Journal of Universal Computer Science, vol. 8, no. 8 (2002), 751-773
submitted: 11/2/02, accepted: 29/4/02, appeared: 28/8/02

The Design of an Object-based System for Representing and Classifying Spatial Structures and Relations

Florence Le Ber
(LORIA UMR 7503, BP 239, 54506 Vandœuvre-lès-Nancy, France
leber@loria.fr)

Amedeo Napoli
(LORIA UMR 7503, BP 239, 54506 Vandœuvre-lès-Nancy, France
napoli@loria.fr)

Abstract: Our work is concerned with the design of a knowledge-based system for recognizing agricultural landscape models on land-use maps. Landscape models are defined as sets of spatial structures and spatial relations. This paper focuses on the representation of topological relations inside an object-based representation system. In this system, relations are represented by objects with their own properties. We propose to define two types of properties: the first ones are concerned with relations as concepts while the second are concerned with relations as links between concepts. In order to represent the second type of properties, we have defined facets that are inspired from the constructors of description logics. We describe these facets and how they are used for classifying spatial structures and relations on land-use maps. The paper ends with a discussion on the present work and related work in qualitative spatial reasoning.
Key Words: Object-based knowledge representation, topological relations, classification, relation reification, landscape analysis.
Category: I.2.1 I.2.4

1 Introduction

Today, agronomic researchers utilize land-use maps based on satellite images for analyzing agricultural systems and forecasting environmental problems. Their analyses rely on the spatial organizations of the various land-use categories (crops, meadows, forests, villages) that are assumed to reveal the functioning of the farming systems. Models of these organizations, called *landscape models*, have been defined and later formalized for classifying zones on land-use maps [3, 11, 20]. Landscape models are described by a list of agricultural spatial structures and a set of qualitative spatial relations between these structures. Besides, the zones on the land-use maps correspond to so-called "village territories" that are assumed to be exploited by the farmers of a village. A zone is made of image regions, i.e., connected sets of pixels with the same label, the label representing the land-use category [2].

In our framework, the problem of recognizing spatial organizations can be considered as an instance classification problem, where landscape models correspond to classes and image zones correspond to instances. The representation of landscape models depends on agricultural knowledge and on spatial relations. Moreover, methods for extracting regions from an image and for checking spatial relations on an image are needed. Finally, a classification method has to be defined for comparing the regions and relations between regions on an image with the structures and the relations described in the landscape models.

In our application, classification is mainly based on the spatial relations, since they are the most characteristic elements of the landscape models. Thus, we have focused on the representation of the properties of the relations: our goal is both to minimize calculations for checking relations on an image and to enhance reasoning on spatial relations. In particular, we have chosen a hierarchical representation of the relations allowing the factorization of properties and calculation methods.

We have restricted our study to the topological relations because they can be described and manipulated on the basis of a well defined theoretical framework [36]. Within this framework, we need a system for representing topological relations that:

- defines primitives for computing relations on raster images[1]: to compute –or to check– a relation R between two objects x and y means to test if $R(x, y)$ holds,
- integrates all the relations used in the landscape models,
- allows one to store new information,
- allows reasoning on relations: to reason on relations means to deduce new relations from already computed ones.

According to these needs, we have chosen to use an object-based knowledge representation system (or ObKR system), that allows both programming and knowledge representation, and that includes a classification mechanism [26]. We have extended the representation capabilities of the ObKR system in order to represent spatial relations as "first-class citizens", i.e., objects with their own properties. In our proposition, relations are represented by classes, having attributes and facets; they are organized within a hierarchy. The classification mechanism in the ObKR system has been modified accordingly to take reified relations into account. These representation and classification mechanisms have been developed for a particular purpose, namely classification of landscape spatial structures, but they are of general interest and they can be reused in other application contexts needing reified relations. Furthermore our proposition can

[1] "Raster" images are, in opposition to "vector" images, made of points, called *pixels*, that are characterized by their position (line, column) and their label (color or level of gray).

be taken as a general basis for representing and manipulating relations in an object-based knowledge representation system.

The paper is organized as follows. Section 2 describes the domain of our application, i.e., landscape analysis, while Section 3 describes the relations we have used, namely topological relations. Section 4 presents the ObKR system we have used and the way spatial structures have been represented in this system. Section 5 focuses on relation reification. Section 6 describes the functioning of the system and Section 7 proposes a discussion on the present work and related work. Finally we conclude and present some perspectives of our work.

2 Agricultural landscape analysis

We work on raster images made from LANDSAT TM satellite data of the Lorraine region (east of France). These images are composed of labeled regions, each label corresponding to a particular land-use category, e.g. forest, meadow, corn, barley, buildings. A region is a set of connected pixels, where a pixel represents $90m^2$. Figure 1 represents a village territory extracted from an image: the territory is bordered by forests (dark grey); it contains a village (white) and is mainly covered with pastures, meadows and a few crops (middle grey). Black zones correspond to unrecognized land use; the outside of the village territory is also displayed in black [2, 19].

The landscape models to be recognized represent the spatial organization of village territories. There are five main landscape models that are named according to the Lorraine main reliefs: *valley*, *up-coast*, *down-coast*, *plain*, *plateau* [11, 20]. Each model is described as a set of spatial structures connected with spatial relations. For instance the *valley model* (see Figure 2) is described as follows: *"the village territory is bordered by two forests; grasslands (pastures and hay meadows) cover the major part of the territory; they surround the village while crop fields are small and near the forests"*. Thus, different sorts of elements have to be checked on the images: atomic spatial structures (a crop field, a meadow, a forest), complex spatial structures (a village territory, a group of fields), qualitative spatial relations (bordered, cover, near, etc.), and some features of the image regions: areas, forms, etc.

During the knowledge acquisition process, we and the agronomists have defined a lexicon for the landscape models [20]: each model is then expressed as a set of statements based on this lexicon elements; each element of the lexicon is associated to a checking method on the image. For example, a landscape model can be described in the following way:

- *"all crop fields belonging to the territory are large fields"*,
- *"all state forests are disconnected from the territory"*,
- *"the village is externally connected with at least one meadow"*,

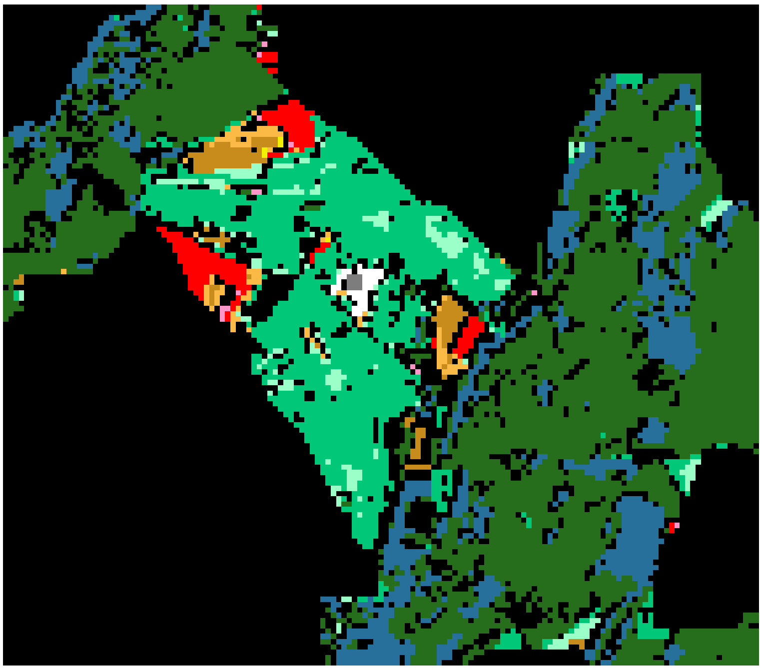


Figure 1: A village territory.

