Topological properties in ontology-based applications

M.A. Serrano, J. Gómez-Romero, M.A. Patricio, J. García, J.M. Molina
Applied Artificial Intelligence Group (GIAA)
Universidad Carlos III de Madrid, UC3M
Colmenarejo, Madrid, Spain
Email: {miguel.serrano, juan.gomez.romero, miguelangel.patricio, jesus.garcia, josemanuel.molina}@uc3m.es

Abstract—**Representation and reasoning with spatial properties is essential in several application domains where ontologies are being successfully applied; e.g., Information Fusion systems. This requires a full characterization of the semantics of relations such as adjacent, included, overlapping, etc. Nevertheless, ontologies are not expressive enough to directly support widely-use spatial or topological theories, such as the Region Connection Calculus (RCC). In addition, these properties must be properly instantiated in the ontology, which may require expensive calculations. This paper presents a practical approach to represent and reason with topological properties in ontology-based systems, as well as some optimization techniques that have been applied in a video-based Information Fusion application.**

Index Terms—**Information Fusion; Ontology-based Systems; Spatial Properties; Topological Reasoning**

I. INTRODUCTION

The representation of spatial properties with ontologies has received a notable interest in the last years. An ontological spatial knowledge model uses a formal vocabulary to describe qualitative spatial relations between the entities of the domain. Qualitative spatial representations have become more important because of their proximity to the way humans define the spatial knowledge. Abstract representations of spatial and topological properties –e.g.,'A is inside B' or 'A is above B'– are close to the natural language, and can be exploited to bridge the semantic gap between symbolic and numerical representations.

Topological properties of spatial objects can be represented with the Region Connection Calculus (RCC) [1]. RCC is a logic-based theory that allows qualitative representation of spatial properties and automatic inference of implicit knowledge from explicit assertions. Nevertheless, most Description Logics –the formalisms underpinning ontologies– are not expressive enough to capture the semantics of RCC and, when possible, it comes at a very high computational cost.

This paper proposes *Dynamic RCC*, an approach to incorporate qualitative topological relations based on the RCC theory in a previous framework for the contextual interpretation of data acquired from a visual sensor network [2]. The cornerstone of the framework is an ontological model designed according to the Joint Directors of Laboratories (JDL) fusion model [3] that represents sensor and context information in several levels –from low-level tracking data to high-level situation knowledge. Dynamic RCC solves two problems: (i) representation and reasoning with spatial properties in the ontological model; (ii) efficient instantiation and update of spatial properties of detected objects.

Dynamic RCC provides support for the representation, discovery, and maintenance of spatial relations between entities of the scene model. Essentially, Dynamic RCC defines an additional layer to the main ontology specifically aimed to store spatial entities and topological relations in an Euclidean planar linear geometric submodel. These relations are instantiated as a result of changes detected in the numerical positions of the scene objects. To do so, Dynamic RCC uses an auxiliary data structure where quantitative position values are stored. When object positions change, Dynamic RCC recalculates the geospatial arrangement of the entities of the auxiliary data structure, and updates the qualitative geometric submodel by translating the obtained results into symbolic topological relations.

It is worth to notice that in order to keep the spatial relationships updated, it is necessary to perform a pairwise comparison between all the scene objects instantiated at a given instant. This process, which will be frequently executed, has a quadratic complexity, and may be unacceptable in several applications –as a matter of fact, it is unaffordable in video-based object tracking, where almost real time results are usually required. To solve this problem, Dynamic RCC creates a partition of the space in the auxiliary data structure in order to reduce the number of checks by comparing only close objects. To illustrate the functioning of Dynamic RCC, a case study based on a multi-camera sport scenario is presented.

The paper is organized as follows. Section 2 reviews related approaches that apply topological theories into ontology-based applications. Section 3 presents the overall architecture of the ontology-based computer vision framework and the Dynamic RCC extension. An explanation of the implementation is given in Section 4, where the architecture to manage dynamic topological relations is showed. Section 5 depicts an example to detect interesting situations in sport videos by applying this approach. Last but not least, Section 6 summarizes the conclusions obtained and present some prospective directions for future work.

II. Related Works

There are several research works in the literature that study the cognitive aspects of Euclidean spatial properties –e.g., topology, direction, distance–, as well as formal theories that focus on the symbolic representation of their semantics and the features of related reasoning procedures –e.g., decidability.

