A Logic-based Formalism for Reasoning about Visual Representations

Volker Haarslev
University of Hamburg, Computer Science Department
Vogt-Kölln-Str. 30, 22527 Hamburg, Germany
http://kogs-www.informatik.uni-hamburg.de/~haarslev/

Abstract

This article presents a logic-based formalism for formal reasoning about visual representations. This formalism is based on previous work about describing visual notations [Haarslev, 1998a]. However, in this article we discuss major extensions to this formalism providing *decidable* reasoning mechanisms that support truly spatial domains such as geographical information systems (GIS). We demonstrate the application of this formalism to specifying semantics of visual query languages for GIS and to meta reasoning about spatial queries.

Contents

1 Motivation and Introduction

Our logic formalism integrates research from the AI community –especially description logic and constraint solving– and from the qualitative spatial reasoning community. It can be considered as a decidable formalism for spatial knowledge representation. We think that our formalism is well suited to handling truly visual representations. In the scope of this article we consider representations as 'truly visual' that exploit geometric and/or depictional constraints permitted by the plane. A good example is the domain of visual spatial query languages for GIS, where query languages usually deal with geographic (two-dimensional) entities such as lakes, rivers, forests, etc. that are also represented as two-dimensional elements of the query language. With the help of our logic formalism we propose solutions for specifying semantics of visual spatial queries and for reasoning about the soundness of and subsumption between queries. This approach to specifying semantics is innovative since it based on a decidable logic offering means for expressing qualitative spatial constraints.

Our formalism is based on the recently developed description logic $\mathcal{ALCRP}(\mathcal{D})$ [Haarslev et al., 1998a; Haarslev et al., 1999] offering mechanisms for integrating so-called *concrete domains*. We already explored suitable candidates for concrete domains that are based on theories about qualitative spatial reasoning (e.g. Egenhofer's work [Egenhofer, 1991] or RCC8 [Randell et al., 1992]) or about time (e.g. Allen's interval logic [Allen, 1983]). These findings are also relevant for visual language (VL) theory since they can provide a basis for specifying visual procedural semantics (time) or for specifying qualitative and quantitative (e.g. geometric) relationships about space.

However, in this article we only focus on *spatial* reasoning about visual representations. In our earlier work [Haarslev, 1995; Haarslev and Wessel, 1996; Haarslev, 1998a] spatial relations (e.g. touching, containing, etc.) are considered as uninterpreted (primitive) binary relations with respect to description logic theory. In order to correctly deal with spatial objects we needed an external geometric reasoner about the factual world asserting proper spatial relationships and object properties. This was a feasible approach but its reasoning about space was still incomplete.

Our new decidable description logic $\mathcal{ALCRP}(\mathcal{D})$ offers a means to correctly specify qualitative spatial relations and to ground them on computational geometry [Haarslev and Möller, 1997]. We motivate the application of this formalism to visual spatial query languages with our recently developed spatial query system VISCO (Vivid Spatial Constellations) which offers a sketch-based query language for GIS [Haarslev and Wessel, 1997; Wessel and Haarslev, 1998].

We like to emphasize that the work on VL theory presented in this article extends our previous research as summarized in [Haarslev, 1998a], where we used a logic that is more expressive than $\mathcal{ALCRP}(\mathcal{D})$ since it allows *qualified number restrictions* but also less expressive than $\mathcal{ALCRP}(\mathcal{D})$ since it has no *defined roles*. To the best of our knowledge there is no other decidable logic known that offers the kind of spatial reasoning supported by $\mathcal{ALCRP}(\mathcal{D})$. Furthermore, the use of $\mathcal{ALCRP}(\mathcal{D})$ as a tool for VL theory, especially for formally describing semantics of visual spatial query languages, is rather innovative and unprecedented.

The remainder of this article is structured as follows. The next section introduces our description logic $\mathcal{ALCRP}(\mathcal{D})$ and its application as a formalism for reasoning about visual representations. Afterwards we focus on reasoning with description logic and concrete domains in more detail and discuss advantages of $\mathcal{ALCRP}(\mathcal{D})$. The fourth section applies $\mathcal{ALCRP}(\mathcal{D})$ to specifying semantics of visual spatial query languages and to meta-reasoning about queries. We conclude the article with a review of closely related work and give an outlook to ongoing research.

2 The Description Logic $\mathcal{ALCRP}(\mathcal{D})$

This section gives a brief introduction to the description logic $\mathcal{ALCRP}(\mathcal{D})$ and to description logic (DL) theory in general summarizing the notions important for this article. We refer to [Brachman et al., 1991; Woods and Schmolze, 1992; Borgida, 1995] for more complete information about description logic theory.

Many DL theories can be viewed as subsets of first-order predicate logic. However it is important to note that particular DL theories are only considered as practical if they are based on *sound and complete* reasoning algorithms, i.e. the *decidability* of a DL is of utmost importance. Of course, this is a major distinction to reasoning with general first-order predicate logic.

DL theories are based on the ideas of structured inheritance networks [Brachman and Schmolze, 1985]. The syntax of a DL has similarities to a term rewriting language usually (but not necessarily) restricting the left-hand side of equations to single unique term names. In a DL a factual world consists of named individuals and their relationships that are asserted through binary relations. Hierarchical descriptions about sets of individuals form the terminological knowledge. Descriptions about sets of individuals are called *concepts* and binary relations are called *roles*. Descriptions consist of identifiers denoting concepts, roles, and individuals, and of description constructors. For any individual x the set $F_{\mathsf{x}} = \{\mathsf{y} | \mathsf{r}(\mathsf{x}, \mathsf{y})\}$ is called the set of *fillers* of the role r. A role which may have at most one filler for each individual (i.e. the set F_{x} contains at most one element for all x) is referred to as *feature*.

For instance, consider the following description used in our GIS scenario with the intended meaning "a cottage that is enclosed by a forest" that contains concept names (e.g. cottage), role names (e.g. is_g_inside), and constructors (e.g. $\sqcap$ and $\exists$).

$$\textbf{cottage\_in\_forest} \doteq \mathsf{cottage} \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{forest}$$

An informal first-order logic description for this example and other following examples is also given to help readers unfamiliar with description logics (see also the next section for specifying a model-based semantics). Concepts are described by unary predicates.

$$\text{cottage\_in\_forest}(x) \equiv \text{cottage}(x) \wedge \exists\, y : \text{is\_g\_inside}(x, y) \wedge \text{forest}(y)$$

The expressiveness and computational complexity of a particular DL depends on the variety of employed description constructors. Various complexity results for subsumption algorithms for specific description logics are summarized in [Woods and Schmolze, 1992]. Recent findings for $\mathcal{ALCRP}(\mathcal{D})$ suggest that deciding satisfiability is at least in EXPTIME.

2.1 Terminologies

In this section, the language (syntax and semantics) for defining concepts and roles in $\mathcal{ALCRP}(\mathcal{D})$ is presented. $\mathcal{ALCRP}(\mathcal{D})$ is parameterized with a concrete domain which consists of a set of concrete objects and a set of predicates.

Concrete Domains: A *concrete* domain $\mathcal{D}$ is a pair $(\Delta_\mathcal{D}, \Phi_\mathcal{D})$, where $\Delta_\mathcal{D}$ is a set called the domain, and $\Phi_\mathcal{D}$ is a set of predicate names. Each predicate name P from $\Phi_\mathcal{D}$ is associated with an arity n, and an n-ary predicate $\mathsf{P}^\mathcal{D} \subseteq \Delta_\mathcal{D}^n$. A concrete domain $\mathcal{D}$ is called *admissible* iff (1) the set of its predicate names is closed under negation (i.e. for any $\mathsf{P} \in \Phi_\mathcal{D}$ there exists a $\overline{\mathsf{P}} \in \Phi_\mathcal{D}$ denoting the negation of P) and contains a name $\top_\mathcal{D}$ for $\Delta_\mathcal{D}$ and (2) the satisfiability problem for finite conjunctions of predicates is decidable.

A concrete domain can be understood as a device providing a bridge between conceptual reasoning with abstract entities and (qualitative) constraint reasoning with concrete or symbolic data. Examples for admissible concrete domains are $\mathcal{R}$ (over the set $\mathbb{R}$ of all real numbers with predicates built by first order means from (in)equalities between integer polynomials in several indeterminates, see [Tarski, 1951]) or $\mathcal{S}_2$ (over the set of all two-dimensional polygons with topological relations from Figure 2 as predicates, see [Haarslev et al., 1999]). The name 'concrete domain' is in some sense misleading since it suggests that a concrete domain realizes reasoning about 'concrete' (e.g. numeric) data. This kind of reasoning is sometimes supported (e.g. in the domain $\mathcal{R}$) but in our application we mainly use concrete domains for reasoning about the satisfiability of finite conjunctions of predicates. For instance, the domain $\mathcal{S}_2$ qualitatively decides the satisfiability of conjunctions such as $\mathsf{touching}(I_1, I_2) \wedge \mathsf{contains}(I_2, I_3) \wedge \mathsf{touching}(I_1, I_3)$ without any notion for 'concrete' polygons. This is a well-known example for a constraint satisfaction problem.

Without loss of generality we introduce a λ-like notation for anonymous predicates of the domain $\mathcal{R}$. Formally, each anonymous predicate and its negation could be replaced by unique names for the λ-term and its negated counterpart and, moreover, the negation sign in front of a λ-term can be safely moved inside of this term.

The fact that $\mathcal{ALCRP}(\mathcal{D})$ can be parameterized by only one concrete domain is not a limitation because in [Haarslev et al., 1999] it is shown that the union of two admissible concrete domains again results in an admissible concrete domain. For instance, the union of the concrete domains $\mathcal{R}$ and $\mathcal{S}_2$ might also be a useful domain providing tools for expressing spatial and arithmetic constraints. We are now ready to define role terms in $\mathcal{ALCRP}(\mathcal{D})$.

Role Terms: Let R and F be disjoint sets of role and feature names, respectively. Any element of R and any element of F is an *atomic role term*. The elements of F are also called *features*. A composition of features (written $\mathsf{f}_1\mathsf{f}_2\cdots$) is called a feature chain. A feature chain of length one is also a feature chain. If $\mathsf{P} \in \Phi_{\mathcal{D}}$ is a predicate name with arity $n+m$ and $\mathsf{u}_1, \ldots ,\mathsf{u}_n$ as well as $\mathsf{v}_1, \ldots ,\mathsf{v}_m$ are $n+m$ feature chains, then the expression

- $\exists\,(\mathsf{u}_1,\ldots,\mathsf{u}_\mathsf{n})(\mathsf{v}_1,\ldots,\mathsf{v}_\mathsf{m})\,.\,\mathsf{P}$ (*role-forming predicate restriction*)

is a *complex role term*. Let S be a role name and let T be a role term. Then $\mathsf{S} \doteq \mathsf{T}$ is a terminological axiom.