– *"the territory tangentially contains at most one group of fields"*.

The lexicon elements *crop field*, *large field*, *state forest*, represent atomic spatial structures that can be directly checked on the image. *Belonging*, *disconnected*, etc., are spatial relations quantified with *all*, *at least*, *at most*. The lexicon element *group of fields* represents a spatial structure composed of a set of connected fields.

In the following, we show how we represent these lexicon elements (spatial structures, spatial relations and quantifiers) in order to recognize the landscape models on the land-use maps.

3 Topological relations

As introduced in the previous section, the landscape models are described with qualitative spatial relations. These relations are of three types: orientation, distance and topology. We have focused on topological relations since they are binary relations well formalized in logical frameworks [9, 30, 35]. By contrast, orientation and distance relations are ternary relations (they need frames or reference objects) and their formalization is still incomplete, making them more

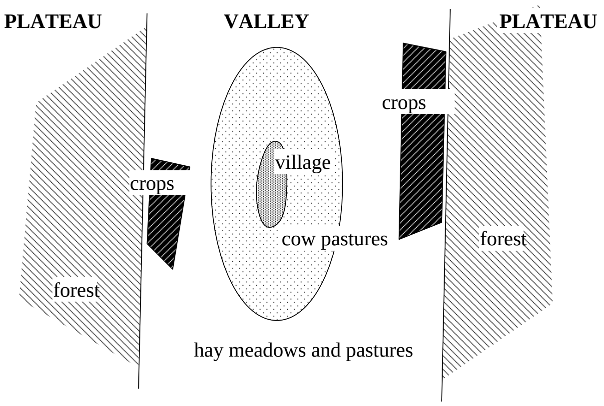


Figure 2: The valley model.

difficult to represent and to use in an ObKR system [7, 17]. Finally, topological relations are a priori sufficient to describe the main characteristics of the landscape models [20, 25].

We rely on the **RCC-8** theory [8, 30], that is based on the connection relation: two objects are connected if they share at least a point. There are eight basic relations whose names and iconic representations are given in Table 1.

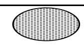
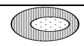
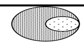
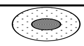
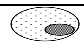
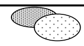
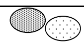
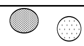
Relation	Notation	Icons
"x is identical with y"	$EQ(x, y)$	
"x is a non tangential proper part of y"	$NTPP(x, y)$	
"x is a tangential proper part of y"	$TPP(x, y)$	
"x non tangentially contains y"	$NTPP^{-1}(x, y)$	
"x tangentially contains y"	$TPP^{-1}(x, y)$	
"x partially overlaps y"	$PO(x, y)$	
"x is externally connected with à y"	$EC(x, y)$	
"x is disconnected from y"	$DC(x, y)$	

Table 1: Names and icons associated to the eight base relations of **RCC-8**.

In our framework, topological relations are computed on the images thanks to set operations as it is done in [15]. In our application, $\varDelta$ denotes the set of the regions of an image, that are regular, with no hole, closed, and not necessarily connected. A region $x \in \varDelta$ is made of two sets, the interior (x°) and the boundary (∂x) (see [21] for details about the definition of interiors and boundaries for raster regions). We consider four operations between two regions x and y: intersection of the interiors, $x^\circ \cap y^\circ$, intersection of the boundaries, $\partial x \cap \partial y$, differences of the interiors, $x^\circ - y^\circ$, $y^\circ - x^\circ$. The result of these operations may be empty or not empty. We have accordingly defined eight conditions denoted as follows:

- $P(x, y)$, "x is a part of y": $x^\circ - y^\circ = \emptyset$
- $Dx(x, y)$, "x is not a part of y": $x^\circ - y^\circ \neq \emptyset$
- $P^{-1}(x, y)$, "x contains y": $y^\circ - x^\circ = \emptyset$
- $Dy(x, y)$, "x does not contain y": $y^\circ - x^\circ \neq \emptyset$ [2]
- $O(x, y)$, "x overlaps y": $x^\circ \cap y^\circ \neq \emptyset$
- $DR(x, y)$, "x is discrete from y": $x^\circ \cap y^\circ = \emptyset$
- $A(x, y)$, "x shares a boundary with y": $\partial x \cap \partial y \neq \emptyset$
- $NA(x, y)$, "x does not share any boundary with y": $\partial x \cap \partial y = \emptyset$

The conjunctions of these eight conditions are equivalent to the eight base relations of **RCC-8** (see Table. 2). Then, computing a relation on the image is the same operation as verifying a set of conditions. For example, the relation "x is externally connected with y", $EC(x, y)$, is associated with the set of conditions:

$$\mathcal{C}(EC) = \{Dx, Dy, DR, A\} = \{x^\circ - y^\circ \neq \emptyset, y^\circ - x^\circ \neq \emptyset, x^\circ \cap y^\circ = \emptyset, \partial x \cap \partial y \neq \emptyset\}$$

Relying on this connection between the **RCC-8** base relations and the eight conditions, a Galois lattice has been defined that contains 34 elements (Figure 3). An element `E` of the lattice is an ordered pair $(\mathcal{C}, \mathcal{R})$, where $\mathcal{C}$ is a subset of conditions and $\mathcal{R}$ is a subset of base relations such that:

$$\forall (x, y) \in \varDelta^2, \bigvee_{r \in \mathcal{R}} r(x, y) \leftrightarrow \bigwedge_{c \in \mathcal{C}} c(x, y)$$

Thus, each relation that is represented in the lattice can be checked on the image using conditions. For instance the relation PP is checked using the three conditions (P, O, Dy). Furthermore the partial ordering $\sqsubseteq$ in the lattice is equivalent to the logical implication on the relations:

$$(\mathcal{C}_1, \mathcal{R}_1) \sqsubseteq (\mathcal{C}_2, \mathcal{R}_2) \quad \leftrightarrow \quad \forall (x, y) \in \varDelta^2, \bigvee_{r \in \mathcal{R}_1} r(x, y) \rightarrow \bigvee_{r \in \mathcal{R}_2} r(x, y)$$

This lattice structure provides interesting properties for reasoning purposes. Any pair of elements of the lattice has a greatest lower bound, or *glb* (denoted by

[2] Actually $Dy = Dx^{-1}$ and the two conditions could be renamed according to this statement.

