of a map. Geographical objects can be inserted, deleted, modified and previewed.

level, the model is described by the class diagram in Fig. 4. The tessellation model follows the usual topographic practice of superimposing a regular grid on a map, in order to simplify the localization process. Indeed, each map in the repository is divided into square cells of the same size. Eight one-to-one associations among cells allow map-reading to proceed from a cell to one of its neighbors in the map. For each cell the raster image in GIF format is stored together with its coordinates and component objects. In the topological model of each cell it is possible to distinguish two different structural hierarchies: *physical* and *logical*.

The physical hierarchy describes the geographical objects by means of the most appropriate physical entity, that is: point, line or region. In different maps of the same geographical area, the same object may have different physical representations. For instance, a road can be represented as a line on a small-scale map, or as a region on a large-scale map. Points are described by their spatial coordinates, while (broken) lines are characterized by the list of line vertices, and regions are represented by their boundary lines. Some topological relationships between points, lines and regions are modeled in the conceptual design, namely, points inside a region or on its border and regions disjoining/meeting/overlapping/containing/equaling/covering other regions. The meaning of the topological relationships between regions is a variant on that reported in the nine-intersection model by Egenhofer and Herring (1994), in order to take into account problems caused by approximation errors.

The logical hierarchy expresses the semantics of geographical objects, independent of their physical representation. Since the conceptual data model has been designed to store topographic maps, the entity *logical_object* is a total generalization of eight distinct entities, namely, hydrography, orography, land administration, vegetation, administrative (or political) boundary, ground transportation network, construction and built-up area. Each of them is in turn a generalization, that is, for instance, an administrative boundary must be classified in one of the following classes: city, province, county or state.

According to the principles of all database design methodologies, the conceptual schema in Fig. 4 contains entities taken from the real world and is independent of the characteristics of the DBMS. Generally, the transformation of a conceptual schema into a logical schema is not straightforward. However, this task is simplified when the reference logical model is object-oriented. The main decisions made for the logical model concern the implementation of some relationships, by means of either functions or data members. For instance, the relationship between points and lines is represented by means of instance data members in the corresponding classes, while all the topological relationships between regions are implemented by some functions.

3. Map description and inductive learning: a case study

INGENS has been applied to the recognition of four morphological elements in topographic maps of the Apulia region, Italy, namely, *regular grid system of farms*, *fluvial landscape*, *system of cliffs* and *royal cattle track*. Such elements are deemed relevant for environmental protection, and are of interest to town planners. A regular grid system of farms is a particular model of rural space organization that originated from the process of rural transformation. The fluvial landscape is characterized by the presence of waterways, fluvial islands and embankments. The system of cliffs presents a number of terrace slopes with the emergence of blocks of limestone. A royal cattle track is an ancient path for transhumance that is peculiar to the South-Eastern part of Italy and is characterized by the presence of an uncultivated, quite regular trace, about 90–130 m wide, with a north-west orientation, which is nowadays incorporated into the road network (see Fig. 5).

The territory considered in this application covers 131 km^2 in the surroundings of the Ofanto River, spanning from the zone of Canosa to the Ofanto mouth. More precisely, the examined area is covered by five map sheets on a scale of 1:25,000 produced by the IGMI (Ofanto mouth—165 II SW, Barletta 176 I NW, Canne della Battaglia—176 IV NE, Montegrosso 176 IV SE, Canosa 176 IV SW).

The maps was segmented into square observation units of 1 km^2 each. The choice of the gridding step, which is crucial for the recognition task, was made based on the advice of a team of 15 geomorphologists and experts in environmental planning, giving rise to a one-to-one mapping between observation units of the map and cells in the database.