An example for using a role-forming predicate operator is the definition of a role is_g_inside for a corresponding topological predicate g_inside[1] (see Section 2.1 and Figure 1 for an explanation of the semantics). Intuitively speaking, this role holds for any pair of individuals (I_1, I_2) iff the associated spatial area (via the feature has_area) of I_1 is 'generally inside' of the area of I_2.

$$\textsf{\textbf{is\_g\_inside}} \doteq \exists\,(\mathsf{has\_area})(\mathsf{has\_area})\,.\,\mathsf{g\_inside}$$

The informal description in first-order logic is given as a binary predicate.

$$\text{is\_g\_inside}(x, y) \equiv \exists\, a, b : \text{has\_area}(x, a) \wedge \text{has\_area}(y, b) \wedge \text{g\_inside}(a, b)$$

Concept Terms: Let C be a set of concept names which is disjoint from R and F. Any element of C is an *atomic concept term*. If C and D are concept terms, R is an arbitrary role term or a feature, $\mathsf{P} \in \Phi_{\mathcal{D}}$ is a predicate name with arity n, and $\mathsf{u}_1, \ldots, \mathsf{u}_n$ are feature chains, then the following expressions are also concept terms:

- $\mathsf{C} \sqcap \mathsf{D}$ (*conjunction*),

[1] g_inside stands for 'generally inside' representing the union of 'equal', 'tangentially inside', and 'strictly inside' (see also Figure 2). We use the name g_inside instead of inside in order to avoid confusion with other approaches referring to 'inside' with a different meaning.

- $\mathsf{C} \sqcup \mathsf{D}$ (*disjunction*),
- $\neg\mathsf{C}$ (*negation*),
- $\exists\, \mathsf{R} \,.\, \mathsf{C}$ (*concept exists restriction*),
- $\forall\, \mathsf{R} \,.\, \mathsf{C}$ (*concept value restriction*), and
- $\exists\, \mathsf{u_1}, \ldots, \mathsf{u_n} \,.\, P$ (*predicate exists restriction*).

We illustrate the notion of concept and role terms by extending the cottage example mentioned above.

$$\textbf{reed\_cottage\_in\_forest} \doteq$$
$$\mathsf{cottage} \sqcap \exists\, \mathsf{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x \geq 30 \wedge x < 70)$$
$$\exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{forest} \sqcap \forall\, \mathsf{has\_roof} \,.\, (\mathsf{roof} \sqcap \forall\, \mathsf{has\_material} \,.\, \mathsf{reed})$$

The definition of reed_cottage_in_forest roughly has the intended meaning "something is a reed cottage in a forest if and only if it is a cottage located in a forest with 30-70 square meters of total space for the cottage and its roof is exclusively made of reed." This definition also gives an example for a predicate exists restriction for the domain $\mathcal{R}$ using a feature has_space. The informal description in first-order logic is as follows.

$$\begin{aligned} \text{reed\_cottage\_in\_forest}(x) \equiv\ & \text{cottage}(x) \wedge \\ & \exists\, a : \text{has\_space}(x, a) \wedge a \geq 30 \wedge a < 70 \wedge \\ & \exists\, y : \text{is\_g\_inside}(x, y) \wedge \text{forest}(y) \wedge \\ & \forall\, z_1 : \text{has\_roof}(x, z_1) \Rightarrow \\ & (\text{roof}(z_1) \wedge \forall\, z_2 : \text{has\_material}(z_1, z_2) \Rightarrow \text{reed}(z_2)) \end{aligned}$$

In order to ensure the decidability, we had to restrict the possible combinations of concepts terms w.r.t. defined roles (e.g. a nested concept term with defined roles such as $\forall\, \mathsf{is\_touching} \,.\, \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{cottage}$ is not allowed). Note that all examples in this article are restricted. However, this restrictedness criterion is beyond the scope of this article and is fully explained elsewhere [Haarslev et al., 1999].

Terminology: For any exists and value restrictions, the role term or list of feature chains may be written in parentheses. Let A be a concept name and D be a concept term. Then $\mathsf{A} \doteq \mathsf{D}$ and $\mathsf{A} \sqsubseteq \mathsf{D}$ are terminological axioms as well. A finite set of terminological axioms $\mathcal{T}$ is called a *terminology* or *TBox* if no concept or role name in $\mathcal{T}$ appears more than once on the left-hand side of a definition and, furthermore, if no cyclic definitions are present.

The previous examples already informally introduced concept axioms for defined concepts using the $\doteq$ operator. For convenience, we also allow the $\sqsubseteq$ operator for the definition of primitive concepts, i.e. their definition consists only of necessary conditions. The concept cottage is a good candidate for a primitive definition documenting that we omitted in our terminology other conditions that are not relevant for this modeling task. For instance, a cottage has to be at least a building (informal description in first-order logic in parentheses).

$\mathsf{\textbf{cottage}} \sqsubseteq \mathsf{building} \qquad (\forall\, x : \text{cottage}(x) \Rightarrow \text{building}(x))$

Of course, there exist other description logics that allow more than one axiom for a particular concept name or even support generalized concept inclusions (implications) with arbitrary concept terms on the left and right side of terminological axioms. These axioms can be used as a powerful modeling tool but are currently not supported in $\mathcal{ALCRP}(\mathcal{D})$ w.r.t. decidability.

Semantics: An *interpretation* $\mathcal{I} = (\Delta_\mathcal{I}, \cdot^\mathcal{I})$ consists of a set $\Delta_\mathcal{I}$ (the abstract domain) and an interpretation function $\cdot^\mathcal{I}$. The sets $\Delta_\mathcal{D}$ (see above) and $\Delta_\mathcal{I}$ must be disjoint. The interpretation function maps each concept name C to a subset $\mathsf{C}^\mathcal{I}$ of $\Delta_\mathcal{I}$, each role name R to a subset $\mathsf{R}^\mathcal{I}$ of $\Delta_\mathcal{I} \times \Delta_\mathcal{I}$, and each feature name f to a partial function $\mathsf{f}^\mathcal{I}$ from $\Delta_\mathcal{I}$ to $\Delta_\mathcal{D} \cup \Delta_\mathcal{I}$, where $\mathsf{f}^\mathcal{I}(a) = x$ will be written as $(a, x) \in f^\mathcal{I}$. If $u = \mathsf{f}_1 \cdots \mathsf{f}_n$ is a feature chain, then $u^\mathcal{I}$ denotes the composition $\mathsf{f}_1^\mathcal{I} \circ \cdots \circ \mathsf{f}_n^\mathcal{I}$ of the partial functions $\mathsf{f}_1^\mathcal{I}, \ldots, \mathsf{f}_n^\mathcal{I}$. Let the symbols C, D, R, P, u_1, ... ,u_m, and v_1, ... ,v_m be defined as above. Then the interpretation function can be extended to arbitrary concept and role terms as follows:

$$
\begin{aligned}
(\mathsf{C} \sqcap \mathsf{D})^\mathcal{I} &:= \mathsf{C}^\mathcal{I} \cap \mathsf{D}^\mathcal{I} \\
(\mathsf{C} \sqcup \mathsf{D})^\mathcal{I} &:= \mathsf{C}^\mathcal{I} \cup \mathsf{D}^\mathcal{I} \\
(\neg \mathsf{C})^\mathcal{I} &:= \Delta_\mathcal{I} \setminus \mathsf{C}^\mathcal{I} \\
(\exists\, \mathsf{R} \,.\, \mathsf{C})^\mathcal{I} &:= \{a \in \Delta_\mathcal{I} \mid \exists\, b \in \Delta_\mathcal{I} : (a,b) \in \mathsf{R}^\mathcal{I}, b \in \mathsf{C}^\mathcal{I}\} \\
(\forall\, \mathsf{R} \,.\, \mathsf{C})^\mathcal{I} &:= \{a \in \Delta_\mathcal{I} \mid \forall\, b \in \Delta_\mathcal{I} : (a,b) \in \mathsf{R}^\mathcal{I} \Rightarrow b \in \mathsf{C}^\mathcal{I}\} \\
(\exists\, \mathsf{u_1}, \ldots, \mathsf{u_n} \,.\, \mathsf{P})^\mathcal{I} &:= \\
&\quad \{a \in \Delta_\mathcal{I} \mid \exists\, \mathsf{x}_1, \ldots, \mathsf{x}_n \in \Delta_\mathcal{D} : \\
&\quad\; (a, \mathsf{x}_1) \in \mathsf{u}_1^\mathcal{I}, \ldots, (a, \mathsf{x}_n) \in \mathsf{u}_n^\mathcal{I}, (\mathsf{x}_1, \ldots, \mathsf{x}_n) \in \mathsf{P}^\mathcal{D}\} \\
(\exists\, (\mathsf{u_1}, \ldots, \mathsf{u_n})(\mathsf{v_1}, \ldots, \mathsf{v_m}) \,.\, \mathsf{P})^\mathcal{I} &:= \\
&\quad \{(a,b) \in \Delta_\mathcal{I} \times \Delta_\mathcal{I} \mid \exists\, \mathsf{x}_1, \ldots, \mathsf{x}_n, \mathsf{y}_1, \ldots, \mathsf{y}_m \in \Delta_\mathcal{D} : \\
&\quad\; (a, \mathsf{x}_1) \in \mathsf{u}_1^\mathcal{I}, \ldots, (a, \mathsf{x}_n) \in \mathsf{u}_n^\mathcal{I}, \\
&\quad\; (b, \mathsf{y}_1) \in \mathsf{v}_1^\mathcal{I}, \ldots, (b, \mathsf{y}_m) \in \mathsf{v}_m^\mathcal{I}, \\
&\quad\; (\mathsf{x}_1, \ldots, \mathsf{x}_n, \mathsf{y}_1, \ldots, \mathsf{y}_m) \in \mathsf{P}^\mathcal{D}\}
\end{aligned}
$$

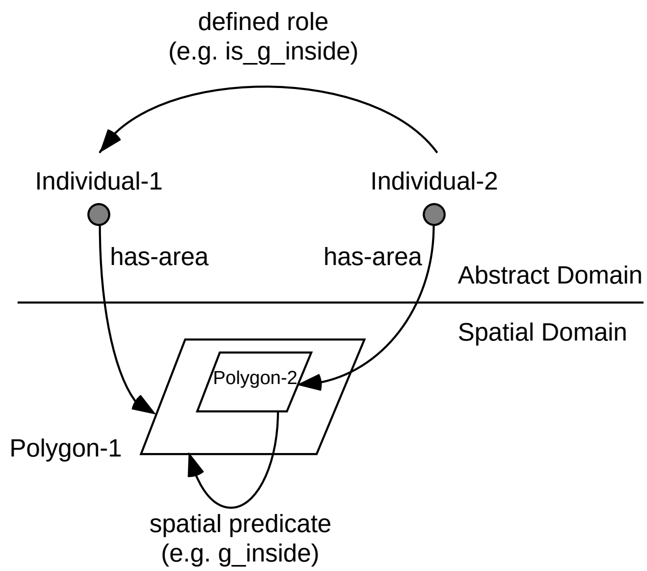

Figure 1: Relationship between abstract and spatial domain.