$$
\begin{aligned}
EQ(x,y) &\leftrightarrow (P \wedge P^{-1} \wedge O \wedge A)(x,y)\\
NTPP(x,y) &\leftrightarrow (P \wedge Dy \wedge O \wedge NA)(x,y)\\
TPP(x,y) &\leftrightarrow (P \wedge Dy \wedge O \wedge A)(x,y)\\
NTPP^{-1}(x,y) &\leftrightarrow (Dx \wedge P^{-1} \wedge O \wedge NA)(x,y)\\
TPP^{-1}(x,y) &\leftrightarrow (Dx \wedge P^{-1} \wedge O \wedge A)(x,y)\\
PO(x,y) &\leftrightarrow (Dx \wedge Dy \wedge O \wedge A)(x,y)\\
EC(x,y) &\leftrightarrow (Dx \wedge Dy \wedge DR \wedge A)(x,y)\\
DC(x,y) &\leftrightarrow (Dx \wedge Dy \wedge DR \wedge NA)(x,y)
\end{aligned}
$$

Table 2: Computing the **RCC-8** relations: each relation is associated with a conjunction of conditions.

$\frown$), and a least upper bound or *lub* (denoted by $\smile$). The *glb* of two elements is equivalent to the conjunction of these two elements. By contrast, this equivalence is not true for the *lub*, and the disjunction of two elements only implicates the *lub* of the two elements [21, 22]. This fact has to be linked to a property of closed sets: the *lub* of two closed sets is generally not a closed set, whereas the *glb* of two closed sets is a closed set [10].

As a consequence, this lattice is ordered and closed under conjunction but is not closed under disjunction. Considering for example the $TP(x,y)$ and $TP^{-1}(x,y)$ relations (on the left of the lattice, Figure 3) we have:

$TP \frown TP^{-1} = EQ$ and $TP \smile TP^{-1} = OetA$

$\forall(x,y),\ EQ(x,y) \equiv TP(x,y) \wedge TP^{-1}(x,y)$

$\forall(x,y),\ TP(x,y) \vee TP^{-1}(x,y) \rightarrow OetA(x,y)$

$\forall(x,y),\ OetA(x,y) \rightarrow (TP \vee TP^{-1} \vee OetAetDy \vee OetAetDx)(x,y)$

Moreover, the composition of pair of relations can be deduced from the Galois lattice and from the composition table of the base relations as it is done in [29]. For instance, if the relations $P(x,y)$ and $EC(y,z)$ hold for the three regions x, y and z, it can be inferred that $DR(x,z)$ holds (DR is the *lub* of the pair (EC, DC), see Figure 3):

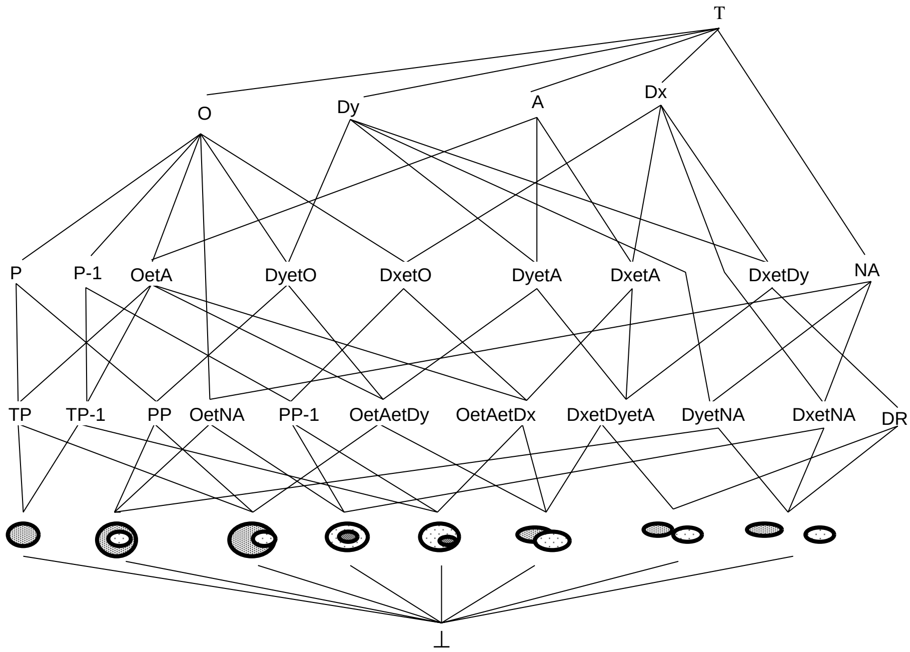


Figure 3: A Galois lattice of topological relations: the eight **RCC-8** relations are at the bottom of the lattice (icons) [25].

$$
\begin{aligned}
\text{from the lattice: } & P(x,y) = TPP(x,y) \vee NTTP(x,y) \vee EQ(x,y) \\
\text{from the table: } & TPP(x,y) \circ EC(y,z) = DC(x,z) \vee EC(x,z) \\
\text{and: } & NTPP(x,y) \circ EC(y,z) = DC(x,z) \\
\text{and: } & EQ(x,y) \circ EC(y,z) = EC(x,z) \\
\text{thus: } & P(x,y) \circ EC(y,z) = DC(x,z) \vee EC(x,z) = DR(x,z)
\end{aligned}
$$

Due to the characteristic of the *lub*, the composition of two relations is unfortunately not always a relation of the Galois lattice. Thus, this lattice is not closed under composition. A complete analysis of the properties of the Galois lattice and a comparison with other lattices is given in [21, 22].

4 The representation framework

We want to build a system for helping agronomists to recognize landscape models on images. This system has to contain: the models to be recognized (i.e., the landscape models); the elements (i.e., structures, relations, quantifiers) that compose these models; a method to match images and models, i.e. a classification method. According to these needs, we have chosen to use the Y3 ObKR system. In this section, we first describe the characteristics of Y3 and then the representation of spatial structures.

4.1 The Y3 system

He Y3 system is an ObKR system based on a frame language called YAFOOL and a graphical interface called YAFEN [13]. In YAFOOL, all objects (classes and instances) are represented by *frames*; frames are composed of *slots*, representing both attributes and methods. Attributes can be characterized by declarative and procedural *facets*: the former are used to represent the range and the value of the attributes while the latter are used to specify local behaviors. Attributes, facets, and methods are objects. Binary relations are special kinds of attributes which are characterized by the fact that their range is a user-defined class. Relations are specializations of the special class `RELATION`. A relation may have an inverse relation, and the system is in charge of managing their interrelated values. The classification and inheritance mechanisms are based on attribute unification. When classifying an object into a class, the systems checks whether the (attribute, range) or (attribute, value) pairs in the object are conform to the pairs in the reference class; if this is the case, the object can be classified as an instance of the reference class.