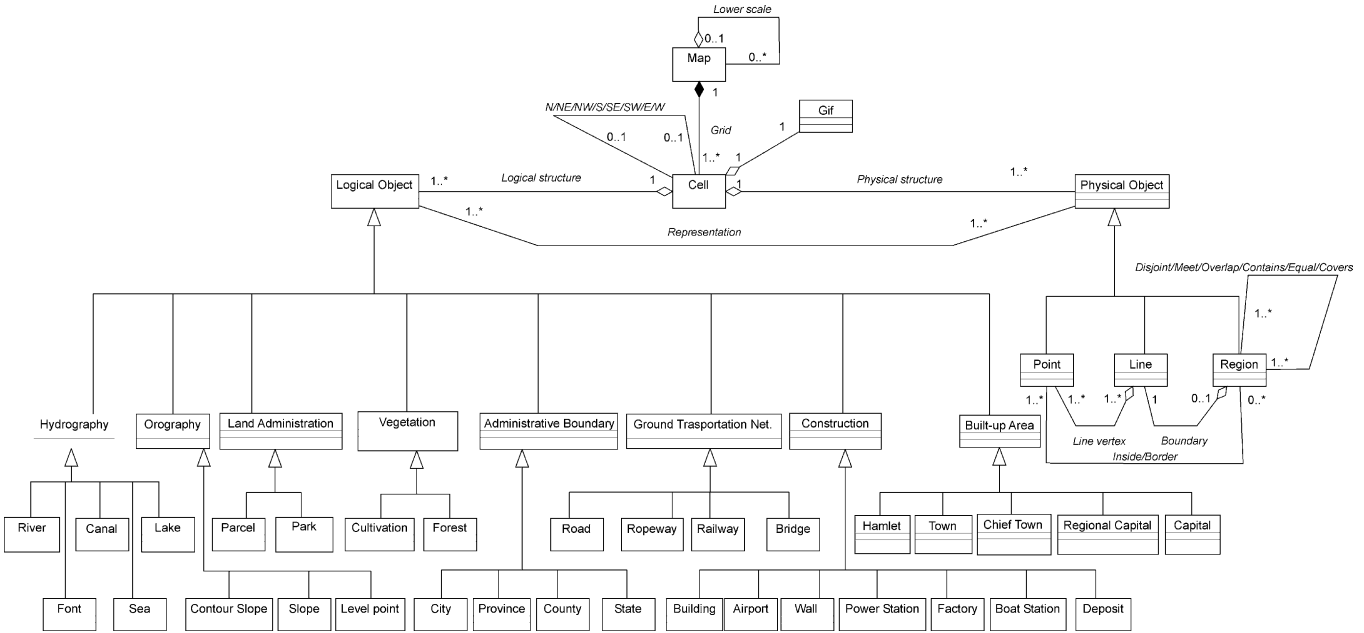


Fig. 4. Class diagram of INGENS conceptual model in Unified Modeling Language (UML). Only some subclasses of logical object are reported.

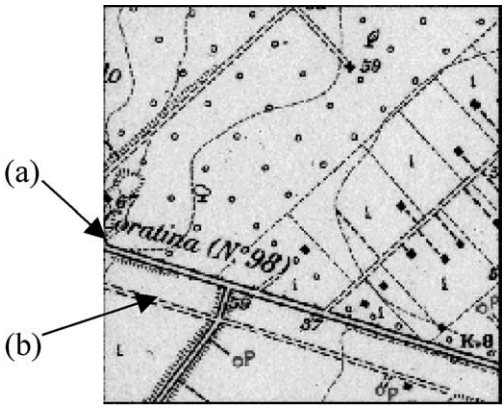


Fig. 5. An example of royal cattle track. It corresponds to the region between the primary road (a) and the inter-farm road (b).

Thus, the problem of recognizing the four morphological elements can be reformulated as the problem of labeling each cell with at most one of four labels. Unlabelled cells are considered uninteresting for environmental protection.

As previously mentioned, INGENS is a GIS extended with a training facility and an inductive learning capability in order to overcome the difficulties related to the acquisition of operational definitions for the recognition task. Details on the automatic generation of the symbolic map descriptions are reported in Section 3.1, while Section 3.2 is devoted to the task of mining spatial classification rules.

3.1. The generation of symbolic map descriptions

Given a cell size and a vector map in the MAP87 format, the *Map Converter* computes the coordinates of all cells in a map and searches the vector file for geographical objects inside or intersecting each cell. Some properties of geographical objects are extracted, such as their physical representation (point, line or region), the coordinates of some traversed points and the altitude of contour slopes. Only information on some layers is automatically extracted from vector descriptions provided by IGMI. Then the description of the cells is manually augmented by means of the *Map Editor*. Editing is also required in order to correctly link some lines that are improperly segmented in the original vector map.

By applying algorithms derived from geometrical, topological, and topographical reasoning, the *Map Descriptor* generates map descriptions in first-order logic.

