Generating Logic Descriptions for the Automated Interpretation of Topographic Maps

Antonietta Lanza, Donato Malerba, Francesca A. Lisi,
Annalisa Appice, and Michelangelo Ceci

Dipartimento di Informatica, Università degli Studi di Bari
via Orabona 4, 70125 Bari, Italy
{lanza,malerba,lisi,appice,ceci}@di.uniba.it

Abstract. Automating the interpretation of a map in order to locate some geographical objects and their relations is a challenging task, which goes beyond the transformation of map images into a vectorized representation and the recognition of symbols. In this work, we present an approach to the automated interpretation of vectorized topographic maps. It is based on the generation of logic descriptions of maps and the application of symbolic Machine Learning tools to these descriptions. This paper focuses on the definition of computational methods for the generation of logic descriptions of map cells and briefly describes the use of these logic descriptions in an inductive learning task.

1 Introduction

Automating the interpretation of a map in order to locate some geographical objects and their relations [7] is a challenging task, which goes beyond the transformation of map images into a vectorized representation [3] and the extraction of single elements (symbol recognition), such as buildings [8] and roads [1]. In fact, though geographical information systems (GIS) store vectorized maps, information given as the basis of GIS models is often insufficient to recognize *geographical objects* relevant for a certain application. This deficiency is even more evident for *patterns* of geographical objects that are of interest to geographers, geologists and town planners. Map interpretation tasks, such as the detection of morphologies characterizing the landscape, the selection of both natural and artificial environmental elements, and the recognition of territorial organization forms require abstraction processes and deep domain knowledge that only human experts have.

In this work, we present an approach to the automated interpretation of vectorized topographic maps that is based on the application of machine learning tools to logic descriptions of maps. These descriptions are conjunctions of both attributive and relational features, which are automatically generated from vectorized maps. Two of the main problems in automating the map description process are the choice of an appropriate set of features and the definition of computational methods for their extraction. As to the former issue, the contribution of experts interested in automating the identification of some morphological elements in topographic maps is crucial. In a

previous work [4], the collaboration of experts in geomorphology and territory planning allowed us to define a sufficiently general set of relevant features, which are *used by humans* in their map interpretation process.

In this work, we investigate the second issue, that is, the definition of computational methods for the generation of the chosen set of features. The next section is devoted to the presentation of some feature extraction algorithms, while Section 3 illustrates the use of the automatically generated logic descriptions in an inductive learning task. Conclusions are drawn and ideas for future work are presented in Section 4.

2   Generating Logic Descriptions of Map Cells

To simplify the localization process we follow the usual topographic practice of superimposing a regular grid on a map. This corresponds to a virtual segmentation of a map into square *cells*, whose size depends on the kind of geographical object or pattern we want to recognize in a cell.

The content of a map cell is described by means of a set of *features*. Here the term feature is intended as a characteristic (property or relationship) of a geographical entity. This meaning is similar to that commonly used in Pattern Recognition and differs from that attributed by people working in the field of GIS, where the term denotes the unit of data which represents a geographical entity in computer systems, according to the Open GIS Consortium terminology [16, 13].

Several spatial features can be extracted from vectorized maps. They can be distinguished on the basis of their *arity*, that is the number of arguments they can take. An *attribute* is a feature that predicates the property of one spatial object, and can be represented by a unary function or predicate. A *relation* is a feature that holds between two or more objects and can be represented by an n-ary ($n>1$) function or predicate. Spatial relations are actually conditions imposed on object locations.

According to their *nature*, it is possible to classify features as follows:

- *Locational* features, if they concern the location of objects. Locations are represented by numeric values that express co-ordinates.
- *Geometrical* features, if they depend on the computation of some metric/distance. Examples are area, perimeter, and length. Their domain is typically numeric.
- *Topological* features, if they are relations that are invariant under the topological transformations (translation, rotation, and scaling). Topological features are generally represented by nominal values.
- *Directional* features, if they concern orientation. Generally, a directional feature is represented by means of nominal values.