ObKR systems share many characteristics with description logics. In description logics, *concepts* are used to represent classes of individuals while *roles* are used to represent relations between classes [12, 27]. The description of a concept is composed of roles introduced by *constructors* expressing restrictions on the role, e.g. range of the role, cardinality, universal and existential quantification. Reasoning is based on subsumption, concept classification and satisfiability. In particular, concept classification is used to insert a new defined concept in the concept hierarchy by searching for its most specific subsumers and its most general subsumees. The search in the hierarchy is usually performed top-down and depth-first. The classification process in the Y3 system is based on the same principles.

4.2 Representing spatial structures

Landscape models are described as sets of spatial structures connected with spatial relations. In this section, we focus on the representation of spatial structures while the representation of topological relations is detailed in Section 5. It must yet be noticed that the elements of the Galois lattice, i.e., topological relations, are represented by classes (e.g. `DC`, `EC`, `PP`) that are specializations of the Y3 `RELATION` class, according to the lattice ordering.

As explained in Section 2, spatial structures can be atomic or complex. The atoms correspond to the regions of the land-use maps, i.e. raster images, and are recognized on the images using the label of the regions and other indices (surface, form, etc.). Atoms are represented by classes, e.g. FIELD, BARLEY-FIELD, FOREST, organized within a hierarchy. For example, the class LARGE-FIELD is a specialization of the class FIELD: it represents fields having a surface greater than $5000\,m^2$, as illustrated in Figure 4. Each class includes a recognition method (called recognize, see Figure 4), and whenever an atom is recognized on an image, the corresponding class is instantiated.

```
(defclass LARGE-FIELD
  (is-a FIELD)
  (recognize (method ()
    (subcar (lambda (a-region)
      (if (>= (:: surface-region a-region) 50)
         (put-recognize LARGE-FIELD a-region)))
      (li-recognized FIELD))
   )) )

(defclass SMALL-FIELD
  (is-a FIELD)
  (recognize (method ()
    (subcar (lambda (a-region)
      (if (< (:: surface-region a-region) 50)
        (put-recognize SMALL-FIELD a-region)))
      (li-recognized FIELD))
   )) )
```

Figure 4: Classes of atomic spatial structures coded in Y3: large and small fields are classified according to their area.

The complex spatial structures are nested structures. They rely on atomic spatial structures connected with spatial relations. They are represented by classes whose attributes are particular relations, instances of the topological relation classes. A hierarchy of classes has been defined this way, including the landscape models (VALLEY, PLATEAU, etc.) and other complex spatial structures: groups of crop fields (GROUP, GROUP-EC-FOREST) or village territories (TERRITORY, TERRITORY-EC-FOREST, TERRITORY-PP^{-1}-GROUP, etc.). All these classes are linked through the relations classes (see Figure 5).

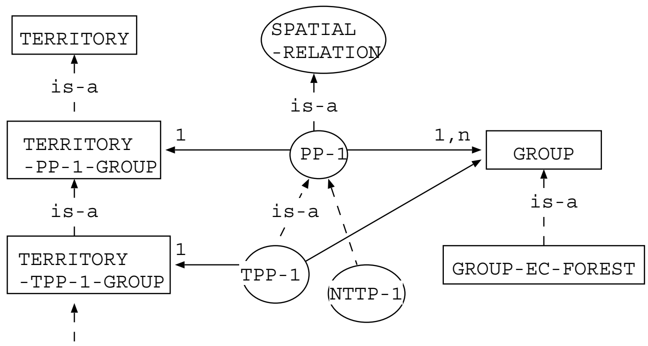


Figure 5: Linking the classes of spatial structures (`TERRITORY` and `GROUP`) through the classes of relations. A *group* denotes a set of connected fields; a *territory* denotes a village territory.

The classes representing atomic and complex spatial structures can be specialized in four different ways:

- Adding an attribute: for example, the `FIELD` class is specialized into the `SMALL-FIELD` class or in `CORN-FIELD`, `BARLEY-FIELD`, etc.
- Adding a relation: for example, the `TERRITORY` class is specialized into `TERRITORY-EC-FOREST` or `TERRITORY-DC-FOREST`.
- Specializing a relation: for example, the TERRITORY-PP^{-1}-GROUP class can be specialized into TERRITORY-TPP^{-1}-GROUP.
- Specializing the range of a relation: for example, the TERRITORY-PP^{-1}-GROUP class is specialized into TERRITORY-PP^{-1}-GROUP-EC-FOREST.

The recognition of a complex spatial structure `s` is based on the classification of all the structures and relations composing `s`. Like the classes representing atomic structures, each class representing a complex spatial structure, say `C`, includes a main *classification method*, called `recognize`. The role of the method `recognize` is to check whether a region or a set of regions of an image can be classified into the class `C`. If the test succeeds, an instance of `C` is created (more details are given in Section 6).

5 The representation of relations

Up to now there is no satisfying reification of relations in object-based representation systems [24, 31]. Moreover, the *relation* concept can be considered as opposed to the *object* concept. The specialization and instantiation relations are commonly taken into account, as well as composition in some ObKR systems, whereas many other relations, such as spatial or temporal relations, may be useful in many applications, but are not handled in a satisfying and convenient way [1, 4, 32].

5.1 The generic class SPATIAL-RELATION

A generic class, named SPATIAL-RELATION, introduces the attributes and the methods common to all classes representing topological relations (see Figure 6). Every class representing a topological relation is a specialization of this generic class and inherits its properties. For instance the classes EC, PP^{-1}, DC, respectively representing the relations $EC(x, y)$, $PP^{-1}(x, y)$, $DC(x, y)$, are subclasses of SPATIAL-RELATION.

```
(defclass SPATIAL-RELATION
    (is-a RELATION)
    (complement (a . SPATIAL-RELATION))
    (converse (a . SPATIAL-RELATION))
    (incompatible
       (method (RS) (not (pgcd frame* RS)) ))
    (specialize
       (method (RS) ... ))
    (local-condition (a . CONDITION))
    (search-conditions
       (method ()  (let ())...) ))
    (verify-relation
       (method (O1 O2) ... )))
```

Figure 6: The SPATIAL-RELATION class coded in Y3.

The generic class SPATIAL-RELATION introduces three main methods (in the sense of ObKR systems) which are inherited by each relation class. The verify-relation method checks whether the relation (e.g., EC) exists between two regions of an image. It uses the search-conditions method that returns the set of conditions associated with the relation (e.g. $\mathcal{C}(EC) = \{Dx, Dy, DR, A\}$). If the verify-relation method succeeds (e.g. all conditions of $\mathcal{C}(EC)$ are true), it creates an instance of the relation class. If it fails (e.g. one of the conditions of $\mathcal{C}(EC)$ is false, say A), it searches which relation is associated with the set of conditions it has computed and creates accordingly an instance of this relation class (e.g. $\{Dx, Dy, DR, NA\} = \mathcal{C}(DC)$). The method incompatible checks whether two relations are compatible ($R_1 \frown R_2 \neq \bot$). These three methods use the lattice structure of the relation classes to find out:

- the set of conditions associated with a relation,
- the relation associated with a set of conditions,
- the *glb* of two relations.