The descriptors used for this application are listed in Table 1. Since they are quite general, they can also be used to describe maps on different scales. Descriptors can be distinguished as locational (spatial) and non-locational (aspatial). The former encompass geometrical attributes like *line_shape* and *extension*, a directional attribute like *geographic_direction*, a geometrical relationship like *distance*, and topological relationships like *region_to_region* and *point_to_region*. Three non-locational descriptors are *color*, *type_of* and *subtype_of*.

The descriptors *contain*, *type_of*, *subtype_of* and *color* are unconstrained, meaning that they are always computed for each logical component in the cell of interest for the application. On the contrary, all other descriptors are computed only when some conditions reported in the third column of Table 1 are satisfied. The feature extraction process can be formalized as shown in Fig. 6.

In the following, the algorithms for the generation of some sample descriptors are outlined. The descriptor *color* is an unconstrained nominal attribute, whose values depend on the nature of the object itself, so the color is blue if the object belongs to the hydrographic layer, or brown if it belongs to the orographic layer, otherwise it is black. Since *color* is a property of the entity *logical_object* in the data model, its extraction does not require further computation.

Three different values are considered for the constrained nominal descriptor *line_shape*, namely, *straight*, *curvilinear* and *cuspidal*. Let O be a geographical object represented as a line passing through n points (x_i,y_i). The angles of incidence w_i are:

$$w_i = \operatorname{arctg}\frac{x_{i+1} - x_i}{y_{i+1} - y_i}$$

where $i = 1, 2, \ldots, n-1$.

Then, the differences dw_i are calculated as follows:

$$dw_i = w_{i+1} - w_i,$$

where $i = 1, 2, \ldots, n-1$.

The value *cuspidal* is associated to *line_shape*, if the greatest difference among dw_i's exceeds a given threshold τ_{cuspidal}. If the cuspidality condition does not hold, then a check on a straight trend is performed. The value *straight* is generated if all differences dw_i are smaller than a threshold τ_{straight}, which depends on the examined territory. Otherwise, the value *curvilinear* is generated for the object O. Some examples of values computed by this algorithm are reported in Fig. 7.

Finally, the constrained linear descriptor *distance* is computed as follows. Let O and O' be two geographical objects represented by two almost parallel lines passing through n and m points, respectively. Without loss of generality, let us assume that $n \leqslant m$ (see Fig. 8), then, the average distance between O and O' is:

$$\text{distance} = \frac{\sum_{h=1}^{n} d\min_h}{n} \tag{1}$$

where $d\min_h$ is the minimum distance between the h-th point of O and any point of O'.

The descriptions obtained for each cell are quite complex, since some cells contain dozens of geographical objects of various types. For instance, the cell shown in Fig. 2 contains 132 distinct objects, and its description is a clause with 756 literals in the body. A partial description is given in Fig. 9.

Table 1
Descriptors used for the application to Apulian map interpretation

Descriptor	Meaning	Constraint	Domain	
			Type	Values
contain(X,Y)	Cell X contains the geographical object Y		Boolean	{true, false}
type_of(Y)	Type of Y		Nominal	33 nominal values
subtype_of(Y)	Specialization of the type of Y		Nominal	101 nominal values that are specializations of the type_of domain
color(Y)	Color of Y		Nominal	{blue, brown, black}
altitude(Y)	Altitude of Y	Y is represented by a point	Linear	[0...MAX_ALT]
area(Y)	Area of Y	Y is represented by a region	Linear	[0...MAX_AREA]
density(Y)	Density of Y	Y is vegetation or buildings	Ordinal	Symbolic names chosen by expert user
extension(Y)	Extension of Y	Y is represented by a line	Linear	[0...MAX_EXT]
geographic_direction(Y)	Geographic direction of Y	Y is represented by a medium–long line	Nominal	{north, east, north_west, north_east}
line_shape(Y)	Shape of the linear object Y	Y is represented by a line	Nominal	{straight, curvilinear, cuspidal}
line_to_line(Y,Z)	Spatial relation between two lines Y and Z	Y and Z are represented by two medium-long lines	Nominal	{almost parallel, almost perpendicular}
distance(Y,Z)	Distance between two lines Y and Z	Y and Z are almost parallel	Linear	[0...MAX_DIST]
region_to_region(Y,Z)	Spatial relation between two regions Y and Z	Y and Z are represented by two regions	Nominal	{disjoint, meet, overlap, covers, covered_by, contains, equal, inside}
line_to_region(Y,Z)	Spatial relation between a line Y and a region Z	Y (Z) is represented by a line (region)	Nominal	{along_edge, intersect}
point_to_region(Y,Z)	Spatial relation between a point Y and a region Z	Y (Z) is represented by a point (region)	Nominal	{inside, outside, on_boundary, on_vertex}

```
Procedure extract_features(cell)

foreach object y in cell do
    extract the unconstrained descriptors
    extract all the admissible descriptors w.r.t. the
                                        logic type of y
endforeach
    return symbolic cell description
```