Geo-referenced objects also have aspatial features, such as name, layer, and temperature. Many other features can be extracted from maps, some of which are hybrid, in the sense that they merge properties of two or more feature categories. For instance, features that express parallelism or orthogonality between lines are both topological and geometrical. They are topological since they are invariant with respect to translation, rotation and scaling, while they are geometrical, since their definition is based on their angle of incidence. The relation of "faraway-west", whose

definition mixes both directional and geometrical concepts, is another example of a hybrid spatial feature. Finally, some features might mix spatial relations with aspatial properties, such as the feature that describes coplanar roads by combining the condition of parallelism with the type of spatial objects (road).

Vectorized representations of topographic maps have been mainly used for rendering purposes in the field of GIS. Few works on feature extraction from vectorized maps are reported in the literature [2] and they refer only to cadastral maps. The first application of feature extraction algorithms to vectorized topographic maps can be found in the work by Esposito *et al.* [4]. For environmental planning tasks, fifteen features were specified with the help of domain experts (see Table 1). Since they are quite general, they can also be used to describe maps on different scales.

When features are explicitly modeled in a GIS, as is the case of the (sub-)type, color, and altitude of a geographical object, they can be easily computed. On the contrary, other features have to be extracted from vectorized maps. Actually, computational methods for feature extraction from vectorized maps are far from being a simple "adaptation" of existing graphics recognition algorithms. In fact, the different data representation (raster vs. vector) makes the available algorithms totally unsuitable for vectorized maps, as is the case of all filters based on mathematical morphology [17]. Each feature to be extracted needs the development of a specific procedure that relies on the geometrical, topological and topographical principles, which are involved in the definition of that feature. The problem is similar to that faced by some geographers who tried to capture the semantics associated with natural-language spatial relations through formal concepts of geometry, such as the 9-intersection model [15]. However, this model is appropriate for expressing only topological relations between linear and areal objects and cannot be used to define the semantics of some features, such as the relation of "parallelism" between contour slopes, which is deemed important for the recognition of some morphologies in the territory. In order to express the semantics of some features used in geometrical, topological and topographical reasoning, we propose the adaptation of 2D mathematical methods used for raster map processing [14].

For instance, the attribute *line_shape(O)* indicates the shape and the trend of the object *O*. It is a geometric attribute and has a nominal domain with values: straight, curvilinear, and cuspidal. It is extracted only for linear objects according to the following procedure. Let *O* be represented by *n* coordinate pairs (x_i, y_i); the angles of incidence w_i are:

$$w_i = \arctg \frac{x_{i+1} - x_i}{y_{i+1} - y_i}, \text{ where } i = 1, 2, \ldots, n-1 .$$

Then, the differences dw_i are calculated as follows:

$$dw_i = w_{i+1} - w_i, \text{ where } i = 1, 2, \ldots, n-1 .$$

The *cuspidal* value is associated to *line_shape*, if the greatest difference between dw_i's exceeds a given threshold $\tau_{cuspidal}$. If the cuspidality condition does not hold, then a check on a straight trend is performed. The *straight* value is generated if all differences dw_i are smaller than a threshold $\tau_{straight}$, which depends on the examined

territory. Otherwise, the *curvilinear* value is generated for the object O. The case of the cuspidal line is illustrated in Figure 1, where dw_3 is the greatest difference between dw_i and is greater than the given threshold.