The attributes of the class `SPATIAL-RELATION` mainly describe relations between the relations: the value of the `complement` attribute of a relation `R1` is the relation `R2` that is false whenever `R1` is true (and reciprocally):

$$\forall(x, y) \in \Delta^2, R_1(x, y) \leftrightarrow \neg R_2(x, y)$$

The `converse` attribute of a relation `R1` gives the relation `R2` that is true for (y, x) whenever `R1` is true for (x, y) (and reciprocally):

$$\forall(x, y) \in \Delta^2, R_1(x, y) \leftrightarrow R_2(y, x)$$

The value of the attribute `local-condition` is the set of conditions equivalent to the relation.

5.2 Quantifiers and facets

As introduced in Section 2, the agronomists describe the relations between the image regions with various quantifiers such as: *all, none, at least, at most*, etc. Some of these quantifiers are basic constructors in description logics. We describe below the four main examples of quantifiers used in our application, their correspondence in description logics and how we represent them in Y3.

- *"all crop fields belonging to x are large fields"*.
  In predicate calculus: $\forall y,\ \text{crop-field}(y) \wedge \text{contain}(x, y) \rightarrow \text{large-field}(y)$
  In description logics and in Y3:
  ```
  x = (all (range contain crop-field) large-field)
  ```
- *"all state forests are disconnected from x"*.
  In predicate calculus: $\forall y,\ \text{state-forest}(y) \rightarrow \text{disconnected}(x, y)$
  This implication is different from the previous one since its conclusion involves the relation "disconnected". Thus the `all` constructor, as it is used in description logics, i.e. `(all R C)`, cannot be directly used to represent this sentence. An alternative is to use the contraposition $\neg B \rightarrow \neg A$ rather than the original implication $A \rightarrow B$. The contraposition can be represented in Y3 as in description logics:
  ```
  x = (all (not disconnect)(not state-forest))
  ```
  Note that this construction implies the use of the negation on roles (not commonly used in description logics [23]). We have therefore defined and implemented a specific facet, called `all-role`, described below.

– *"x is externally connected with at least one meadow"*.
  This sentence cannot be easily represented in predicate calculus, unless using a specific construction to express the cardinality restriction. This sentence is represented in Y3 as in description logics using the generalized numerical restriction `at-least`:
  `x = (at-least 1 externally-connected meadow)`
– *"there is at most one group which is a tangential part of x"*.
  The same remark applies for predicate calculus: this cardinality restriction must be handled in a specific way. This sentence is represented in Y3 as in description logics using the generalized numerical restriction `at-most`.
  `x = (at-most 1 tangentially-contain group)`

The quantifiers *all*, *at least*, *at most* are not available in the original version of Y3. Thus, relying on the models of the corresponding constructors in description logics, we have implemented facets representing these quantifiers, that can be associated with relation classes. According to the four sentences above, we have designed four facets, namely `super-range`, `all-role`, `c-atleast`, and `c-atmost`. Furthermore, the facet `c-exactly` is the conjunction of the two facets `c-atleast` and `c-atmost`. Examples of the use of these facets for representing spatial structures are given in Figure 7:

– the `GROUP-EC-FOREST` class is defined as a `GROUP` that is externally connected to exactly one `FOREST`,
– the instances of `TERRITORY-PP`$^{-1}$`-GROUP` are instances of `TERRITORY` that contain at least an instance of `GROUP`,
– the instances of `TERRITORY-TPP`$^{-1}$`-GROUP` are instances of `TERRITORY-PP`$^{-1}$`-GROUP` where at least one group is a tangentially proper part of the territory,
– the `TERRITORY-PP`$^{-1}$`-GROUP-EC-FOREST` class is a specialization of the class `TERRITORY-PP`$^{-1}$`-GROUP` where all instances of `GROUP` are instances of the class `GROUP-EC-FOREST`,
– the `TERRITORY-DC-FOREST` class is defined as a `TERRITORY` that is disconnected from all `FOREST`.

One can note that other facets corresponding to other specific needs could be defined as well.

5.3 Interpretation and complexity

As in description logics frameworks, we call an *interpretation* the pair $\mathcal{I} = (\Delta_\mathcal{I}, .^\mathcal{I})$ where $\Delta_\mathcal{I}$ is a set of objects, called the *interpretation domain*, and $.^\mathcal{I}$ is an interpretation function, mapping a class to a subset of $\Delta_\mathcal{I}$ and an attribute to a subset of $\Delta_\mathcal{I} \times \Delta_\mathcal{I}$. The interpretation of a class relies on the interpretation of its attributes. Methods are not taken into account since they have no side-effects: actually they are comparable to the test functions of Classic [5]. Furthermore, the interpretation of an attribute `p` depends on its facets. In the following, we

```
(defclass GROUP-EC-FOREST
    (is-a GROUP)
    (g-ec-f (c-exactly 1 EC FOREST)))

(defclass TERRITORY-PP^-1-GROUP
    (is-a TERRITORY)
    (t-ppi-g (c-atleast 1 PP^-1 GROUP)))

(defclass TERRITORY-TPP^-1-GROUP
    (is-a TERRITORY-PP^-1-GROUP)
    (t-tppi-g (c-atleast 1 TPP^-1 GROUP)))

(defclass TERRITORY-PP^-1-GROUP-EC-FOREST
    (is-a TERRITORY-PP^-1-GROUP)
    (t-ppi-gf (super-range PP^-1 GROUP GROUP-EC-FOREST)))

(defclass TERRITORY-DC-FOREST
    (is-a TERRITORY)
    (t-dc-f (all-role DC FOREST)))
```

Figure 7: Using topological relations and facets to represent complex spatial structures. The attributes (e.g. `t-dc-f`) are instances of the relation classes (e.g. `DC`). They are characterized by facets representing quantifiers (e.g. `all-role`).

give the interpretation of the four facets introduced above for representing spatial structures, where `p` is an instance of the `R` relation class, `CD` is the range of `p`, and `SUP-CD` is more general than `CD`:

$$(\texttt{c-atleast}\ n\ \texttt{R CD})^{\mathcal{I}} =$$
$$\{x \in \Delta_{\mathcal{I}} | card\{y \in \Delta_{\mathcal{I}} | (x,y) \in \texttt{R}^{\mathcal{I}} \wedge y \in \texttt{CD}^{\mathcal{I}}\} \geq n\}$$
$$(\texttt{c-atmost}\ n\ \texttt{R CD})^{\mathcal{I}} =$$
$$\{x \in \Delta_{\mathcal{I}} | card\{y \in \Delta_{\mathcal{I}} | (x,y) \in \texttt{R}^{\mathcal{I}} \wedge y \in \texttt{CD}^{\mathcal{I}}\} \leq n\}$$
$$(\texttt{c-exactly}\ n\ \texttt{R CD})^{\mathcal{I}} =$$
$$\{x \in \Delta_{\mathcal{I}} | card\{y \in \Delta_{\mathcal{I}} | (x,y) \in \texttt{R}^{\mathcal{I}} \wedge y \in \texttt{CD}^{\mathcal{I}}\} = n\}$$
$$(\texttt{all-role R CD})^{\mathcal{I}} =$$
$$\{x \in \Delta_{\mathcal{I}} | \forall y \in \Delta_{\mathcal{I}}, y \in \texttt{CD}^{\mathcal{I}} \Rightarrow (x,y) \in \texttt{R}^{\mathcal{I}}\}$$
$$(\texttt{super-range R SUP-CD CD})^{\mathcal{I}} =$$
$$\{x \in \Delta_{\mathcal{I}} | \forall y \in \Delta_{\mathcal{I}}, (y \in \texttt{SUP-CD}^{\mathcal{I}} \wedge (x,y) \in \texttt{R}^{\mathcal{I}}) \Rightarrow y \in \texttt{CD}^{\mathcal{I}}\}$$