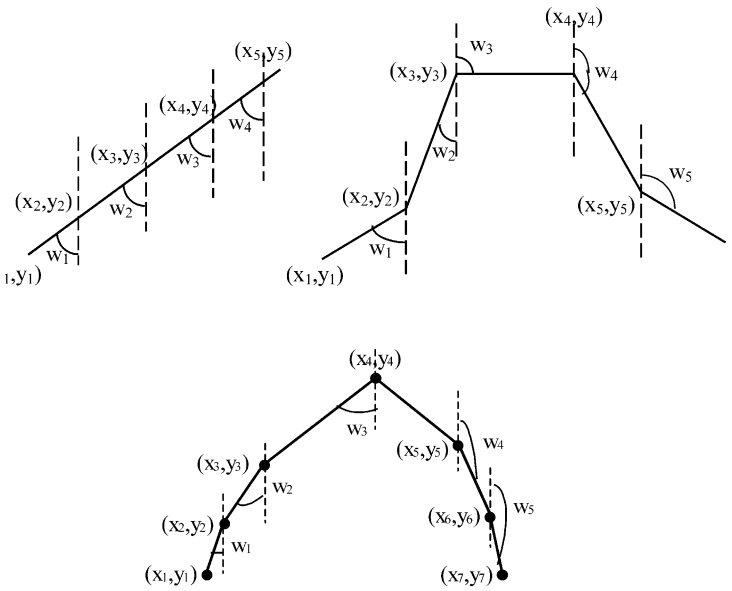


Fig. 7. Example of a straight line (upper left), a curvilinear shape (upper right), and a cuspidal shape (bottom).

3.2. Mining spatial classification rules

Sophisticated end users may train INGENS to learn operational definitions of some geographical objects that are not explicitly modeled in the database, such as those relevant for the application to Apulian map interpretation. In order to support this category of users, the *Data Mining Server* makes an inductive learning system available to them, namely ATRE. This system can induce first-order logic descriptions of some concepts from a set of training examples (Michalski, 1983). A distinguishing feature of ATRE is that it can induce recursive definitions of concepts and can autonomously discover concept dependencies, the latter being an important functionality for many map interpretation problems (Malerba, Esposito, Lanza, & Lisi, 2001). This system, which has been applied to the interpretation of Apulian maps, is briefly presented below.

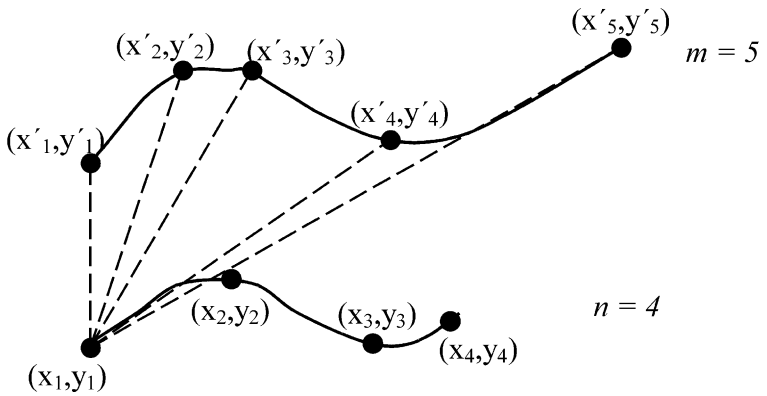


Fig. 8. Computation of the average distance between two almost parallel lines.