A further example is the computation of the *distance* relation between two "parallel" lines. Let O and O' be two geographical linear objects represented by n and m coordinate pairs, respectively. Without loss of generality, let us assume that $n \leq m$. The algorithm first computes $d\min_h$ as the minimum distance between the h-th point of O and any point of O' (see Figure 2). Then, the distance between O and O' is computed as follows:

Table 1. Features extracted for the generation of map descriptions

| Feature	Meaning	Type	Domain	
			Type	**Values**
contain(X,Y)	Cell X contains object Y	Topological relation	boolean	{true, false}
type_of(Y)	Type of Y	Aspatial attribute	nominal	33 nominal values
subtype_of(Y)	Specialization of the type of Y	Aspatial attribute	nominal	101 nominal values that are specializations of the type_of domain
color(Y)	Color of Y	Aspatial attribute	nominal	{blue, brown, black}
area(Y)	Area of Y	Geometrical attribute	linear	[0..MAX_AREA]
density(Y)	Density of Y	Geometrical attribute	ordinal	Symbolic names chosen by expert user
extension(Y)	Extension of Y	Geometrical attribute	linear	[0..MAX_EXT]
geo_direction(Y)	Geographic direction of Y	Directional attribute	nominal	{north, east, north_west, north_east}
line_shape(Y)	Shape of the linear object Y	Geometrical attribute	nominal	{straight, curvilinear, cuspidal}
altitude(Y)	Altitude of Y	Geometrical attribute	linear	[0.. MAX_ALT]
line_to_line(Y,Z)	Spatial relation between two lines Y and Z	Hybrid relation	nominal	{almost parallel, almost perpendicular}
distance(Y,Z)	Distance between two lines Y and Z	Geometrical relation	linear	[0..MAX_DIST]
region_to_region(Y,Z)	Spatial relation between two regions Y and Z	Topological relation	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	Hybrid relation	nominal	{along_edge, intersect}
point_to_region(Y,Z)	Spatial relation between a point Y and a region Z	Topological relation	nominal	{inside, outside, on_boundary, on_vertex}
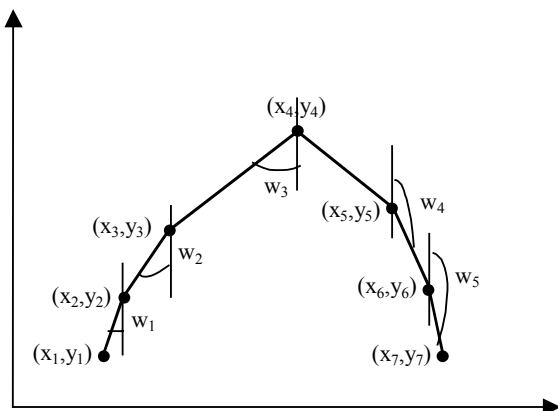
Fig.1. Graphic representation of a cuspidal line

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

The complexity of this simple feature extraction algorithm is O(mn), though less computationally expensive solutions can be found by applying multidimensional access methods [6].
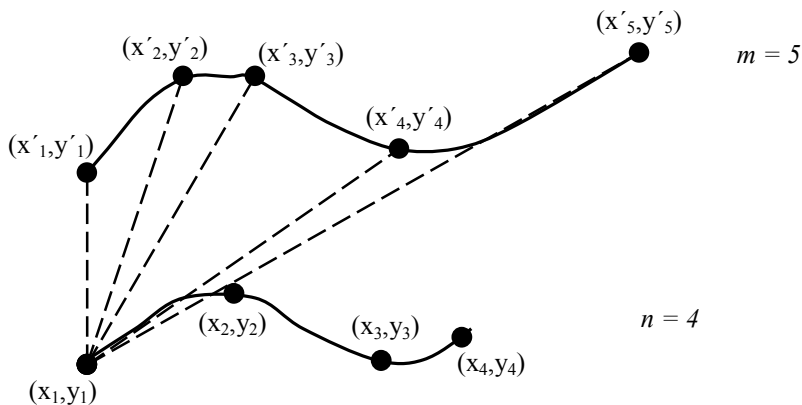