It is interesting to compare this extension of Y3, including the four facets, with the description logics family $\mathcal{AL}$ [12]. The facets `c-atleast`, `c-atmost`, and super-range require the constructor range that itself requires the negation and the disjunction of defined concepts [33]. Thus, the complexity of the extension of Y3 is similar to that of $\mathcal{ALCN}$: detecting the relations of subsumption is PSPACE-complete. Furthermore, the facet all-role requires the negation of roles. This last constructor is generally not used in description logics, but it is described in the Krss norm [28] and appears as a "difference between roles" in [6]. In [23] it is proved that the extension of $\mathcal{AL}$ with role negation makes the satisfiability of concepts ExpTime-complete. Finally, the complexity of reasoning in this extension of Y3 relies on the same results and is comparable to the current versions of description logics such as Race [16] or Shiq [18].

6 A system for agricultural landscape analysis

The classes representing the spatial structures are organized into a lattice hierarchy, ordered with a subsumption relation based on the specialization mechanisms described in Section 4. Figure 8 shows a part of this hierarchy: the name of a class is composed of the name of its immediate subsumers, except for the classes representing the original landscape models described in section 2 (e.g. PLATEAU). Names are in French and are simplified for the sake of readability. For example finage stands for TERRITORY, f-ilot stands for TERRITORY-PP^{-1}-GROUP, f-ecdom stands for TERRITORY-EC-FOREST, etc.

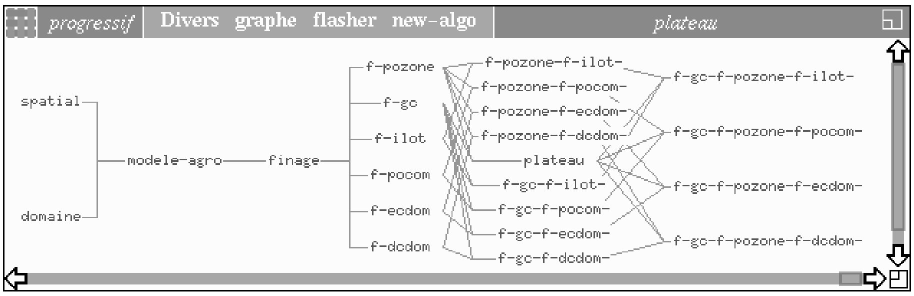


Figure 8: A part of the class hierarchy of spatial structures.

Finally both relations and spatial structures are organized within lattice hierarchies. Thanks to these particular organizations, the classification of an image region is univocal (an instance representing an image region belongs to only one class). Furthermore, the work of the system relies on the two following principles: *i)* checking instance properties is progressive and guided by the class hierarchy; *ii)* the properties are necessary and sufficient conditions for the classification of an instance into a class. These principles constrain the recognition process, as detailed below.

6.1 The recognition process

The purpose of our system is to recognize landscape models, i.e., to classify village territories according to these models. A village territory is checked on an image thanks to its label. It is represented by an instance of the class `TERRITORY`. The recognition principle is to classify an instance from a general class into a more specialized class, traversing downwards the hierarchy. At each step, certain characteristics of the instance are checked. These characteristics can be linked to other image regions. Actually, the classification of an image region `x` requires the classification of the regions which are related to `x` and the classification of the relations linking these regions to the region `x`.

For example, let us suppose that the system tries to classify an instance `t1` of the `TERRITORY` class into the `TERRITORY-DC-FOREST` class (see Figure 7). Accordingly, it uses the `recognize` method of this last class. The `recognize` method works as follows. It verifies that the instance `t1` matches the properties of `TERRITORY-DC-FOREST`, i.e., `t-dc-f` = (`all-role` `DC` `FOREST`). Thus, it looks for forests in the image neighborhood of the territory (calling the `recognize` method of the class `FOREST`) and computes the topological relation between each forest `fi` and the territory `t1` (calling in turn the method `verify-relation`(`DC`, `t1`, `fi`)). If this last method succeeds, the `t-dc-f` property is verified and the instance `t1` is finally classified into the `TERRITORY-DC-FOREST` class. If it fails, the system tries to classify `t1` into another subclass of `TERRITORY` (e.g. `TERRITORY-EC-FOREST`). The classification process goes on downwards the hierarchy of spatial structures until all potential classes have been checked. The general method `recognize` (`C`, `i`), where `C` is a spatial structure class, and `i` is an instance representing an image region, is described in Algorithm 1.

6.2 An example of region classification

At the end of the recognition process, each territory `t` is characterized by a set $\mathcal{E}$ of the classes whose properties are verified by `t`. Finally the village territory `t` belongs to the class `GCD`, the *glb* of $\mathcal{E}$. There are three possibilities:

- `GCD` represents one of the original landscape models. The village territory `t` is recognized as a territory belonging to this model.
- `GCD` is subsumed by one or several landscape models. The classification process has produced more information than it is necessary for the recognition. The village territory `t` is recognized as a territory belonging to the models, with some additional characteristics that can be further interpreted by the agronomists.

Algorithm 1 `recognize` (C, i)

for all P $\in$ C's properties **do**
  R = class of P   %%% *R is a relation class*
  CD = range of P   %%% *CD is a spatial structure class*
  RES = true
  if `all-role` is defined **then**
    for all $i_{CD} \in \Delta_{\mathcal{I}}$ **do**
      if `recognize`(CD, i_{CD}) **then**
        RES = RES and `verify-relation`(R, i, i_{CD})
  if `super-range` is defined **then**
    CDR = super range of P
    for all $i_{CDR} \in \Delta_{\mathcal{I}}$ **do**
      if `verify-relation`(R, i, i_{CDR}) and `recognize`(CDR, i_{CDR}) **then**
        RES = RES and `recognize`(CD, i_{CDR})
  if `c-atleast` or `c-atmost` are defined **then**
    Number = 0
    for all $i_{CD} \in \Delta_{\mathcal{I}}$ **do**
      if `verify-relation`(R, i, i_{CD}) and `recognize`(CD, i_{CD}) **then**
        Number = Number + 1
    if `c-atleast` is defined **then**
      RES = (Number >= (`c-atleast` facet's value))
    if `c-atmost` is defined **then**
      RES = RES and (Number <= (`c-atmost` facet's value))
return RES

– `GCD` subsumes one or several landscape models. No model is recognized. The village territory `t` is characterized by `GCD`.