An interpretation $\mathcal{I}$ is a *model* of a TBox $\mathcal{T}$ iff it satisfies $\mathsf{A}^{\mathcal{I}} = \mathsf{C}^{\mathcal{I}}$ for all terminological axioms $\mathsf{A} \doteq \mathsf{C}$ in $\mathcal{T}$, and $\mathsf{A}^{\mathcal{I}} \subseteq \mathsf{C}^{\mathcal{I}}$ for $\mathsf{A} \sqsubseteq \mathsf{C}$ respectively.

Note that a concrete domain is deliberately separated from an abstract domain. Only features can have elements of the concrete (as well as the abstract) domain as fillers. Predicates (from the concrete domain) may hold only between elements of the *concrete* domain while roles may hold only between elements of the *abstract* domain.

The last formula of the semantics given above defines the interpretation of the role-forming predicate operator. Figure 1 illustrates for the domain $\mathcal{S}_2$ the idea behind the semantics of the role-forming predicate operator. The spatial predicates (e.g. g_inside) operate on concrete domain values (e.g. polygon descriptions) that are attached via features to corresponding abstract individuals. If a role (e.g. is_g_inside) is *defined* by a predicate (e.g. g_inside), then every pair (p_1, p_2) of polygons, where p_1 is the filler of has_area for the abstract individual i_1 and p_2 for i_2 respectively, is tested whether the binary predicate $\mathsf{g\_inside}(\mathsf{p}_2, \mathsf{p}_1)$ is fulfilled. In case of a successful test the role membership (e.g. is_g_inside) is also established for the abstract individuals i_1 and i_2, i.e. it holds $\mathsf{is\_g\_inside}(\mathsf{i}_2, \mathsf{i}_1)$. This also applies for the opposite direction. If a role membership is asserted for a pair of abstract individuals, their associated concrete feature fillers are either established with the corresponding predicate or verified if concrete feature fillers already exist.

2.1.1 TBox Example

Consider the following descriptions that are part of our GIS scenario. The first axiom defines a cottage as a specialization of a building. The next three concepts define cottages of various sizes using the predicate exists restriction. Please note that these three definitions are mutually exclusive due to their size-restricting predicates. The fifth axiom introduces a defined role is_g_inside that is used to characterize a cottage located in a forest. This is specialized to an affordable cottage with some size restrictions. The eighth axiom defines a reed cottage located in a forest. This concept is subsumed by normal_cottage. The last axiom defines an expensive cottage as a spacious cottage located in a forest.

$\textbf{cottage} \sqsubseteq \text{building}$

$\textbf{small\_cottage} \doteq \text{cottage} \sqcap \exists\, \text{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x < 30)$

$\textbf{normal\_cottage} \doteq \text{cottage} \sqcap \exists\, \text{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x \geq 30 \wedge x < 70)$

$\textbf{spacious\_cottage} \doteq \text{cottage} \sqcap \exists\, \text{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x \geq 70)$

$\textbf{is\_g\_inside} \doteq \exists\, (\text{has\_area})(\text{has\_area}) \,.\, \text{g\_inside}$

$\textbf{cottage\_in\_forest} \doteq \text{cottage} \sqcap \exists\, \text{is\_g\_inside} \,.\, \text{forest}$

$\textbf{affordable\_cottage} \doteq \text{cottage\_in\_forest} \sqcap (\text{small\_cottage} \sqcup \text{normal\_cottage})$

$\textbf{reed\_cottage\_in\_forest} \doteq$

$\quad \text{cottage} \sqcap \exists\, \text{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x \geq 30)$

$\quad \exists\, \text{is\_g\_inside} \,.\, \text{forest} \sqcap \forall\, \text{has\_roof} \,.\, (\text{roof} \sqcap \forall\, \text{has\_material} \,.\, \text{reed})$

$\textbf{expensive\_cottage} \doteq \text{cottage\_in\_forest} \sqcap \text{spacious\_cottage}$

2.2 The Assertional Language

The *assertional language* of a DL is designed for stating concept or role memberships of named individuals that are used to describe the factual world. With respect to concrete domains we distinguish *abstract* and *concrete* individuals. Abstract individuals are elements of the abstract domain and have to be members of concepts. Concrete individuals are elements of the concrete domain and are used as parameters for predicates. For instance, in the VL domain abstract individuals could represent both syntactic elements[2] of the query language (e.g. geometric figures such as circles, rectangles, etc.) and the corresponding semantic entities (e.g. lake, estate, etc). Concrete individuals could be used to represent geometric properties of both syntactic and semantic entities.

The set of assertions (ABox) has to comply to the definitions declared in the TBox. An *ABox* of $\mathcal{ALCRP}(\mathcal{D})$ is a finite set of assertions defined as follows.

[2] This will not be discussed in this article but see Section 4.

2.2.1 Syntax

Let I_A and I_D be two disjoint sets of individual names for the abstract and concrete domain. If C is a concept term, R an atomic or complex role term, f a feature name, P a predicate name with arity n, a and b are elements of I_A, x is an element of I_A or I_D, and $\mathsf{x}_1, \ldots, \mathsf{x}_n$ are elements of I_D, then the following expressions are *assertional axioms*.

$$
\begin{aligned}
\mathsf{a} : \mathsf{C} &\quad (\textit{concept membership}) \\
(\mathsf{a}, \mathsf{b}) : \mathsf{R} &\quad (\textit{role filler}) \\
(\mathsf{a}, \mathsf{x}) : \mathsf{f} &\quad (\textit{feature filler}) \\
(\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n}) : \mathsf{P} &\quad (\textit{concrete domain predicate membership})
\end{aligned}
$$

2.2.2 Semantics

For specifying the semantics of ABox assertions we have to extend the interpretation function $\mathcal{I}$. An *interpretation* for the assertional language is an interpretation for the concept language which additionally maps every individual name from I_A to a single element of $\Delta_{\mathcal{I}}$ and every individual name from I_D to a single element from $\Delta_{\mathcal{D}}$. We assume that the unique name assumption does not hold, that is $\mathsf{a}^{\mathcal{I}} = \mathsf{b}^{\mathcal{I}}$ may hold even if $\mathsf{a} \neq \mathsf{b}$.

$$
\begin{aligned}
\mathsf{a} : \mathsf{C} &\text{ iff } \mathsf{a}^{\mathcal{I}} \in \mathsf{C}^{\mathcal{I}} \\
(\mathsf{a}, \mathsf{b}) : \mathsf{R} &\text{ iff } (\mathsf{a}^{\mathcal{I}}, \mathsf{b}^{\mathcal{I}}) \in \mathsf{R}^{\mathcal{I}} \\
(\mathsf{a}, \mathsf{x}) : \mathsf{f} &\text{ iff } \mathsf{f}^{\mathcal{I}}(\mathsf{a}^{\mathcal{I}}) = \mathsf{x}^{\mathcal{I}} \\
(\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n}) : \mathsf{P} &\text{ iff } \mathsf{x}_1^{\mathcal{I}}, \ldots, \mathsf{x}_n^{\mathcal{I}} \in \mathsf{P}^{\mathcal{D}}
\end{aligned}
$$

2.2.3 ABox Example

We assume for this ABox example a terminology as given by the TBox in Section 2.1.1. The following assertions illustrate the four different types of ABox assertions using the cottage scenario.

$$
\begin{aligned}
&\mathsf{c}_1 : \mathsf{cottage},\ \mathsf{c}_1 : \exists\, \mathsf{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x < 30) \\
&\mathsf{c}_2 : \mathsf{cottage},\ (\mathsf{c}_2, 60) : \mathsf{has\_space},\ (\mathsf{c}_2, \mathsf{S}_{\mathsf{c}_2}) : \mathsf{has\_area} \\
&\ \mathsf{f} : \mathsf{forest},\ (\mathsf{f}, \mathsf{S}_\mathsf{f}) : \mathsf{has\_area},\ (\mathsf{S}_{\mathsf{c}_2}, \mathsf{S}_\mathsf{f}) : \mathsf{g\_inside}
\end{aligned}
$$

Based on the semantics given above our $\mathcal{ALCRP}(\mathcal{D})$ reasoner [Turhan, 1998] will infer that c_1 is a member of small_cottage and c_2 is a member of the concept affordable_cottage. $\mathsf{S}_{\mathsf{c}_2}$ and S_{f} denote the associated area polygon of c_2 and f.

2.3 Reasoning Services

The notion of a *model* (see above) is used to define the reasoning services that a DL inference engine has to provide, i.e. it proves for every concept specification whether the following conditions hold:

- a term A *subsumes* another term B if and only if for every model $\mathcal{I}$, $\mathsf{B}^{\mathcal{I}} \subseteq \mathsf{A}^{\mathcal{I}}$;
- a term A is *coherent/satisfiable* if and only if there exists at least one model $\mathcal{I}$ such that $\mathsf{A}^{\mathcal{I}} \neq \emptyset$;
- terms A and B are *disjoint* if and only if for every model $\mathcal{I}$, $\mathsf{A}^{\mathcal{I}} \cap \mathsf{B}^{\mathcal{I}} = \emptyset$;
- terms A and B are *equivalent* if and only if for every model $\mathcal{I}$, $\mathsf{A}^{\mathcal{I}} = \mathsf{B}^{\mathcal{I}}$.
- An *ABox* $\mathcal{A}$ is *consistent* if and only if there exists a model $\mathcal{I}$ of $\mathcal{A}$.

Proper DL systems (i.e. implementations of a DL) are guided by this semantics and implement these inference services. They usually distinguish two reasoning components. The *terminological reasoner* or *classifier* operates on the TBox and classifies concepts with respect to subsumption relationships between them and organizes them into a taxonomy. The classifier automatically performs normalization of concept definitions as well as consistency checking operations and offers retrieval facilities about the classification hierarchy. The *assertional reasoner* or *realizer* operates on the ABox in accordance with the definitions in the TBox and recognizes and maintains the concept membership (i.e. the set of the most specific subsuming concepts) and role membership of individuals. With respect to concrete domains it also proves for concrete individuals the satisfiability of predicates. Assertional reasoners usually support a query language for accessing stated and deduced constraints. Some query languages offer the expressiveness of the full first-order calculus.