Fig.2. Computation of the average distance between two almost parallel lines

3   Learning from Logical Descriptions of Map Cells

Computational methods for the generation of logical descriptions of map cells have been implemented in the Map Descriptor module of INGENS, a prototypical GIS with inductive learning capabilities, that are used to discover geographic knowledge of interest to town planners [10]. The system manages vectorized topographic maps of the Apulia region of Italy and supports users in their map interpretation tasks. In this Section, we show an application of automated map interpretation to illustrate how logical descriptions of map cells are used. The problem is locating a "system of farms" in the large territory of his/her interest. This geographical object is not present in the GIS model, thus only the specification of its operational definition will allow the GIS to find cells containing a system of farms in a vectorized map. Who can provide such a definition? Asking the user to do that is not a feasible solution for a number of reasons.

Firstly, providing the system with operational definitions of some environmental concepts is not a trivial task. For example, the general description of a road given by an expert and reported in [11] is "a consolidated way, in the first place used for motor vehicle traffic, including over- and underpasses. Also dividing strips and roadsides … belong to roads." This declarative, abstract definition is difficult to compile into a query on a map repository.

Secondly, the operational definitions of some geographical objects are strongly dependent on the data model that is adopted by the GIS. For instance, finding relationships between the density of vegetation and the climate is easier with a *raster data model*, while determining the main orientation of some morphological elements is simpler in a *topological data model* [5].

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 such as 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 [12]. The idea is that of extending a GIS with a training facility and a learning capability, so that each time a user wants to query the database about some geographical objects that are not explicitly modeled, he/she can prospectively train the system to recognize such objects. Training is based on a set of examples and counterexamples of geographical objects of interest to the user (e.g., road, ravine or steep slope).

Going back to our application, the user will train the system to recognize systems of farms. The only requirement for the GIS user is the ability to detect and mark some cells that are instances of his/her definition of a system of farms. Let us consider the cell shown in Figure 3, which corresponds to a square kilometer of the Apulian region reported on a map scale of 1:25,000. Suppose that the cell has been visually recognized as a system of farms by the GIS user, then it can be considered as an example of a "system of farms". Its logical description is automatically generated by the Map Descriptor (see Figure 3), which operates on the vectorized representation of the cell. The user obtains an operational definition of a "system of farms" by means of

a machine learning tool embedded in INGENS, named ATRE [9]. Briefly, the learning problem solved by ATRE can be formulated as follows:
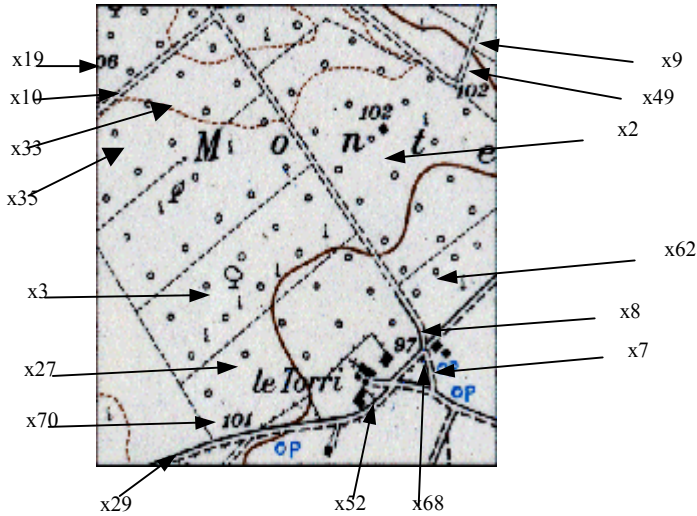

```
class(x1)=system_of_farms ←
  contain(x1,x2), contain(x1,x3), …,contain(x1,x70),
  type_of(x2)=parcel, type_of(x3)=parcel,…,type_of(x70)=quote,
  subtype_of(x2)=cultivation, subtype_of(x3)=cultivation, …,
  subtype_of(x68)=dry_wall, color(x2)=black,…,color(x70)=black,
  altitude(x19)=106.00, …, altitude(x70)=101.00,
  area(x2)=187525.00, area(x3)=99962.5, …, area(x62)=30250.00,
  density(x2)=high, density(x3)=medium, …, density(x62)=low,
  extension(x7)=111.018, extension(x8)=131.37, …,
  extension(x68)=101.119, line_shape(x7)=straight,
  line_shape(x8)=curvilinear, …, line_shape(x68)=straight,
  geo_direction(x7)=north, geo_direction(x10)= north_east, …,
  geo_direction(x68)=north, line_to_line(x7,x68)=almost_parallel, …,
  line_to_line(x52,x68)=almost_perpendicular,
  distance(x7,x68)=5.00,…,distance(x29,x52)=116.00,…,
  line_to_region(x8,x27)=adjacent, …,
  line_to_region(x33,x35)=intersect, region_to_region(x2,x3)=disjoint,
  region_to_region(x2,x9)=disjoint, …,
  region_to_region(x49,x62)=disjoint
```