Figure 9 shows the result of the classification of the village territory displayed in Figure 1. Data have been simplified: only the regions used for the classification are annotated. The territory verifies the properties of the six classes that are enumerated at the bottom of the figure: `F-ILOT`, `F-ILOT-LIM`, `F-ILOT-EC-DOM`, `F-PO-COM`, `F-EC-DOM`, `F-ENTRE-DOM`. The system concludes that this territory belongs to the *valley* model. Actually the class `VALLEY` is the *glb* of the pair `F-ENTRE-DOM` and `F-ILOT-LIM`. The class `F-EC-DOM` subsumes `F-ENTRE-DOM` and the class `F-ILOT` subsumes `F-ILOT-LIM`. The classes `F-ILOT-EC-DOM` and `F-PO--COM` give additional characteristics: this particular valley village is overlapped by a communal forest – that is a supplementary income for the village – (`F-PO-COM`) and contains at least one group of crops touching a state forest – that indicates surface constraints – (`F-ILOT-EC-DOM`).

7 Discussion

Our system has been implemented and used with land-use maps based on satellite images of the Lorraine region (East of France) [2]. As explained in Section 6, the result of a village territory analysis is a collection of classes of which the

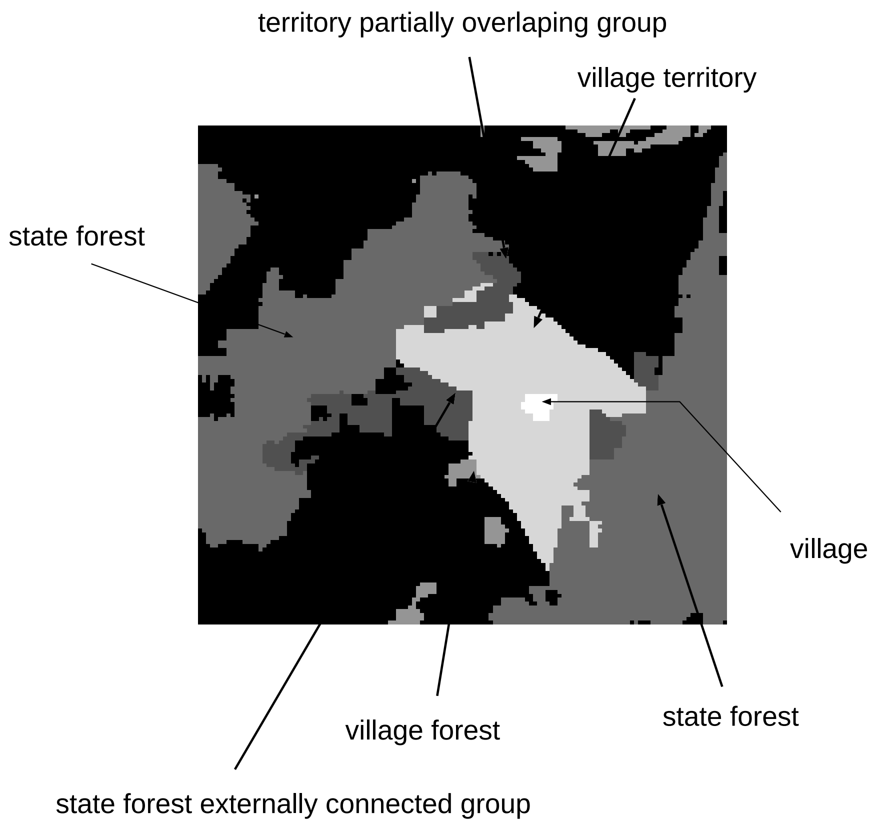


Figure 9: Classifying the village territory of Figure 1.

territory is an instance. Two territories that are instances of the same classes are supposed to share the same spatial structure. According to these results, the territories of an image are grouped into regions whose maps are drawn and analyzed by the agronomists. About 25% of the territories are misclassified or not classified: the first ones show that the landscape models must be made more precise. The second ones are territories that cannot be classified according to the models. The agronomists are very interested both in the village territories that are classified and in those that are not classified. The first ones confirm and generalize their knowledge which is acquired from field studies. The last ones are "special" territories that can indicate changes of the land use and thus that have to be further investigated. Finally, the agronomists can use these informations to choose a sample of village territories for their farm surveys. These first results are very positive but have to be confirmed by further studies and evaluations.

We have chosen to use an ObKR system since we needed both computation and representation capabilities. Furthermore, we have improved the representation capabilities of Y3: we have implemented attributes and facets to represent relations as "first class citizens". Attributes and facets are adapted to our problem but they are more general: they can be reused in other contexts for manipulating other sorts of structures. Furthermore, relations are organized within a hierarchy and we have accordingly modified and extended the classification mechanism of Y3.

We have improved Y3 relying on the representation and reasoning capabilities of description logics. We did not use description logics in the present work for a number of reasons, among which:

- both computation and representation capabilities were needed, and there are no description logics providing such kind of services (test functions in Classic do not fulfill our needs),
- it was necessary to deal with individuals. At the beginning of this work (1995), description logics with an Abox, a hierarchy of roles and negation of roles were not available.

At present, certain description logics have been developed with objectives that are similar to ours. For example, the description logic presented in [16] is appropriate for reasoning on qualitative spatial relations and spatial objects. This logic is extended to the polygon concrete domain, for the integration of quantitative reasoning. Special modeling constructs can be used to represent topological relations as defined roles. Spatial reasoning relies on two main operations in description logics reasoning, namely consistency checking and classification. Our work can be considered as complementary. It holds on regular raster regions and focus on the reification of **RCC-8** relations (relations correspond to concepts and not to roles), within an ObKR system, and on a lattice-based classification of spatial structures. The underlying reasoning mechanism is based in both cases on classification, and especially relation classification.

8 Conclusion

Our work is concerned with the representation of spatial structures and relations in an ObKR system. A system for checking spatial relations on raster images and for classifying image regions has been implemented and used for an application in agronomics. A number of extensions, inspired from description logics, have been done within the underlying ObKR system for representing relations as "first-class citizens", i.e., objects with their own properties.

In the future, improvements can be made regarding relation reification: the current state is sufficient for our application but it lacks generality. The inheritance mechanism in particular does not recognize the specialization of relation domains but only that of relation ranges, whereas the former should be taken into account to provide a general reification of relations. Sharing properties via "horizontal" relations (by contrast to "vertical" relations, such as specialization and instantiation) should also be taken into account.

Finally improvements can be made in two ways considering the application: adding indices to characterize spatial atoms and adding spatial relations to characterize spatial structures (qualitative distance, extended topology, orientation). Representing these last relations is more difficult than representing topological relations since the former are (mostly) ternary relations. Moreover, it is still necessary to find a unique logical framework to represent all these qualitative relations.