The first modern formalizations are due to Clarke [4], [5]. These approaches are based on the extension of the basic connection relationship and the application of logical theories to obtain additional well-defined relations. RCC is one of this axiomatizations made in first order logic [1]. The basic RCC theory assumes just one primitive dyadic relation $C(x, y)$ –read as 'x connects with y'–, where x and y denote spatial regions. This relation is reflexive and symmetric. Many different subsets of relations extending the connection relation have been defined in the context of the RCC theory. The most popular is a set of eight relations called RCC-8, since it can be encoded in propositional modal logic [6], and therefore it is decidable. An alternative approach to RCC is the 9-intersection [7], which defines nine binary relations including exterior, interior, and boundary relations between regions. Unfortunately, the 9-intersection has not proved to be decidable.

Not surprisingly, these factors have favoured the use of RCC-8 in ontology-based approaches. First attempt was from Katz and Grau [8], who carried out a translation from the feature relations of RCC-8 to the OWL language –which is based on the Description Logic $\mathcal{SHOIN}$ [9]. The main problem of this approach is that it cannot completely capture the semantics of RCC-8 because of the absence of reflexive roles in OWL, which is one of the key assumptions of the RCC relations. According to the authors, the problem could be easily solved by using an extension of the language. This approach has additional drawbacks, as described in [10]: a huge amount of TBox axioms are generated as a result of the definition of the RCC-8 roles and the axioms specifying the non-emptiness of some regions.

Next version of the language, OWL 2 –based on the Description Logic $\mathcal{SROIQ}$ [11]–, includes reflexive roles. Grütter and Bauer-Messme propose in [12], [13] a translation of RCC-8 into OWL 2. This approach faced new problems. For example, OWL 2 does not allow the definition of a concept as an individual, and therefore regions have to be represented as individuals. As a result, the spatial domain cannot be represented as a strict set of concepts and relations. Another issue is that OWL 2 does not support all the role inclusion axioms used in the composition tables needed for the RCC reasoning. According to [14], RCC-8 also requires role negation, conjunction, and disjunction, as well as complex role inclusion axioms. Using $\mathcal{SROIQB}_s$ logic [15], which adds role boolean operators to $\mathcal{SROIQ}$, some of these needs can be covered. Unfortunately, this logic does not support complex role inclusion axioms on the right hand side nor boolean role operators on complex roles.

Other proposals have faced the problem at a knowledge representation level, instead of at a formalism level. Specific components named RCCBoxes have been defined to manage spatial relationships in ontology supporting tools. These RCC-Boxes have predefined RCC relationships and composition tables, and use OWL 2 –they need support for negation roles to define a disconnected relation if none of the other relations are detected. RCCBoxes have been implemented in Pellet [10] and RACER [16] reasoners.

Geographical information systems (GIS) are one of the most common applications that require intensive use of spatial properties are. GIS have been applying ontologies during the last years in a a wide variety of applications; e.g., disaster management [17], data retrieval [18], etc. Most of them are focused on representation issues, since they use ontologies to improve interoperability between heterogeneous systems. Recently, new systems to query over spatial objects and features have appeared, though their expressiveness is quite limited; this is the case of AllegroGraph [19] and Geospatial-web [20]. A more general approach is [21], which offers a tight integration with OWL reasoning procedures and implements geometric operations supported by external libraries. To the best of our knowledge, there are no vision systems to recognize situations that use ontology-based spatial representation and reasoning.

III. System Architecture

The extended framework encompasses three main modules: the tracking module, the annotation module, and the knowledge module (see Figure 1). The inputs to the system are three: the sequence of video frames (to the tracking module), and a priori data expressed in a predefined data formalism (to the knowledge module).