The learning problem solved by ATRE can be formulated as follows:

Given

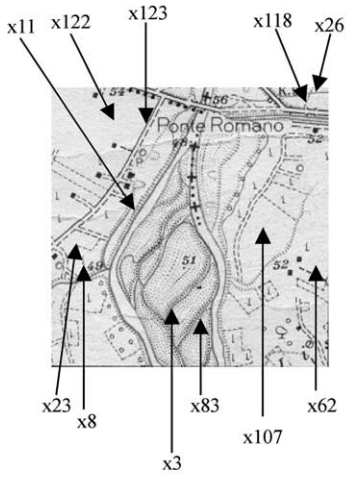
- a set of concepts C_1, C_2,..., C_r to be learned,
- a set of observations O described in a language $\mathcal{L}_O$,
- a background knowledge BK described in a language $\mathcal{L}_{BK}$,
- a language of hypotheses $\mathcal{L}_H$,
- a user's preference criterion PC,

Find

a (possibly recursive) logical theory T for the concepts C_1, C_2,..., C_r, such that T is complete and consistent with respect to O and satisfies the preference criterion PC.

The *completeness* property holds when the theory T explains all observations in O of the r concepts C_i, while the *consistency* property holds when the theory T explains no counter-example in O of any concept C_i. The satisfaction of these properties guarantees the correctness of the induced theory with respect to the given set of observations, O. Whether the theory T is actually correct, that is, whether it classifies correctly all other examples not in O, is an extra-logical matter, since no information on the generalization accuracy can be extracted from the training data themselves. In fact, the selection of the "best" theory is always made on the basis of an inductive *bias*, either embedded in some heuristic function or expressed by the user of the learning system (preference criterion).

In the context of geographical knowledge discovery, each C_i is a geographical object not explicitly reported in map captions, such as "fluvial landscape". Although signs and symbols on a map correspond to general concepts (e.g. river, boundary, and built-up area) which are assumed to be shared by both the map creator and the map user (Keates, 1996), other geographical objects interesting for the latter might not have been explicitly modeled by the former. In this case inductive learning can support sophisticated users by generating the operational definitions of these



class(x1)=fluvial_landscape ←
contain(x1,x2)=true, contain(x1,x3)=true, …,
contain(x1,x133)=true,
type_of(x2)=street, type_of(x3)=canal,…, type_of(x11)=river, …,
type_of(x133)=building,
subtype_of(x2)=main_road, …, subtype_of(x133)=lamp,
color(x2)=black, …, color(x11)=blue, …, color(x133)=black,
extension(x2)=340.352, …, extension(x11)=1131.84, …,
extension(x129)=147.281,
geographic_direction(x14)=east, …,
geographic_direction(x62)=north_west, …,
geographic_direction(x127)=north_east,
line_shape(x2)=curvilinear, line_shape(x3)=curvilinear , …,
line_shape(x11)=curvilinear, …, line_shape(x129)=cuspidal,
altitude(x28)=49.00, altitude(x77)=51.00, …, altitude(x131)=52.00,
area(x17)=10562.5, area(x23)=5762.5, …, area(x122)=97712.5,
density(x17)=low, density(x23)=low, …, density(x122)=medium,
line_to_line(x3,x83)=almost_parallel, …,
line_to_line(x26,x118)=intersect, …,
line_to_line(x123,x125)=almost_parallel,
region_to_region(x17,x23)=disjoint,
region_to_region(x17,x43)=disjoint, …,
region_to_region(x107,x121)=disjoint,
distance(x3,x83)=120.00, …, distance(x123,x125)=80.00,
line_to_region(x2,x122)=adjacent, …,
line_to_region(x26,x118)=intersect, …,
line_to_region(x123,x122)=adjacent,
point_to_region(x8,x23)=outside, …,
point_to_region(x8,x100)=inside,…,
point_to_region(x131,x118)=outside

Fig. 9. A partial logical description of the cell shown in Fig. 2. It represents an example of fluvial landscape. Constant $x1$ represents the whole cell, while all other constants denote the 132 enclosed objects.

geographical objects from training observations (see Fig. 9), which are described by means of a logic language $\mathcal{L}_{O}$, whose operational descriptors are listed in Table 1.

The *language of hypotheses* $\mathcal{L}_{H}$ is that of *linked*, *range-restricted* definite clauses (De Raedt, 1992). An exemplification is reported in Fig. 10.