Fig. 3. (*Above*) A cell containing a system of farms. (*Below*) The partial logical description of the cell. Constant *x1* represents the whole cell, while all other constants denote the sixty-nine enclosed objects. Linear values are expressed in (square) meters

Given
- a set of concepts C_1, C_2, …, C_r to be learned,
- a set of observations O,
- a background knowledge BK,
- 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.

In the context of map interpretation, each C_i is a geographical object not explicitly reported in the map legend, such as a "system of farms". Observations are logic descriptions of the cells, like that in Figure 3, while the background knowledge defines the relevant domain knowledge, such as the following rule for qualitative spatial reasoning:

   close_to(X,Y)=true ← region_to_region(X,Y)=meet
   close_to(X,Y)=true ← close_to(Y,X)=true

which states that two adjacent zones are also close.

From a training set of twenty-nine observations, eight of which refer to the concept system of farms, ATRE induced the following two clauses:
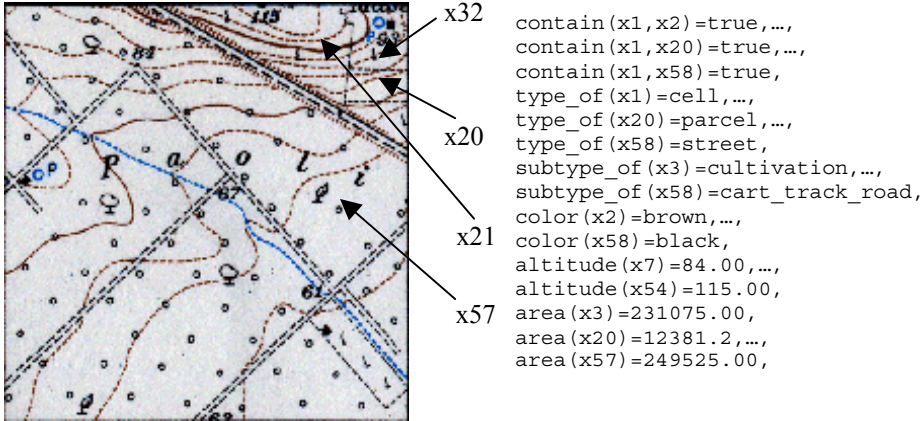
class(S1)=system_of_farms ←
         contain(S1,S2)=true, region_to_region(S2,S3)=meet,
         area(S2)∈[68437.5 .. 187525],
         region_to_region(S2,S4)=disjoint, region_to_region(S4,S3)=meet,
         type_of(S1)=cell, type_of(S2)=parcel, type_of(S4)=parcel,
         type_of(S3)=parcel
class(S1)=system_of_farms ←
         contain(S1,S2)=true, region_to_region(S2,S3)=disjoint,
         density(S3)=high, region_to_region(S2,S4)=meet,
         region_to_region(S4,S5)=meet, region_to_region(S2,S5)=meet,
         type_of(S1)=cell, area(S2)∈[12381.2 .. 25981.2], type_of(S2)=parcel

The first clause states that "there are two pairs of adjacent parcels (S2, S3) and (S4, S3), one of which is relatively large (the area is between 68437.5 and 187525 m^2)". This clause explains six training observations of a "system of farms". The second clause states that "there are three adjacent regions (S2, S4, S5), 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 (S3) with a high density (presumably vegetation), disjoint from the parcel S2". The second clause explains the remaining two training observations of a "system of farms".