The data formalism consists on a set of terminological boxes (TBoxes). Each one of them contains a hierarchical definition of computer vision real world concepts and relations at different abstraction levels. This formalism can be accompanied with predefined data, such as knowledge base assertions coming from previous analysis, and context information inserted by the user supervisor. This data is introduced as instances of the ontologies of the knowledge module (ABoxes).
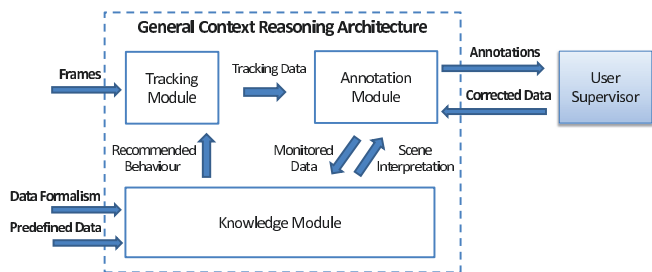

Fig. 1: Architecture of the extended framework

The video stream is managed by the tracking module. The processing in the framework starts when a new frame is captured by the video camera. Frame data is processed by the tracking system in order to detect and segment moving objects

of the scene. As a result, the tracker obtains data representing the moving entities –e.g., id number, position or size–, which is sent to the annotation module. The annotation module also receives a scene interpretation from the previous frame. Current tracking data and previous interpretations are presented as scene annotations to the a human user –e.g., objects are marked with squares and labelled when they participate in a detected activity. The user can monitor the performance of the tracker and introduce new context information through the annotation interface. Once the information is checked by the user, the annotation module sends the validated data, encompassing tracking and user information, to the knowledge module. This input triggers the knowledge module reasoning abilities which finally obtain a new interpretation of the video scene and some recommendations about how the tracking module should behave during the processing of the next frame.

The knowledge module includes the knowledge model of the system, represented with an ontology, and an ontological reasoner. The model, based on the JDL data processing model for Information Fusion systems, is stepped in several levels ranging from low-level track data to high-level scene situations [2]. These levels are:

- Tracking Entities (TREN) level, to model input data coming from the tracking module: track information (color, position, speed, etc.) and frames (to support the temporal consistency).
- Scene Objects (SCOB) level, to model real-world entities, properties, and relations: moving and static objects, topological relations, etc.
- Activities (ACTV) level, to model behavior descriptions: grouping, approaching, picking an object, etc.
- Impact (IMPC) level, to model the association between a cost value and an activity description.

These abstract ontologies are the building blocks of application-specific knowledge models. Each ontology level provides a skeleton that includes general concepts and relations to describe very general computer vision entities and relations, in such a way that they can be extended with more concrete concepts and relations to suit to the requirements of a specific domain.

Ontologies in the knowledge module may contain both perceptual and context data. Perceptual data is automatically extracted by the tracking algorithm, while the context data is external knowledge used to complete the comprehension of the scene. Context data includes information about scene environment, the parameters of the recording, information previously computed and user-requested information. For example, the description of a static object (size, position, kind of object, etc.) is regarded as context data.

IV. IMPLEMENTATION OF THE KNOWLEDGE MODULE

A system prototype has been built to implement the specifications of the architecture presented in Section 3. As explained, the cornerstone of this architecture is the knowledge module. The implementation of the module is based on the RACER reasoner [16] (see Figure 2). RACER has been chosen because it includes support for different kind of inference rules such as deductive, abductive, spatial, temporal, etc. [2]. In addition, RACER manages the spatial knowledge using the RCC theory as a substrate. A substrate associates an assertional box (ABox), which store statements about domain individuals, with an additional representation layer. RCC substrate also offers querying facilities, such as spatial queries and combined spatial and non-spatial queries.
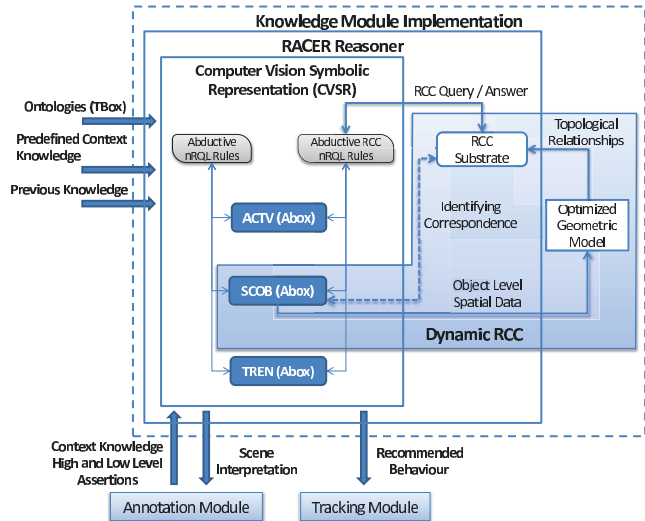

Fig. 2: Implementation of the Knowledge Module

In this implementation, the reasoner hosts the computer vision symbolic representation (CVSR), which includes the three lowest levels of the ontology-based model; namely, TREN, SCOB, and ACTV. The corresponding instances of these ontologies are also included. These individual assertions may come from a priori contextual sources (given by the user or obtained in previous executions) or from internal reasoning mechanisms.