3 Comparison with $\mathcal{ALC}$ and $\mathcal{ALC}(\mathcal{D})$

Why is the incorporation of concrete domains into DL theory so important for VL theory? In our previous approach we used a standard DL that can neither deal with concepts defined with the help of algebra nor with roles defined by predicates. For instance, it was not possible to specify a *defined* concept normal_cottage that represents every conceivable cottage whose space is between $30m^2$ and $70m^2$. It was only possible to rely on an external reasoner that had to assert the concept membership for normal_cottage in the ABox. The TBox reasoning was incomplete in our previous approach with respect to the use of algebra. For instance, the DL reasoner could not detect the mutual exclusiveness of a normal_cottage and a spacious_cottage. Its reasoning with respect to topological roles was also incomplete and is still incomplete in other logics such as $\mathcal{ALC}$ and $\mathcal{ALC}(\mathcal{D})$. In the following we demonstrate the shortcomings of $\mathcal{ALC}$ and $\mathcal{ALC}(\mathcal{D})$ and compare them with $\mathcal{ALCRP}(\mathcal{D})$.

$\mathcal{ALCRP}(\mathcal{D})$ is a major extension of the description logic $\mathcal{ALC}(\mathcal{D})$ [Baader and Hanschke, 1991a; Baader and Hanschke, 1991b]. $\mathcal{ALCRP}(\mathcal{D})$ adds the role-forming predicate operator (see above for the semantics) that is not present in $\mathcal{ALC}(\mathcal{D})$. This operator significantly extends the expressivity of the language. Moreover, the possible union of concrete domains enhances $\mathcal{ALCRP}(\mathcal{D})$'s expressiveness due to the role-forming predicate restriction that can be used to relate pairs of individuals in a way that is not possible in $\mathcal{ALC}(\mathcal{D})$ (see also Section 2.1). This is in contrast to $\mathcal{ALC}(\mathcal{D})$ where the union of two concrete domains can be reduced to syntactic transformations.

The six concept-forming operators of $\mathcal{ALCRP}(\mathcal{D})$ but not the role-forming predicate operator are exactly the language elements of $\mathcal{ALC}(\mathcal{D})$ that is a true subset of $\mathcal{ALCRP}(\mathcal{D})$. The basic logic $\mathcal{ALC}$, in turn, is a true subset of $\mathcal{ALC}(\mathcal{D})$. $\mathcal{ALC}$ has no notion of concrete domains. Its language consists only of the first five concept operators as introduced in Section 2.1.

3.1 Reasoning with $\mathcal{ALC}$ and $\mathcal{ALC}(\mathcal{D})$

With the logic employed in [Haarslev, 1998a] we can define the various 'cottage concepts' only as primitives and an external reasoner has to assert the concept membership of corresponding individuals.

$\textbf{small\_cottage} \sqsubseteq \text{normal\_cottage}$

$\textbf{normal\_cottage} \sqsubseteq \text{spacious\_cottage}$

$\textbf{spacious\_cottage} \sqsubseteq \text{cottage} \sqcap \exists\, \text{has\_space} \,.\, \top$

However, an $\mathcal{ALC}$ reasoner cannot catch the intended contradiction[3] contained in the following ABox that might be caused by an erroneous assertion of the external reasoner.

$\text{c}_1 : \text{small\_cottage},\ (\text{c}_1, 65) : \text{has\_space}$

The $\mathcal{ALC}$ reasoner would correctly classify this ABox as consistent although one would like to catch this as an invalid entry. This is the motivation behind our notion of *unintended models*. We define a model as *unintended* in a description logic, if the logic cannot fully capture the intended semantics of a particular domain and thus its reasoner correctly classifies a TBox or ABox as coherent that should turn out as incoherent in a more powerful logic.

[3] A small cottage has a floor space of less than $30m^2$.

The deficiency of the logic $\mathcal{ALC}$ to deal with concepts defined with the help of algebra was a primary motivation for the 'concrete domain' approach realized by the description logic $\mathcal{ALC}(\mathcal{D})$. The interesting question for VL theory is "what can be already modeled with $\mathcal{ALC}(\mathcal{D})$ and when and why do we need $\mathcal{ALCRP}(\mathcal{D})$?"

The modeling capabilities of $\mathcal{ALC}(\mathcal{D})$ are expressive enough to sufficiently categorize cottages of varying sizes, if we use the concrete domain $\mathcal{R}$ over the set $\mathbb{R}$ of all real numbers with predicates built by first order means from (in)equalities between integer polynomials in several indeterminates. In the following we present a slightly modified definition for the concepts small_cottage, normal_cottage, and spacious_cottage.

$$\textbf{small\_cottage} \doteq \text{cottage} \sqcap \exists\, \text{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x < 30)$$
$$\textbf{normal\_cottage} \doteq \text{cottage} \sqcap \exists\, \text{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x < 70)$$
$$\textbf{spacious\_cottage} \doteq \text{cottage} \sqcap \exists\, \text{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x < 200)$$

A reasoner for $\mathcal{ALC}(\mathcal{D})$ immediately recognizes the subsumption relationship between these concepts, i.e. spacious_cottage subsumes normal_cottage that, in turn, subsumes small_cottage.

Using the following ABox assertions, the $\mathcal{ALC}(\mathcal{D})$ reasoner will recognize c_1 as a member of spacious_cottage and c_2 as a member of both spacious_cottage and normal_cottage.

$$c_1 : \text{cottage}, \ (c_1, 80) : \text{has\_space}$$
$$c_2 : \text{cottage}, \ (c_2, 60) : \text{has\_space}$$

3.2 Unintended Models in $\mathcal{ALC}(\mathcal{D})$

In the following we assume that both $\mathcal{ALC}(\mathcal{D})$ and $\mathcal{ALCRP}(\mathcal{D})$ are instantiated with the union of the concrete domains $\mathcal{R}$ (see previous section) and $\mathcal{S}_2$ for polygons in the plane (see also [Haarslev et al., 1999]). The domain $\mathcal{S}_2$ defines a set of predicates recognizing spatial relationships between polygons in analogy to well-known qualitative spatial reasoning approaches (e.g. [Egenhofer, 1991] or [Randell et al., 1992]). We organized the spatial predicates in a subsumption hierarchy that is shown in Figure 2. In contrast to [Haarslev et al., 1999] we use a different (more natural) naming scheme for the predicates. Figure 3 illustrates some of these predicates. As an ontological commitment we use a feature has_area that may have a polygon as concrete filler describing the occupied space of an individual.

Characteristics of visual languages very often depend on spatial relationships between language elements. For instance, the topological relations used by GenEd [Haarslev and Wessel, 1996; Haarslev, 1998a] cannot be adequately defined in

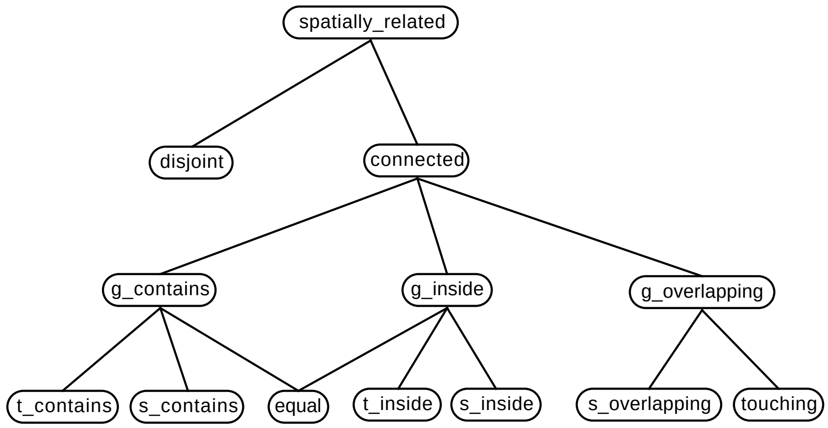

Figure 2: Subsumption hierarchy of spatial predicates.

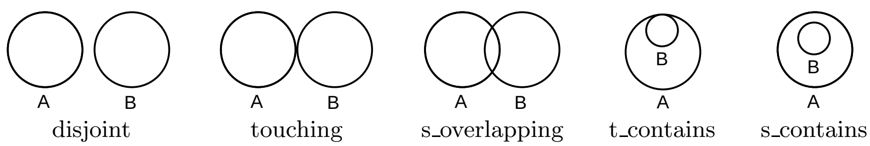

Figure 3: Spatial relations between A and B

$\mathcal{ALC}(\mathcal{D})$. With $\mathcal{ALC}(\mathcal{D})$ one can only define concepts describing individuals that are dependent on properties reachable via feature chains originating from a single individual and expressed with concrete predicates. It is not possible to define roles specified by predicates that express relationships between individuals. Therefore, TBox and ABox reasoning in $\mathcal{ALC}(\mathcal{D})$ is *spatially incomplete*[4] for the intended semantics of spatial relations.

Using the new extended concrete domain we illustrate the spatial incompleteness of $\mathcal{ALC}(\mathcal{D})$ with an example demonstrating unintended models. The unintended models are possible because $\mathcal{ALC}(\mathcal{D})$ cannot appropriately capture the semantics of spatial relations. Note that in contrast to the other sections we have to consider in this subsection the roles is_touching, is_connected, and is_g_inside as primitive since $\mathcal{ALC}(\mathcal{D})$ has no means to express defined roles, e.g. $\mathsf{has\_area}(\mathsf{i}_1, \mathsf{r}_1) \wedge \mathsf{has\_area}(\mathsf{i}_2, \mathsf{r}_3) \wedge \mathsf{touching}(\mathsf{r}_1, \mathsf{r}_2) \not\Rightarrow \mathsf{is\_touching}(\mathsf{i}_1, \mathsf{i}_2)$.

[4]We use the informal term *spatially incomplete* for referring to *unintended models* that cannot be recognized by any $\mathcal{ALC}(\mathcal{D})$ reasoner. Of course, TBox and ABox reasoning in $\mathcal{ALC}(\mathcal{D})$ is sound and complete with respect to the semantics of $\mathcal{ALC}(\mathcal{D})$.