This definition is complete and consistent, since it covers all observations classified as a system of farms and no other observation in the training set. In this example, ATRE is asked to generate more than 1200 consistent clauses (though not necessarily complete) before choosing the best. The preference criterion defined for the selection of the best consistent clause maximizes both the number of observations explained by the clause and the number of literals in the body of the clause.

INGENS can now recognize other examples of cells including a system of farms, such as that reported in Figure 4. This is done by matching the learned clauses against

logical descriptions of other map cells. In the example shown in Figure 4, there is only one matching substitution that associates variables of the second clause with constants of the observation, namely S2 with x20, S3 with x57, S4 with x21 and S5 with x32. Generally speaking, INGENS recognizes complex geographical objects that have not been explicitly modeled by the computation of matching substitutions.


```
contain(x1,x2)=true,…,
contain(x1,x20)=true,…,
contain(x1,x58)=true,
type_of(x1)=cell,…,
type_of(x20)=parcel,…,
type_of(x58)=street,
subtype_of(x3)=cultivation,…,
subtype_of(x58)=cart_track_road,
color(x2)=brown,…,
color(x58)=black,
altitude(x7)=84.00,…,
altitude(x54)=115.00,
area(x3)=231075.00,
area(x20)=12381.2,…,
area(x57)=249525.00,
```

```
density(x3)=high,…, density(x57)=high,
extension(x2)=784.618,…, extension(x58)=707.992,
line_shape(x2)=straight,…, line_shape(x58)=curvilinear,
geographic_direction(x2)=north_west,…,
geographic_direction(x43)=north_west,
line_to_line(x2,x4)=almost_perpendicular,
line_to_line(x2,x13)=almost_parallel,…,
line_to_line(x29,x43)=almost_parallel,
distance(x2,x13)=23.00,…, distance(x29,x43)=247.00,
line_to_region(x2,x21)=intersect,…, line_to_region(x53,x57)=intersect,
region_to_region(x3,x24)=disjoint, …, region_to_region(x20,x21)=meet,
region_to_region(x20,x32)=meet, region_to_region(x20,x57)=disjoint,
region_to_region(x21,x32)=meet,…, region_to_region(x50,x57)=disjoint, …,
region_to_region(x50,x57)=disjoint, point_to_region(x5,x50)=outside,…,
point_to_region(x48,x50)=outside.
```

Fig. 4. An example of cell classified as a "system of farms" by matching its logical description against the first learned rule

4   Conclusions and Future Work

Automated map interpretation is a challenging application domain for graphics recognition. Knowledge of the meaning of symbols reported in the map legends is not generally sufficient for recognizing interesting complex geographical objects or patterns on a map. Moreover, it is quite difficult to describe such patterns in a machine-readable format. That would be tantamount to providing GIS with an operational definition of abstract concepts that are often reported in texts and

specialist handbooks. In order to enable the automation of map interpretation tasks in GIS, we propose a novel approach which is based on the generation of logic descriptions of maps and the application of machine learning tools to these descriptions. More specifically, every time a GIS user, such as a geographer, an urban planner, or a geologist, needs to "process" patterns of geographical objects of interest, he/she may generate the corresponding operational definitions by indicating the cell images which are instances of the patterns. This is done by invoking the computational methods for the generation of the logic descriptions of these cells, and then by applying an inductive learning algorithm to these descriptions. The induced rules act as operational definitions, and can be used to recognize new occurrences of the patterns in maps.