Beyond the standard ontology reasoning mechanism based on subsumption, RACER also support abductive and deductive rule-based inference. During the execution, abductive nRQL (new Racer Query Language) rules defined in a sub-ontology create new instances in the same level or to an upper level. Eventually, the creation of new instances as defined in the consequents of the rules will draw instances corresponding to an interpretation of the scene in terms of the ACTV ontology. Deductive rules, in turn, are used to maintain the logical consistency of the scene. Both types of rules support the creation of feedback information to improve the behavior of the tracking system. The management of recommendations has been partially adapted to the prototype. The current development implements direct modifications of track properties according to predefined situations, such as occlusions.

A significant amount of knowledge of SCOB and ACTV is obtained by rules that include spatial properties in their antecedent. It must be noticed that abductive reasoning with spatial entities is very expensive in terms of computation time,

since it grows with the number of entities and the complexity of the scene increases. The Dynamic RCC module, integrated into the system prototype, solves this problem.

A. Dynamic RCC

Dynamic RCC includes three main components: a knowledge base with spatial features from individuals corresponding to the SCOB abstraction level; an optimized geometric model composed of a geometric model and an auxiliary data structure; and a RCC implementation which stores the qualitative spatial relationships. The overall architecture is depicted in Figure 3.
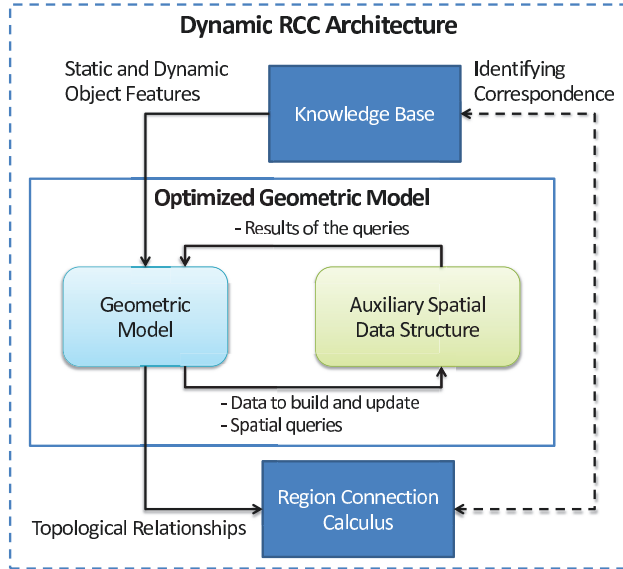

Fig. 3: Dynamic RCC architecture

The knowledge base of Dynamic RCC contains representations of both dynamic and static objects. Geometric features of each individual, such as position or size, are used as the input to the geometric model. These features are instantiated into the geometric model only in two cases: (i) when they do not exist previously in the model; (ii) when they were already instantiated, but its position or size has been modified during the last update. To obtain the new topological relationships when the situation changes, it is necessary to perform a full topological analysis between the newly instantiated or updated geometries and the remaining geometries. This implies that the geometries which have changed their position or size have to be checked against the static (M) and dynamic geometries (N). The total amount of checks (X) can be seen in (1).

$$X = N \cdot (M + N - 1) \tag{1}$$

Notice that topological relationships are symmetric. Therefore, the total amount of checks is (2).

$$X = \frac{N^2 + N \cdot (2M - 1)}{2} \tag{2}$$

Accordingly, the checking procedure has a quadratic complexity. This situation can turn out into a decrease of the system performance. In order to avoid this problem, it is recommendable to reduce the number of checks by choosing only those geometries that are candidates to modify the spatial relations of a given geometry. An auxiliary data structure can be used to determine these candidates (see next subsection), which usually form a clearly distinguished subgroup of the scene objects. Once the candidates have been obtained by querying the auxiliary data structure, the topological relations of a geometry can be updated by analysing only a few candidates. The topological relations which change from the previous state in the geometric model are then updated in the RCC system. Interestingly enough, changes in the ABox of the CVSR are not necessary, because the instance properties and topological relationships are stored in separated substrate. The consistency between them is achieved by using a common identifier.