$\textbf{fishing\_cottage} \doteq \text{cottage} \sqcap \exists\, \text{is\_touching} \,.\, \text{river}$

$\textbf{mosquito\_free\_forest} \doteq \text{forest} \sqcap \forall\, \text{is\_connected} \,.\, \neg\text{river}$

$\textbf{paradise\_cottage} \doteq$

$\text{fishing\_cottage} \sqcap \exists\, \text{is\_g\_inside} \,.\, \text{forest} \sqcap \forall\, \text{is\_g\_inside} \,.\, \text{mosquito\_free\_forest}$

We define a paradise cottage as a fishing cottage located in a mosquito-free forest, i.e. the forest is not spatially connected with a river. However, a fishing cottage is defined as a cottage that touches a river. It follows that the forest containing a fishing cottage must also be spatially connected with this river. Obviously, the paradise cottage is only a dream that cannot exist in the real world. This is due to the intended semantics of the underlying spatial relations:

> A situation where a region r_1 (cottage) is *g_inside* another region r_2 (forest) and this region r_1 is also *touching* a third region r_3 (river) implies that r_2 is *connected* to r_3, i.e.
> $\text{g\_inside}(r_1, r_2) \wedge \text{touching}(r_1, r_3) \Rightarrow \text{connected}(r_2, r_3)$

Thus, the concept paradise_cottage should be recognized as incoherent. This is not possible in $\mathcal{ALC}(\mathcal{D})$ due to the absence of defined roles capturing the semantics of the spatial predicates. With $\mathcal{ALC}(\mathcal{D})$ these roles can only be defined as primitive, i.e. they cannot interact with one another.

This deficiency also holds for ABox reasoning. The following assertions describe a spatial constellation in correspondence with the TBox defined above.

$\text{c} : \text{normal\_cottage},\ (\text{c}, 60) : \text{has\_space},\ (\text{c}, \text{S}_\text{c}) : \text{has\_area}$

$\text{r} : \text{river},\ (\text{r}, \text{S}_\text{r}) : \text{has\_area},\ (\text{c}, \text{r}) : \text{is\_touching}$

$\text{f} : \text{forest},\ (\text{f}, \text{S}_\text{f}) : \text{has\_area},\ (\text{c}, \text{f}) : \text{is\_g\_inside}$

If we pose the query "Is the forest f spatially connected with the river r?" using the TBox and ABox declarations as defined above, any $\mathcal{ALC}(\mathcal{D})$ reasoner would correctly answer *no*. Even adding the assertion

$\text{f} : \exists\, \text{is\_touching} \,.\, \text{river}$

to the previous ABox would *not* cause a contradiction for the individual f because the reasoner has no knowledge about the interaction between is_touching and is_connected.

3.3 Reasoning with $\mathcal{ALCRP}(\mathcal{D})$

A solution to the 'unintended model' problem is possible in $\mathcal{ALCRP}(\mathcal{D})$ with *defined* roles using the *role-forming predicate restriction*. With this operator one can define roles that properly reflect spatial relationships. In the following we demonstrate with the same scenario as above that these unintended models cannot exist with $\mathcal{ALCRP}(\mathcal{D})$. Note that the roles is_g_inside et cetera are now defined via spatial predicates.

cottage $\sqsubseteq$ building

small_cottage $\doteq$ cottage $\sqcap \exists$ has_space . $\lambda_{\mathcal{R}} x . (x < 30)$

normal_cottage $\doteq$ cottage $\sqcap \exists$ has_space . $\lambda_{\mathcal{R}} x . (x \geq 30 \wedge x < 70)$

spacious_cottage $\doteq$ cottage $\sqcap \exists$ has_space . $\lambda_{\mathcal{R}} x . (x \geq 70)$

is_g_inside $\doteq$ $\exists$ (has_area)(has_area) . g_inside

cottage_in_forest $\doteq$ cottage $\sqcap \exists$ is_g_inside . forest

affordable_cottage $\doteq$ cottage_in_forest $\sqcap$ (small_cottage $\sqcup$ normal_cottage)

expensive_cottage $\doteq$ cottage_in_forest $\sqcap$ spacious_cottage

is_touching $\doteq$ $\exists$ (has_area)(has_area) . touching

fishing_cottage $\doteq$ cottage $\sqcap \exists$ is_touching . river

is_connected $\doteq$ $\exists$ (has_area)(has_area) . connected

mosquito_free_forest $\doteq$ forest $\sqcap \forall$ is_connected . $\neg$river

paradise_cottage $\doteq$

fishing_cottage $\sqcap \exists$ is_g_inside . forest $\sqcap \forall$ is_g_inside . mosquito_free_forest

Any reasoner for $\mathcal{ALCRP}(\mathcal{D})$ will recognize that paradise_cottage is incoherent, i.e. an individual can never be a member of this concept. This is due to the spatial inference that a mosquito-free forest has to be connected with a river since it contains a cottage that is touching a river. However, the definition of mosquito-free forest requires that anything that is spatially connected must not be a river. This is an obvious contradiction in the TBox.

c : cottage, (c, 60) : has_space, (c, S_c) : has_area

r : river, (r, S_r) : has_area, (S_c, S_r) : touching

f : forest, (f, S_f) : has_area, (S_c, S_f) : g_inside

Using a corresponding ABox as shown above, the reasoner will also correctly answer the query "Is the forest f spatially connected with the river r?" with *yes*, i.e. adding the assertion

f : $\neg\exists$ is_connected . river

to the following ABox would cause a contradiction for f and thus validate the query. The contradiction with the negated query can be explained with the same spatial inference as the one used in the TBox. Adding for f the fact $\neg\exists\,\mathsf{is\_connected}\,.\,\mathsf{river}$ (that is equivalent to $\forall\,\mathsf{is\_connected}\,.\,\neg\mathsf{river}$) will cause a clash with the entailed relationship $(\mathsf{f},\mathsf{r}) : \mathsf{is\_connected}$ since r is asserted as a member of river.

4 Application to Visual Query Languages

The previous sections motivated the development of the description logic $\mathcal{ALCRP}(\mathcal{D})$ and demonstrated its usefulness for spatial reasoning with visual representations. We introduced semantic entities such as buildings, cottages, forests, rivers, etc. These entities are suitable candidates for elements of visual spatial query languages. This is motivated by the development of the VISCO system providing a visual, sketch-based query language for GIS [Haarslev and Wessel, 1997; Wessel and Haarslev, 1998]. In VISCO we assume that basic map objects are predefined in a GIS. Furthermore, spatial areas are defined by polygons. Map elements (e.g. polylines, polygons) are annotated with labels such as “forest”, “building”, “river” etc. that directly correspond to the semantic entities characterized above.

VISCO’s queries are basically considered as spatial constellations based on topological and geometric relationships. The syntax of VISCO’s visual query language can be easily specified with our description logic framework introduced in [Haarslev, 1995]. This was extensively exemplified for various diagrammatic languages in [Haarslev, 1998a]. Therefore, we omit any discussions about specifying the syntax of visual spatial query languages and refer the reader to [Haarslev, 1998a]. However, an important contribution of this article is the first attempt to specify the semantics of visual spatial query languages with the help of a decidable logic such as $\mathcal{ALCRP}(\mathcal{D})$ offering means to correctly specify qualitative spatial relations and to ground them on computational geometry.

The application scenario of VISCO for querying a GIS is as follows. Instead of textually writing a complicated SQL query, a user simply draws (sketches) a constellation of spatial entities that resemble the intended constellation of interest. The user also has to assign the intended semantics to drawing elements (e.g. this polygon represents a forest, etc) using the basic vocabulary provided by the GIS. The spatial parser of VISCO analyzes the input, a drawing of a spatial constellation. In case of its syntactic correctness, the parser creates an abstract syntax tree which is the source for the query translation. The semantics of VISCO relies on the abstract syntax and is specified within a typed lambda calculus. A translation process generates internal data structures that correspond to the semantics of query descriptions. The semantic query description is subject to a query optimizer that feeds its result to the spatial database engine of VISCO [Wessel and Haarslev, 1998]. The query matches are collected and visualized with VISCO’s query inspector. Afterwards, the query might be modified and refined for further execution.

In the following section we discuss the application of our formalism to query understanding and processing but do not consider the actual integration into

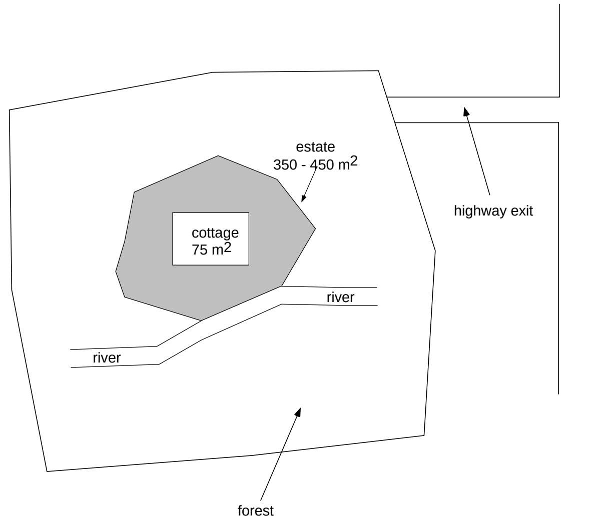

Figure 4: Spatial sketch for first query

the VISCO system for sake of clarity. However, we can imagine a successor of VISCO incorporating the ideas presented in this article.

4.1 Reasoning about Visual Spatial Queries

We demonstrate the applicability of $\mathcal{ALCRP}(\mathcal{D})$ by extending our GIS example. We imagine a scenario for a GIS query where somebody is planning to acquire a nice cottage. We assume the existence of a GIS offering information about suitable areas in the countryside. The person intends to use the cottage for weekends and short holidays. However, the potential buyer is living and working in a major city and is only interested in real estate that can be conveniently reached via a highway. A short travel distance by car from a highway exit to the cottage is the first precondition. Furthermore, the cottage should be located in a forest with a river in the immediate vicinity. The buyer and its family also want a cottage that provides at least 75 m^2 floor space. The estate itself should have about 400 m^2. Having these requirements in mind we can sketch a query (see Figure 4) reflecting the topological and geometric constraints.[5] A parser can translate the sketch to an equivalent semantic description on the basis of a taxonomy containing concept descriptions for the spatial vocabulary of this GIS domain. For Figure 4 we get the following Abox $\mathcal{A}_0$.

[5]We are aware of the scaling problems with drawings and offer with VISCO's query language first solutions. However, in this article we deliberately ignore these problems.