References

1. A. Artale, E. Franconi, N. Guarino, and L. Pazzi. Part-whole relations in object-oriented systems: An overview. *Data & Knowledge Engineering*, 20(3):347–383, 1996.
2. J. Bachacou. Map of the land use in Lorraine based on Landsat images, crop year 1989-90. Document, 1995. Laboratoire d'Intelligence Artificielle et de Biométrie, INRA–Nancy, France.
3. M. Benoît. La gestion territoriale de l'activité agricole dans un village lorrain. *Mappemonde*, (4):15–17, 1990.
4. E. Blake and S. Cook. On Including Part Hierarchies in Object-Oriented Languages, with an Implementation in Smalltalk. In J. Bézivin, J.-M. Hullot, P. Cointe, and H. Lieberman, editors, *European conference on object-oriented programming (ECOOP'87)*, LNCS 276, pages 45–54. Springer Verlag, 1987.
5. R.J. Brachman, D.L. McGuinness, P.F. Patel-Schneider, L.A. Resnick, and A. Borgida. Living with CLASSIC: When and How to Use a KL-ONE-Like Language. In J. Sowa, editor, *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, pages 401–456. Morgan Kaufmann Publishers, 1991.
6. D. Calvanese. *Unrestricted and Finite Model Reasoning in Class-Based Representation Formalisms*. Dottorato di Ricerca in Informatica, Università di Roma "La Sapienza", Italia, 1996.
7. E. Clementini, P. D. Felice, and D. Hernàndez. Qualitative Representation of positional information. *Artificial Intelligence*, 95:317–356, 1997.
8. A. G. Cohn, B. Bennett, J. Gooday, and N. M. Gotts. Representing and reasoning with qualitative spatial relations about regions. In Stock [34], pages 97–134.
9. A. G. Cohn and A. C. Varzi. Connection relations in mereotopology. In H. Prade, editor, *13th European Conference on Artificial Intelligence, ECAI'98*, pages 150–154. John Wiley & Sons, 1998.
10. B.A. Davey and H.A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, Cambridge, UK, 1990.
11. J.-P. Deffontaines. Organisation spatiale de l'activité agricole et développement d'une petite région lorraine. *Mappemonde*, (4):12–14, 1990.
12. F.-M. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in description logics. In G. Brewka, editor, *Principles of Knowledge Representation*, pages 191–236. CSLI Publications, 1996.
13. R. Ducournau. Y3 : YAFOOL, the object oriented language, 1991. SEMA Group.

14. M. J. Egenhofer. A Formal Definition of Binary Topological Relationships. In W. Litwin and H.J. Schek, editors, *Foundations of data organization and algorithms, Proceedings of the 3rd international conference, FODO 1989*, LNCS 367, pages 457–472. Springer Verlag, 1989. Paris, juin.
15. M. J. Egenhofer and J. Sharma. Topological Relations Between Regions in $\Re^2$ and Z^2. In D. J. Abel and B. C. Ooi, editors, *Advances in spatial databases: Proceedings of the Third International Symposium, SSD'93*, LNCS 692, pages 316–336. Springer Verlag, 1993.
16. V. Haarslev, C. Lutz, and R. Möller. A description logic with concrete domains and a role-forming predicate operator. *Journal of Logic and Computation*, 9(3):351–384, 1999.
17. D. Hernández. *Qualitative Representation of Spatial Knowledge*. Springer-Verlag, 1994. LNAI 804.
18. I. Horrocks, U. Sattler, and S. Tobies. Practical Reasoning for Expressive Description Logics. In H. Ganziger, D. McAllester, and A. Voronkov, editors, *Logic for Programming and Automated Reasoning, 6th International Conference LPAR'99*, LNAI 1705, pages 161–180. Springer, 1999.
19. F. Le Ber. Modélisation des connaissances et raisonnements pour l'analyse des paysages agraires à partir de données satellitaires. Doctorat de l'Université de Nancy I, France, 1993.
20. F. Le Ber and L. Mangelinck. A formal representation of landscape spatial patterns to analyze satellite images. *AI Applications*, 12(1-3):51–59, 1998.
21. F. Le Ber, L. Mangelinck, and A. Napoli. Design and comparison of lattices of topological relations for spatial representation and reasoning. INRIA Research Report RR-4321, 2001.
22. F. Le Ber and A. Napoli. Design and comparison of lattices of topological relations based on Galois lattice theory. In D. Fensel, F. Giunchiglia, D. McGuinness, and M.-A. Williams, editors, *Principles of Knowledge Representation and Reasoning – KR2002 Conference Proceedings*, pages 37–46. Morgan Kaufmann, 2002.
23. C. Lutz and U. Sattler. Mary likes all cats. In F. Baader and U. Sattler, editors, *2000 International Workshop on Description Logics*, pages 213–226. Aachener Beiträge zur Informatik, 2000.
24. R.M. MacGregor. Representing reified relations in Loom. *Journal of Experimental & Theoretical Artificial Intelligence*, 5(2&3):179–183, 1993.
25. L. Mangelinck. Représentation et classification de structures spatiales. Application à la reconnaissance de paysages agricoles. Thèse de doctorat, Université Henri Poincaré Nancy 1, France, 1998.
26. A. Napoli, C. Laurenço, and R. Ducournau. An object-based representation system for organic synthesis planning. *International Journal of Human-Computer Studies*, 41(1/2):5–32, 1994.
27. B. Nebel. *Reasoning and Revision in Hybrid Representation Systems*. LNCS 422. Springer Verlag, 1990.
28. P. F. Patel-Schneider and W. Swartout. Description logic specification from the KRSS effort. Working version (draft), 1993.
29. D. A. Randell and A. G. Cohn. Exploiting Lattices in a Theory of Space and Time. *Computers Math. Applic.*, 23(6-9):459–476, 1992.
30. D. A. Randell, Z. Cui, and A. G. Cohn. A Spatial Logic based on Regions and Connection. In B. Nebel, C. Rich, and W. Swartout, editors, *3rd International Conference on Principles of Knowledge Representation and Reasoning*, pages 165–176. Morgan Kaufmann Publishers, 1992.
31. J. Rumbaugh. Relations as Semantic Constructs in an Object-Oriented Language. In L. Power and Z. Weiss, editors, *Conference on Object-Oriented Programming Systems, Languages and Applications, (OOPSLA 87)*, pages 466–481, 1987. ACM SIGPLAN Notices 22(12).

32. J. Rumbaugh. Building boxes: Composite Objects. *Journal of Object-Oriented Programming*, 7(7):12–22, 1994.
33. M. Schmidt-Schauß and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991.
34. O. Stock, editor. *Spatial and Temporal Reasoning*. Kluwer Academic Publishers, 1997.
35. L. Vieu. A Logical Framework for Reasoning about Space. In A. U. Frank and I. Campari, editors, *Spatial Information Theory, COSIT'93*, LNCS 716, pages 25–35. Springer-Verlag, 1993.
36. L. Vieu. Spatial Representation and Reasoning in Artificial Intelligence. In Stock [34], pages 5–42.