The proposed approach still presents some unresolved problems. In the data model for topographic maps, 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 excessively large cells with an unmanageable number of components. To cope with over-segmentation, it is necessary to consider the *context* of a cell, that is the neighboring cells, both in the training and the recognition phase. To solve problems caused by under-segmentation it is crucial to provide users with appropriate tools that hide irrelevant information in the cell description. Indeed, a set of generalization and abstraction operators will be implemented in INGENS to simplify the complex descriptions currently produced by the Map Descriptor module.

Acknowledgments

References

1. Barzohar, M., and Cooper, D.B. (1996). Automatic finding of main roads in aerial images by using geometric-stochastic models and estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7), 707-720
2. den Hartog, J., Holtrop, B. T., de Gunst, M. E., and Oosterbroek, E. P. (2000). Interpretation of Geographic Vector-Data in Practice. In A.K. Chhabra and D. Dori (*eds*.), *Graphics Recognition Recent Advances*, Lecture Notes in Computer Science 1941, Berlin: Springer, 50-57
3. Dupon, F., Deseilligny, M. P., and Gondran, M. (1998). Automatic Interpretation of Scanned Maps: Reconstruction of Contour Lines. In K. Tombre & A.K. Chhabra (Eds.), *Graphics Recognition: Algorithms and Systems*, Lecture Notes in Computer Science 1389, Berlin: Springer, 194-206

4. Esposito, F., Lanza, A., Malerba, D., and Semeraro, G. (1997). Machine learning for map interpretation: an intelligent tool for environmental planning. *Applied Artificial Intelligence*. 11(7-8), 673-695
5. Frank, A. U. (1992). Spatial concepts, geometric data models, and geometric data structures. *Computers & Geosciences*, 18(4), 409-417
6. Gaede V., and Günther O. (1998). Multidimensional Access Methods*, ACM Computing Surveys*, 30(2), 170-231
7. Keates, J. S. (1996). *Map understanding*. Edinburgh:Longman
8. Liow, Y.-T., and Pavlidis, T. (1990). Use of shadows for extracting buildings in aerial images. *Computer Vision, Graphics, and Image Processing*, 49, 242-277
9. Malerba, D., Esposito, F., and Lisi, F. A. (1998). Learning Recursive Theories with ATRE. In *Proceedings of the 13th European Conference on Artificial Intelligence*, ed. H. Prade, 435-439, Chichester (UK): John Wiley & Sons
10. Malerba, D., Esposito, F. , Lanza, A., Lisi, F. A., and Appice, A. (2002). Empowering a GIS with Inductive Learning Capabilities: The case of INGENS. *Journal of Computers, Environment and Urban Systems*, (to appear)
11. 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
12. Mitchell, T. (1997). *Machine learning*. New York: McGraw-Hill
13. Open GIS Consortium (1996). The OpenGIS Abstract Specification. http://www.opengis.org/public/abstract.html
14. Pavlidis, T. (1982). Algorithms for graphics and image processing. Springer, Berlin
15. Rashid, A., Shariff, B. M., Egenhofer, M. J., and Mark, D. M. (1998). Natural language spatial relations between linear and aeral objects: the topology and metric of English Language terms. *International Journal of Geographical Information Science*, 12(3), 215-246
16. Sondheim, M., Gardels, K., and Buehler, K. (1999). GIS Interoperability. In P.A. Longley, M.F. Goodchild, D.J. Maguire, and D.W. Rhinds (eds.), *Geographical Information Systems, Principles and Technical Issues*, Volume 1, New York: John Wiley & Sons, 347-358
17. Yamada, H., Yamamoto, K., and Hosokawa, K. (1993). Directional Mathematical Morphology and Reformalized Hough Transformation for the Analysis of Topographic Maps, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4), 380-387