$\mathsf{c} : \mathsf{cottage} \sqcap \exists\, \mathsf{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x > 75),\ (\mathsf{c}, \mathsf{e}) : \mathsf{is\_g\_inside}$

$\mathsf{e} : \mathsf{estate\_area} \sqcap \exists\, \mathsf{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x > 350 \wedge x < 450)$

$\mathsf{r} : \mathsf{river},\ (\mathsf{r}, \mathsf{e}) : \mathsf{is\_touching}$

$\mathsf{f} : \mathsf{forest},\ (\mathsf{e}, \mathsf{f}) : \mathsf{is\_g\_inside}$

$\mathsf{h} : \mathsf{highway\_exit},\ (\mathsf{h}, \mathsf{f}) : \mathsf{is\_touching}$

We use concept and role expressions as defined in the previous TBoxes. The cottage is described by the individual c with a predicate-exists restriction asserting a floor space of more than 75 m^2. The cottage c has to be inside of an estate with a size between 350 and 450 m^2. As a simplification we assume that the river r has to touch the estate e that is inside of a forest f. The short driving distance from the highway exit h to the cottage c is represented by the condition that h has to touch the borderline of the forest f.[6] For sake of simplicity we deliberately abstracted away the distance constraints. Of course, it is also possible to express these constraints with the domain $\mathcal{R}$.

Additionally, we assume the following new or revised concept definitions ($\top_{\mathcal{R}}$ names the predicate required for testing the membership in the domain $\mathcal{R}$ for a concrete individual x, see also the definition of concrete domains in Section 2.1).

$\textbf{estate} \sqsubseteq \mathsf{spatial\_area} \sqcap \exists\, \mathsf{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (\top_{\mathcal{R}}(x))$

$\textbf{estate\_in\_forest} \doteq \mathsf{estate} \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{forest}$

$\textbf{cottage\_in\_forest} \doteq \mathsf{cottage} \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{estate\_in\_forest}$

$\textbf{fishing\_cottage} \doteq \mathsf{cottage} \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, (\mathsf{estate} \sqcap \exists\, \mathsf{is\_touching} \,.\, \mathsf{river})$

The realizing component[7] of the $\mathcal{ALCRP}(\mathcal{D})$ reasoner will compute the following most specific subsuming concepts (also referred to as *parents*) of the cottage c: expensive_cottage and fishing_cottage. The parents of the estate e will be estate_in_forest. The other individuals r, f, h keep their asserted concepts as parents.

With the help of an abstraction process (see [Hollunder, 1994]) we can replace Abox $\mathcal{A}_0$ by an Abox $\mathcal{A}_1$ containing a single assertion for c with the synthesized concept description $\mathsf{cottage}_{\mathsf{c}_1}$. The other two concept definitions are only used to enhance the readability of $\mathsf{cottage}_{\mathsf{c}_1}$.

[6] This is a simplification again since the extent of a forest can easily cause a long driving distance.

[7] See Section 2.3 for a description of basic reasoning services.

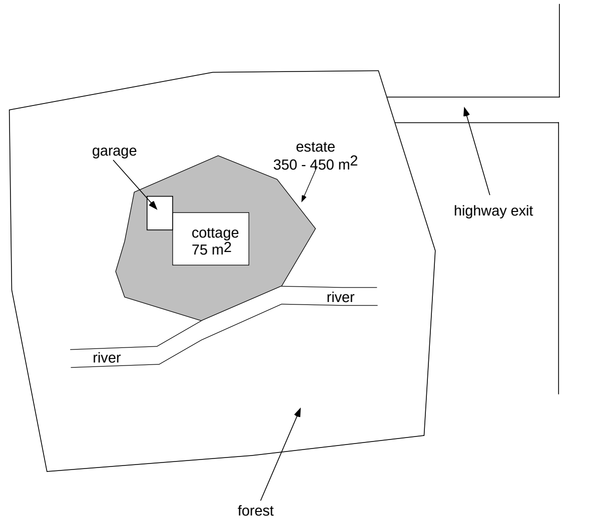

Figure 5: Spatial sketch for second query

$\mathsf{\mathbf{forest_{f_1}}} \doteq \mathsf{forest} \sqcap \exists\, \mathsf{is\_touching} \,.\, \mathsf{highway\_exit}$

$\mathsf{\mathbf{estate_{e_1}}} \doteq \mathsf{estate} \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{forest_{f_1}} \sqcap \exists\, \mathsf{is\_touching} \,.\, \mathsf{river}$

$\mathsf{\mathbf{cottage_{c_1}}} \doteq \mathsf{cottage} \sqcap \exists\, \mathsf{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x > 75) \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{estate_{e_1}}$

The revised ABox $\mathcal{A}_1$ now consists only of the assertion $\mathsf{c} : \mathsf{cottage_{c_1}}$. The newly created concept $\mathsf{cottage_{c_1}}$ is classified by the reasoner and integrated into the concept taxonomy. The semantic validity of this query is automatically verified during classification, i.e. to check whether the concept is coherent (see Section 2.3). For instance, if the forest f were required to be 'mosquito-free' (see above), the $\mathcal{ALCRP}(\mathcal{D})$ reasoner would immediately recognize the incoherence of $\mathsf{cottage_{c_1}}$. This information could be used by the spatial parser for generating an explanation to the user and for identifying the source of the contradiction.

Let us assume that the executed query $\mathsf{c} : \mathsf{cottage_{c_1}}$ returns more than 100 matches. The next step for the user might be to refine the query by adding more constraints.[8] One could add more requirements to the estate, e.g. we ask for a garage connected to the cottage. The extended sketch (see Figure 5) corresponds to the ABox $\mathcal{A}_2$ that results from adding to ABox $\mathcal{A}_0$ the following new assertions.

$\mathsf{g} : \mathsf{garage},\ (\mathsf{c}, \mathsf{g}) : \mathsf{is\_touching}$

[8] Of course, one of the most important criteria is the price of the estate. This is neglected due to the non-spatial nature of this part of the query.

The abstraction process reduces ABox $\mathcal{A}_2$ to ABox $\mathcal{A}_3$ consisting only of the assertion $\mathsf{c} : \mathsf{cottage}_{\mathsf{c}_2}$ using the following synthesized concept description.

$$\mathbf{cottage}_{\mathbf{c}_2} \doteq$$
$$\mathsf{cottage} \sqcap \exists\, \mathsf{has\_space} \,.\, \lambda_{\mathcal{R}} x \,.\, (x > 75) \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{estate}_{\mathsf{e}_1} \sqcap$$
$$\exists\, \mathsf{is\_touching} \,.\, \mathsf{garage}$$

The $\mathcal{ALCRP}(\mathcal{D})$ reasoner recognizes the relationship in the taxonomy that $\mathsf{cottage}_{\mathsf{c}_1}$ subsumes $\mathsf{cottage}_{\mathsf{c}_2}$. It can be rewritten as $\mathsf{cottage}_{\mathsf{c}_3}$ that even textually demonstrates the subsumption relationship.

$$\mathbf{cottage}_{\mathbf{c}_3} \doteq \mathsf{cottage}_{\mathsf{c}_1} \sqcap \exists\, \mathsf{is\_touching} \,.\, \mathsf{garage}$$

For executing the refined query the optimizer can benefit from the detected query subsumption and reduce the search space to the set of query matches already computed for ABox $\mathcal{A}_1$. Note that these query matches are members of the concept $\mathsf{cottage}_{\mathsf{c}_1}$. This type of query optimization is an important aspect in applying description logics to database theory (see [Borgida, 1995] for an introduction to these topics).

The benefits of computing a concept subsumption taxonomy can be even more subtle. Imagine a query from another user looking for a cottage located in a forest that is connected to a river. The ABox $\mathcal{A}_4$ derived from the sketch might be structured as follows.

$$\mathsf{c} : \mathsf{cottage},\ \mathsf{e} : \mathsf{estate\_area},\ (\mathsf{c}, \mathsf{e}) : \mathsf{is\_g\_inside}$$
$$\mathsf{r} : \mathsf{river},\ \mathsf{f} : \mathsf{forest},\ (\mathsf{f}, \mathsf{r}) : \mathsf{is\_connected},\ (\mathsf{e}, \mathsf{f}) : \mathsf{is\_g\_inside}$$

The abstraction process creates the following concept definitions.

$$\mathbf{forest}_{\mathbf{f}_2} \doteq \mathsf{forest} \sqcap \exists\, \mathsf{is\_connected} \,.\, \mathsf{river}$$
$$\mathbf{estate}_{\mathbf{e}_2} \doteq \mathsf{estate} \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{forest}_{\mathsf{f}_2}$$
$$\mathbf{cottage}_{\mathbf{c}_4} \doteq \mathsf{cottage} \sqcap \exists\, \mathsf{is\_g\_inside} \,.\, \mathsf{estate}_{\mathsf{e}_2}$$

The resulting ABox $\mathcal{A}_4$ consists only of the assertion $\mathsf{c} : \mathsf{cottage}_{\mathsf{c}_4}$. It turns out that the concept $\mathsf{cottage}_{\mathsf{c}_4}$ subsumes the other concepts $\mathsf{cottage}_{\mathsf{c}_i}$ although the concept descriptions are textually different. This is a rather complex proof also based on the spatial inference already explained above: $\mathsf{g\_inside}(\mathsf{e}, \mathsf{f}) \wedge \mathsf{touching}(\mathsf{e}, \mathsf{r}) \Rightarrow \mathsf{connected}(\mathsf{f}, \mathsf{r})$

4.2 Open Problems with ABox Abstraction

The abstraction process works rather well for ABoxes containing no joins or cycles, i.e. the same individual is a filler of several roles or even related to itself through a cycle of role assertions. If joins or cycles are present in an ABox, it depends on the expressiveness of the description logic whether an ABox can be reduced to a single concept membership assertion. For instance, joins can be expressed by restricting the number of possible role fillers or by equality restrictions for feature fillers. As mentioned above, other DLs also support the definition of cyclic concepts that might be required to fully reduce some ABoxes. Due to unknown decidability results $\mathcal{ALCRP}(\mathcal{D})$ currently does not allow cyclic concepts or number restrictions. Therefore, in case of ABoxes with joins or cycles, we can only partially reduce these ABoxes. However, the reasoning with $\mathcal{ALCRP}(\mathcal{D})$ as describe above is still valid and usable for query processing. Only the subsumption between ABox queries requires a more sophisticated approach, e.g. by additionally utilizing graph matching techniques for Aboxes. This is work in progress and will be subject to future publications.

5 Related Work

In the following we discuss various approaches related to our research. We start with a general overview of research on visual language theory. Closely related work is discussed in more detail. Afterwards, we briefly review some aspects of other visual (spatial) query languages concerning the semantics of queries. We close this section with a review of research on other spatial logics.