The background knowledge is expressed in a language $\mathcal{L}_{BK}$ with the same constraints as the language of hypotheses. It defines the relevant domain knowledge. The following is an example of spatial background knowledge:

$$close\_to(X, Y) = true \leftarrow region\_to\_region(X, Y) = meet$$

$$close\_to(X, Y) = true \leftarrow close\_to(Y, X) = true$$

which states that two adjacent zones are also close. These rules for qualitative spatial reasoning can be used by the learning system to derive different spatial relations not explicitly represented in the logical description of observations.

As regards the application to the interpretation of Apulian maps, some results are briefly presented below. Twenty-nine cells from the map of Canosa were selected to train the system. Each cell was assigned to one of the following five classes: system of farms, fluvial landscape, system of cliffs, royal cattle track and other. The last

```
class(X1)=fluvial_landscape ← contain(X1,X2)=true, type_of(X1)=cell,
        type_of(X2)=river, color(X2)=blue,  extension(X2)∈[839.394 .. 1639.04],
        contain(X1,X3)=true

class(X1)=system_of_farms ← contain(X1,X2)=true, region_to_region(X2,X3)=disjoint,
        density(X3)=high, region_to_region(X2,X4)=meet,
        region_to_region(X4,X5)=meet, region_to_region(X2,X5)=meet,
        type_of(X1)=cell, area(X2)∈[12381.2 .. 25981.2], type_of(X2)=parcel
```

Fig. 10. A fragment of logical theory induced by ATRE.

class simply "represents 'the rest of the world,'" and no classification rule is generated for it. Indeed, its assigned cells are not interesting for the problem of environmental protection being studied, and they are always used as negative examples when ATRE learns classification rules for the remaining classes.

A fragment of the logical theory induced by ATRE is reported in Fig. 10. The first clause explains all eight training examples of fluvial landscape. It states that cells labeled as *fluvial_landscape* contain a blue object of type 'river' whose extension is between 839.394 and 1639.04 m. Therefore, the presence of a river is not sufficient to recognize a fluvial landscape in a cell. The extension of the river in a cell should also be relatively long.

The second clause refers to the system of farms and covers all eight training examples. From the training observations, the machine learning system induced the following definition: "There are three adjacent regions (X2, X4, X5), one of which is certainly a medium-sized parcel (the area is between 12381.2 and 25981.2 m^2), and there is a fourth region (X3), disjoint from the parcel, with a high density (presumably vegetation)". Some experimental results obtained in a previous work are reported in Esposito, Lanza, Malerba, and Semeraro (1997).

The operational definitions generated by the *Data Mining Server* can be used to retrieve new instances of the learned concepts from the Map Repository and to facilitate the formulation of a query involving geographical objects not present in map captions. For instance, by submitting the following query in SDMOQL:

```
SELECT  C
FROM    M in Map, C in Cell, R in Road
WHERE   M->name="Canosa" AND C->map=M AND R->log_incell=C AND
        R->type_road="main_road" AND class(C)=fluvial_landscape
```

the user asks INGENS to find all cells in the Canosa map that are classified as fluvial landscape and contain a main road. To check the condition defined by the predicate *class(C)=fluvial_landscape*, the *Query Interpreter* generates the symbolic description of each cell in the map and asks the Query Engine of the Deductive Database to prove the goal ← *class(C)=fluvial_landscape* given the logic program in Fig. 10. The result set will also include the cell in Fig. 9.

4. Conclusions

Information given by map legends or given as basis of data models in geographical information systems (GIS) is often insufficient to recognize not only *geographical objects* relevant for a certain application, but also *patterns* of geographical objects which geographers, geologists and town planners are interested in. Moreover, a GIS user may find it quite difficult to describe such geographical objects or patterns in a query language. That would be tantamount to providing GIS with an operational definition of abstract concepts often reported in texts and specialist handbooks. In order to support GIS users in their activity, a new approach has been proposed in this paper. The idea is to ask users for a set of classified instances of the geographical objects or patterns which interest them, and then apply machine learning tools and techniques to generate the operational definitions for such patterns. These definitions can be either used to retrieve new instances from the Map Repository or to facilitate the formulation of a query. INGENS is a prototypical GIS with learning capabilities that has been designed and implemented in order to provide users with a training facility. An application of the system to the problem of Apulian map interpretation has been reported in this paper in order to illustrate the advantages of the proposed approach.