There exist many approaches to specifying syntax (and to some degree semantics) of visual languages. Mostly, these are based on extensions of string grammar formalisms. A complete and recent overview is out of scope of this article. However, we like to mention a few approaches: generalizations of attributed grammars (e.g. picture layout grammars [Golin, 1991]), positional grammars (e.g. [Costagliola et al., 1991]), and graph grammars (e.g. [Göttler, 1989; Najork and Kaplan, 1993; Rekers and Schürr, 1995]). Other approaches closely related to this one use (constraint) logic or relational formalisms (e.g. [Crimi et al., 1991; Helm and Marriott, 1991; Meyer, 1992; Wittenburg et al., 1991; Wittenburg, 1993; Marriott, 1994]) to represent spatial relationships. Wittenburg [Wittenburg, 1993] reports that some grammar approaches have limitations such as no arbitrary ordering of input is supported, only special relations are allowed, connected graphs are required, no bottom-up parsing is provided, no ambiguous grammars, etc. These limitations are sometimes unacceptable for particular application domains. We refer to [Marriott et al., 1998] for an extensive review of related work.

Helm and Marriott [Helm and Marriott, 1991] developed a declarative specification and semantics for VLs. It is based on definite clause logic and implemented with the help of constraint logic programming. Marriott's recent approach is based on these ideas but utilizes constraint multiset grammars [Marriott, 1994].

This is further explored in [Marriott and Meyer, 1997; Marriott and Meyer, 1998a] where a classification of visual languages by grammar hierarchies is presented on the basis of *copy-restricted* constraint multiset grammars. The decidability versus expressivity trade-off is used to shape the hierarchy in analogy to the Chomsky hierarchy in formal language theory. An advantage of our approach is the taxonomic hierarchy of concept definitions and the capabilities to reason about these specifications and their subsumption relationships. We believe that constraint multiset grammars and the $\mathcal{ALCRP}(\mathcal{D})$ approach are getting quite close to each other since constraint specification and solving is now available in $\mathcal{ALCRP}(\mathcal{D})$ via concrete domains. However, this has to be more thoroughly analyzed.

Meyer's recent work [Meyer, 1997] presents *picture logic*, a visual language for the specification of diagrams and diagram transformations. Picture logic is based on constraint logic programming handling constraints over real intervals. $\mathcal{ALCRP}(\mathcal{D})$ can also be instantiated with a similar concrete domain and it might be possible to also specify transformations over time if we utilize Allen's interval logic [Allen, 1983]. However, this is currently an open issue.

Cohn and Gooday [Cohn and Gooday, 1994; Gooday and Cohn, 1996] applied the 'Region-Connection-Calculus' (RCC theory) to the VL domain and developed formal static and procedural semantics for Pictorial Janus.[9] However, their specifications use the first-order theory of RCC that is known to be undecidable (see [Cohn, 1997]). As far as we know, they do not support the graphical construction (e.g. editing and parsing) of diagrammatic representations or mechanical verification processes (e.g. consistency checking of specifications). Of course, due to the undecidability of RCC's first order theory a decision procedure for consistency checking cannot exist.

Another approach to reasoning with pictorial concepts is based on a different, type-theoretic framework [Wang and Lee, 1993a; Wang and Lee, 1993b; Wang et al., 1995]. An important distinction is that our theory is more expressive with respect to concept definitions. For instance, in [Wang and Lee, 1993a] the authors suggest to extend their type-theoretic approach by notions such as parameterization for construction of generic concepts and type dependency for describing pictures consisting of parts of other pictures. Our DL theory already handles the intended effects of parameterization and type dependency since its reasoning component automatically maintains a taxonomy of subsuming concept definitions which may share common subparts.

The logical status of (extended) Venn diagrams is analyzed by Shin [Shin, 1994]. Shin gives axioms for well-formed Venn diagrams and a semantics using first-order predicate logic. However, Shin's formal account is not based on a spatial logic and not supported by reasoning mechanisms comparable to DL systems.

The understanding of diagrams can be also considered as a subproblem of image interpretation and is related to similar approaches in this area. The first treatment in this area was the MAPSEE approach [Reiter and Mackworth, 1989]. It

[9] For information on Pictorial Janus see [Kahn and Saraswat, 1990; Kahn et al., 1991].

is based on specifications with full first-order predicate logic.

In comparison to other logic-based approaches, we argue that the DL notation of $\mathcal{ALCRP}(\mathcal{D})$–featuring object-centered concept and role definitions and the integration of concrete domains– is much more suitable for human and even mechanical inspection (due to decidability results). This is an important issue since theories about VLs are still designed by humans.

Our experience with VISCO [Haarslev and Wessel, 1997; Wessel and Haarslev, 1998] has motivated the research presented in this article. VISCO can be classified as a visual query system for spatial information systems that uses 'sketched' queries combined with deductive reasoning. A recent and complete survey on visual query systems for database systems handling conventional data can be found in [Catarci et al., 1997]. Other relevant work [Meyer, 1994; Egenhofer, 1997] reviews especially visual query system for spatial information systems. A related approach that also uses spatial relations [Del Bimbo et al., 1994] deals with symbolic descriptions and retrieval in image databases. Another approach deals with pictorial query specification for spatially referenced image databases [Soffer and Samet, 1998]. We refer to [Haarslev and Wessel, 1997] for a review of the four approaches [Meyer, 1994; Calcinelli and Mainguenaud, 1994; Lee and Chin, 1995; Egenhofer, 1997] that come closest to the ideas and concepts behind VISCO. [Meyer, 1994] also gives formal semantics for visual spatial queries using a mapping to the Datalog language but there exists no formalization of topological relations or conceptual knowledge. To the best of our knowledge there exists no other approach or (visual) spatial query language addressing the semantics of spatial queries and their subsumption using a spatial logic such as $\mathcal{ALCRP}(\mathcal{D})$.

For formalizing reasoning about spatial structures themselves many specific approaches and have been published (see e.g. [Stock, 1997] for an overview). Ignoring decidability, Borgo et al. [Borgo et al., 1996] have developed a first order theory of space which formalizes different aspects such as mereology etc. An algebraic (but still undecidable) theory about space has been proposed by [Pratt and Lemon, 1997; Pratt and Schoop, 1997]. Research on the RCC theory is also summarized in [Cohn et al., 1997]. While first axiomatizations used first-order logic, recently, the spatial relations used in RCC have been defined in terms of intuitionistic logic and propositional modal logic [Bennett, 1995]. Although qualitative reasoning with RCC can be used in many applications, in GIS also *conceptual* knowledge combined with qualitative relations has to be considered.

6 Conclusion and Ongoing Work

The formalism presented in this article can be used to define the semantics of visual spatial queries and to reason about query validity and subsumption. We would like to emphasize that our approach has no restrictions about the ordering of input and the type of allowed relations provided the corresponding concrete domain is admissible. We do not rely on special parsing techniques because our approach is purely declarative. We can even deal with ambiguous specifications since our DL reasoner can compute every model satisfying the specifications. This is addressed in future work with the help of default reasoning. A problem with our approach could be the worst-case time complexity of the underlying classification algorithms. However, almost every logical or constraint-oriented approach with an interesting expressiveness has to deal with tractability and decidability. It is also important to note that complexity issues of DLs are very well understood and analyzed. Based on recent findings [Horrocks, 1997; Horrocks, 1998] about optimizing DL reasoners for the average case we are currently developing an optimized $\mathcal{ALCRP}(\mathcal{D})$ reasoner [Haarslev et al., 1998b]. We further plan to integrate this reasoner and other DL reasoners into successors of the VISCO system [Haarslev and Wessel, 1997; Wessel and Haarslev, 1998] and the GenEd system [Haarslev and Wessel, 1996].

Acknowledgements

This article is an extended and revised version of [Haarslev, 1998b] presented at the workshop on *Formalizing Reasoning with Visual and Diagrammatic Representations.* We are thankful for the helpful comments from the anonymous referees of this workshop and from the JVLC reviewers.

We gratefully acknowledge the work of our colleagues Ralf Möller and Carsten Lutz, co-authors of $\mathcal{ALCRP}(\mathcal{D})$, Michael Wessel, co-author and developer of VISCO, and Anni-Yasmin Turhan, developer of our $\mathcal{ALC}(\mathcal{D})$ and $\mathcal{ALCRP}(\mathcal{D})$ implementations.

References

Allen, J. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843.

Baader, F. and Hanschke, P. (1991a). A scheme for integrating concrete domains into concept languages. In *Twelfth International Conference on Artificial Intelligence, Darling Harbour, Sydney, Australia, Aug. 24-30, 1991*, pages 452–457.

Baader, F. and Hanschke, P. (1991b). A scheme for integrating concrete domains into concept languages. Technical Report DFKI-RR-91-10, German Center for AI (DFKI).

Bennett, B. (1995). Modal logics for qualitative spatial reasoning. *Bull. of the IGPL*, 3:1–22.

Borgida, A. (1995). Description logics in data management. *IEEE Transactions on Knowledge and Data Engineering*, 7(5):671–682.

Borgo, S., Guarino, N., and Masolo, C. (1996). A pointless theory of space based on strong connection and congruence. In Aiello, L., Doyle, J., and Shapiro, S., editors, *Fifth International Conference on Principles of Knowledge Representation, Cambridge, Mass., Nov. 5-8, 1996*, pages 220–229.

Brachman, R., McGuinness, D., Patel-Schneider, P., Reswnick, L., and Borgida, A. (1991). Living with CLASSIC: When and how to use a KL-ONE-like language. In Sowa, J., editor, *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, pages 401–456. Morgan Kaufmann Publishers, San Mateo.

Brachman, R. and Schmolze, J. (1985). An overview of the KL-ONE knowledge representation system. *Cognitive Science*, pages 171–216.

Calcinelli, D. and Mainguenaud, M. (1994). Cigales, a visual query language for a geographical information system: the user interface. *Journal of Visual Languages and Computing*, 5(2):113–132.

Catarci, T., Costabile, M., Levialdi, S., and Batini, C. (1997). Visual query systems for databases: A survey. *Journal of Visual Languages and Computing*, 8(2):215–260.

Cohn, A. (1997). Qualitative spatial representation and reasoning techniques. In Brewka, G., Habel, C., and Nebel, B., editors, *Proceedings, KI-97: Advances in Artificial Intelligence, 21st Annual German Conference on Artificial Intelligence, Freiburg, Germany*, volume 1303 of *Lecture Notes in Artificial Intelligence*, pages 1–30. Springer Verlag, Berlin.

Cohn, A., Bennett, B., Gooday, J., and Gotts, N. (1997). Representing and reasoning with qualitative spatial relations. In [Stock, 1997], pages 97–134.

Cohn, A. and Gooday, J. (1994). Defining the syntax and the semantics of a visual programming language in a spatial logic. In *AAAI-94, Spatial and Temporal Reasoning Workshop*.

Cohn, T., Schubert, L., and Shapiro, S., editors (1998). *Proceedings of Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR'98), Trento, Italy, June 2-5, 1998*.

Costagliola, G., Tomita, M., and Chang, S. (1991). A generalized parser for 2-D languages. In *1991 IEEE Workshop on Visual Languages, Kobe, Japan, Oct. 8-11*, pages 98–104. IEEE Computer Society Press, Los Alamitos.

Crimi, C., Guercio, A., Nota, G., Pacini, G., Tortora, G., and Tucci, M. (1991). Relation grammars and their application to multi-dimensional languages. *Journal of Visual Languages and Computing*, 2(4):333–346.

Del Bimbo, A., Vicario, E., and Zingoni, D. (1994). A spatial logic for symbolic description of image contents. *Journal of Visual Languages and Computing*, 5(3):267–286.

Egenhofer, M. (1991). Reasoning about binary topological relations. In Günther, O. and Schek, H.-J., editors, *Advances in Spatial Databases, Second Symposium, SSD'91, Zurich, Aug. 28-30, 1991*, volume 525 of *Lecture Notes in Computer Science*, pages 143–160. Springer Verlag, Berlin.

Egenhofer, M. (1997). Query processing in spatial-query-by-sketch. *Journal of Visual Languages and Computing*, 8(4):403–424.

Golin, E. (1991). Parsing visual languages with picture layout grammars. *Journal of Visual Languages and Computing*, 2(4):371–393.

Gooday, J. and Cohn, A. (1996). Using spatial logic to describe visual programming languages. *Artificial Intelligence Review*, 10:171–186.

Göttler, H. (1989). Graph grammars, a new paradigm for implementing visual languages. In *Rewriting Techniques and Applications, 3rd International Conference, RTA-89, 3-5 April 1989, Chapel Hill, NC*, pages 152–166. Springer Verlag, Berlin.

Haarslev, V. (1995). Formal semantics of visual languages using spatial reasoning. In [VL'95, 1995], pages 156–163.

Haarslev, V. (1998a). A fully formalized theory for describing visual notations. In [Marriott and Meyer, 1998b], pages 261–292.

Haarslev, V. (1998b). A logic-based formalism for reasoning about visual representations (extended abstract). In *Proceedings, AAAI Workshop on Formalizing Reasoning with Visual and Diagrammatic Representations, AAAI Fall Symposium Series 1998, October 23-25, Orlando, Florida/USA. Technical Report FS-98-04*, pages 57–66. AAAI Press.

Haarslev, V., Lutz, C., and Möller, R. (1998a). Foundations of spatioterminological reasoning with description logics. In [Cohn et al., 1998], pages 112–123.

Haarslev, V., Lutz, C., and Möller, R. (1999). A description logic with concrete domains and a role-forming predicate operator. *Journal of Logic and Computation*, 9(3):1–34.

Haarslev, V. and Möller, R. (1997). SBox: A qualitative spatial reasoner—progress report. In Ironi, L., editor, *11th International Workshop on Qualitative Reasoning, Cortona, Tuscany, Italy, June 3-6, 1997, Pubblicazioni N. 1036, Istituto di Analisi Numerica C.N.R. Pavia (Italy)*, pages 105–113.

Haarslev, V., Möller, R., and Turhan, A.-Y. (1998b). Implementing an $\mathcal{ALCRP}(\mathcal{D})$ ABox reasoner: Progress report. In Franconi et al., E., editor, *Proceedings of the International Workshop on Description Logics (DL'98), June 6-8, 1998, Trento, Italy*, pages 82–86.

Haarslev, V. and Wessel, M. (1996). GenEd—an editor with generic semantics for formal reasoning about visual notations. In [VL'96, 1996], pages 204–211.

Haarslev, V. and Wessel, M. (1997). Querying GIS with animated spatial sketches. In *1997 IEEE Symposium on Visual Languages, Capri, Italy, Sep. 23-26*, pages 197–204. IEEE Computer Society Press, Los Alamitos.

Helm, R. and Marriott, K. (1991). A declarative specification and semantics for visual languages. *Journal of Visual Languages and Computing*, 2(4):311–331.

Hollunder, B. (1994). *Algorithmic Foundations of Terminological Knowledge Representation Systems*. PhD thesis, University of Saarbrücken, Department of Computer Science.

Horrocks, I. (1997). *Optimising Tableaux Decision Procedures for Description Logics*. PhD thesis, University of Manchester.

Horrocks, I. (1998). Using an expressive description logic: FaCT or fiction? In [Cohn et al., 1998], pages 636–647.

Kahn, K. and Saraswat, V. (1990). Complete visualizations of concurrent programs and their executions. In *1990 IEEE Workshop on Visual Languages, Skokie, Illinois, Oct. 4-6*, pages 7–14. IEEE Computer Society Press, Los Alamitos.

Kahn, K., Saraswat, V., and Haarslev, V. (1991). Pictorial Janus: A Completely Visual Programming Language and its Environment (in German). In Encarnacao, J., editor, *GI-Fachgespräch Programmieren multimedialer Anwendungen der GI-Jahrestagung 1991, Darmstadt, Oktober 1991*, pages 427–436. Springer Verlag, Berlin.

Lee, Y. and Chin, F. (1995). An iconic query language for topological relationships in gis. *International Journal on Geographical Information Systems*, 9(1):25–46.

Marriott, K. (1994). Constraint multiset grammars. In [VL'94, 1994], pages 118–125.

Marriott, K. and Meyer, B. (1997). On the classification of visual languages by grammar hierarchies. *Journal of Visual Languages and Computing*, 8(4):375–402.

Marriott, K. and Meyer, B. (1998a). The CCMG visual language hierarchy. In [Marriott and Meyer, 1998b], pages 129–169.

Marriott, K. and Meyer, B., editors (1998b). *Visual Language Theory*. Springer Verlag, Berlin.

Marriott, K., Meyer, B., and Wittenberg, K. (1998). A survey of visual language specification and recognition. In [Marriott and Meyer, 1998b], pages 5–85.

Meyer, B. (1992). Pictures depicting pictures: On the specification of visual languages by visual grammars. In *1992 IEEE Workshop on Visual Languages, Seattle, Washington, Sept. 15-18*, pages 41–47. IEEE Computer Society Press, Los Alamitos.

Meyer, B. (1994). Pictorial deduction in spatial information systems. In [VL'94, 1994], pages 23–30.

Meyer, B. (1997). Formalization of visual mathematical notations. In *Proceedings of AAAI Symposium on Diagrammatic Reasoning, Boston/MA*, pages 23–30.

Najork, M. and Kaplan, S. (1993). Specifying visual languages with conditional set rewrite systems. In *1993 IEEE Symposium on Visual Languages, Bergen, Norway, Aug. 24-27*, pages 12–17. IEEE Computer Society Press, Los Alamitos.

Pratt, I. and Lemon, O. (1997). Ontologies for plane, polygonal mereotopology. *Notre Dame Journal of Formal Logic*, 2(38):225–245.

Pratt, I. and Schoop, D. (1997). A complete axiom system for polygonal mereotopology of the real plane. Technical Report UMCS-97-2-2, University of Manchester, Department of Computer Science.

Randell, D., Cui, Z., and Cohn, A. (1992). A spatial logic based on regions and connections. In Nebel, B., Rich, C., and Swartout, W., editors, *Principles of Knowledge Representation and Reasoning, Cambridge, Mass., Oct. 25-29, 1992*, pages 165–176. Morgan Kaufman.

Reiter, R. and Mackworth, A. (1989). A logical framework for depiction and image interpretation. *Artificial Intelligence*, 41:125–155.

Rekers, J. and Schürr, A. (1995). A graph grammar approach to graphical parsing. In [VL'95, 1995], pages 195–202.

Shin, S.-J. (1994). *The Logical Status of Diagrams*. Cambridge University Press, Cambridge.

Soffer, A. and Samet, H. (1998). Pictorial query specification for browsing through spatially referenced image databases. *Journal of Visual Languages and Computing*, 9(6):567–596.

Stock, O., editor (1997). *Spatial and Temporal Reasoning*. Kluwer Academic Publishers, Dordrecht.

Tarski, A. (1951). *A Decision Method for Elementary Algebra and Geometry*. University of California Press, Berkeley, CA.

Turhan, A.-Y. (1998). Design and implementation of description logic provers for $\mathcal{ALC}(\mathcal{D})$ and $\mathcal{ALCRP}(\mathcal{D})$ (in german). Bachelors Thesis (Studienarbeit).

VL'94 (1994). *1994 IEEE Symposium on Visual Languages, St. Louis, Missouri, Oct. 4-7*. IEEE Computer Society Press, Los Alamitos.

VL'95 (1995). *1995 IEEE Symposium on Visual Languages, Darmstadt, Germany, Sep. 5-9*. IEEE Computer Society Press, Los Alamitos.

VL'96 (1996). *1996 IEEE Symposium on Visual Languages, Boulder, Colorado, USA, Sep. 3-6*. IEEE Computer Society Press, Los Alamitos.

Wang, D. and Lee, J. (1993a). Pictorial concepts and a concept-supporting graphical system. *Journal of Visual Languages and Computing*, 4(2):177–199.

Wang, D. and Lee, J. (1993b). Visual reasoning: its formal semantics and applications. *Journal of Visual Languages and Computing*, 4(4):327–356.

Wang, D., Lee, J., and Zeevat, H. (1995). Reasoning with diagrammatic representations. In Glasgow, J., Narayanan, N., and Chandrasekaran, B., editors, *Diagrammatic Reasoning: Cognitive and Computational Perspectives*, pages 339–393. AAAI Press / The MIT Press, Menlo Park.

Wessel, M. and Haarslev, V. (1998). VISCO: Bringing visual spatial querying to reality. In *1998 IEEE Symposium on Visual Languages, Halifax, Canada, Sep. 1-4*, pages 170–177. IEEE Computer Society Press, Los Alamitos.

Wittenburg, K. (1993). Adventures in multi-dimensional parsing: Cycles and disorders. In *1993 International Workshop on Parsing Technologies, Tilburg, Netherlands and Durbuy, Belgium, Aug. 8-10.*

Wittenburg, K., Weitzman, L., and Talley, J. (1991). Unification-based grammars and tabular parsing for graphical languages. *Journal of Visual Languages and Computing*, 2(4):347–370.

Woods, W. and Schmolze, J. (1992). The KL-ONE family. In Lehmann, F., editor, *Semantic Networks in Artificial Intelligence*, pages 133–177. Pergamon Press, Oxford.