This work is still in progress and many problems have to be solved. The segmentation of a map in a grid of suitably sized cells is a critical factor, since over-segmentation leads to a loss of recognition of global effects, while under-segmentation leads to large cells with an unmanageable number of components. To cope with the first problem, it is necessary to consider the *context* of a cell, that is, the neighboring cells, both in the training phase and in the recognition phase. To solve problems caused by under-segmentation it is crucial to provide users with appropriate abstraction operators that cover up irrelevant information in the cell description. An empirical indication of possible under-segmentation problems comes from the number of components in each cell, while problems of over-segmentation can be related to the difficulty of the trainer in assigning each example to a unique class.

INGENS can be extended in two directions. Firstly, a set of generalization and abstraction operators will be implemented in order to simplify the complex descriptions currently produced by the *Map Descriptor*. Secondly, further algorithms for the discovery of spatial association rules and the quantitative interpretation of topographic maps will be embedded in the *Data Mining Server*.

Acknowledgements

The authors are grateful to Dino Borri, Angela Barbanente, and Mauro Iacoviello of the Department of Town Planning, Polytechnic of Bari, for their help in the application to Apulian map interpretation. Thanks also to the IGMI for having provided the information and the vector map extracts, which made it possible to set up a Map Converter tool. The authors also wish to thank Lynn Rudd for her help in reading the manuscript.

References

Barbanente, A., Borri, D., Esposito, F., Leo, P., Maciocco, G., & Selicato, F. (1992). Automatically acquiring knowledge by digital maps in artificial intelligence planning techniques. In A. U. Frank, I. Campari, & U. Formentini (Eds.), *Theories and methods of spatio-temporal reasoning. Lecture notes in artificial intelligence* (pp. 89–100). Berlin: Springer.

De Raedt, L. (1992). *Interactive theory revision: an inductive logic programming approach*. London: Academic Press.

Egenhofer, M. J., & Herring, J. R. (1994). Categorising topological spatial relations between point, line, and area objects. In M. J. Egenhofer, D. M. Mark, & J. R. Herring (Eds.), *The 9-intersection: formalism and its use for natural language spatial predicates. Technical Report 94–1*. Santa Barbara: NCGIA.

Esposito, F., Lanza, A., Malerba, D., & Semeraro, G. (1997). Machine learning for map interpretation: an intelligent tool for environmental planning. *Applied Artificial Intelligence*, *11*(7–8), 673–695.

Frank, A. U. (1992). Spatial concepts, geometric data models, and geometric data structures. *Computers & Geosciences*, *18*(4), 409–417.

Keates, J. S. (1996). *Map understanding*. Edinburgh: Longman.

Laurini, R., & Thompson, D. (1992). *Fundamentals of spatial information systems*. San Diego: Academic Press.

Malerba, D., Appice, A. & Vacca, N. SDMOQL: an OQL-based data mining query language for map interpretation tasks. *Proceedings of the EDBT 2002 Workshop on "Database Technologies for Data Mining"* (pp. 3–18), Prague, Czech Republic.

Malerba, D., Esposito, F., Lanza, A., & Lisi, F. A. (2001). Machine learning for information extraction from topographic maps. In H. J. Miller, & J. Han (Eds.), *Geographic data mining and knowledge discovery* (pp. 291–314). London, UK: Taylor and Francis.

Malerba, D., Esposito, F., & Lisi, F. A. (1998). Learning Recursive Theories with ATRE. In H. Prade (Ed.), *Proc. of the 13th European Conf. on Artificial Intelligence* (pp. 435–439). Chichester: Wiley.

Mayer, H. (1994). Is the knowledge in map-legends and GIS-models suitable for image understanding? *International Archives of Photogrammetry and Remote Sensing*, *30*(4), 52–59.

Michalski, R. S. (1983). A theory and methodology of inductive learning. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: an artificial intelligence approach* (pp. 83–134). Palo Alto: Tioga Publishing.

Mitchell, T. M. (1997). *Machine learning*. New York: McGraw-Hill.

Sagonas, K. F., Swift, T., & Warren, D. S. (1994). XSB as an Efficient Deductive Database Engine. In R. T. Snodgrass, & M. Winslett (Eds.), *Proceedings of the 1994 ACM SIGMOD International Conference on Management of Data* (pp. 442–453). Minneapolis, Minnesota: SIGMOD Record.

Smith, T., Donna, P., Sudhakar, M., & Pankaj, A. (1997). KBGIS-II: a knowledge-based geographic information system. *International Journal of Geographic Information Systems*, *1*(2), 149–172.