B. Optimized Geometric Model

The optimized geometric model is composed of two complementary sub-modules: a geometric model and an auxiliary data structure.

The geometric model is a system that represents spatial 2-dimensional entities in an Euclidean plane and obtain their spatial relationships quickly, as explained in the previous subsection. The geometric model is implemented according to the OpenGIS Simple Features standard, which also includes supporting libraries and tools. This standard is a specification for digital storage of geographical data with spatial and non-spatial attributes. It defines a set of methods to evaluate the spatial relationships, like overlaps, contains, etc.; a set of methods to support spatial analysis, like distance, union, difference, etc.; relational operators between entities; and several kinds of representation point, such like multipoint, curve, surface, etc. Although OpenGIS spatial predicates and RCC-8 are not directly compatible, the output from the geometric model can be easily mapped from the OpenGIS format –in some cases, it only involves translating the name of the relationships. Correspondence between OpenGIS spatial predicates and RCC-8 can be found in [22].

The spatial data structure maintains a hierarchical topological sort on the Euclidean space of the geometries of the scene. It supports queries to retrieve the candidate geometries involved into a topological analysis; e.g., 'which geometries share area 2 with geometry 2?'. By default, these candidate geometries are the nearest geometries of the one in the query. The sorting of the auxiliary spatial data structure is not predefined by the architecture. It is required, however, to fulfill the following restrictions: (i) the spatial structure must be able to define a recursive spatial hierarchy; (ii) the spatial structure must handle the overlap between entities; (iii) search, deletion, insertion and update operations must not imply a high overhead. A recursive and regular cell hierarchy can be used; some advantages of this structure are the ability to adapt to

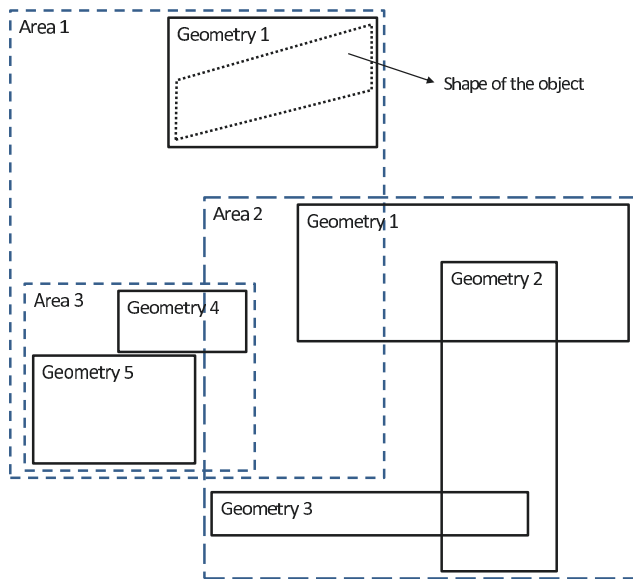the entity size, and the support for proximity-based sorting through a restructuring of the hierarchy.

Fig. 4: Example of structured spatial information

Finally, it is important to highlight that data structures assume that spatial entities are represented by axis-aligned bounding boxes, even though tracked object do not satisfy this condition. Consequently, it is necessary to implement an algorithm to calculate the smallest rectangle which enclose the corresponding object. The geometry inserted into the data structure is not the track, but the smallest axis-aligned rectangle that includes it.

V. CASE STUDY: SPORTS

The prototype presented can be used to improve multi-camera tracking applications in sports environments. In this context, it is desirable to know which is the most reliable camera perspective for situation assessment during a multi-camera video tracking analysis. By using the topological relationships between tracks, it is possible to reason which is the camera or cameras that can offer a greater accuracy. Tracking analysis loses accuracy and quality when grouping or occlusion situations occur; i.e., when tracks are related with the RCC relations partial overlap (po), proper part (pp), or externally connected (ec).

A suitable criteria to make a good camera selection is the following:

- The camera must detect a topological relation of lower degree of overlap between the entities than the previous camera.
- The camera visual field must activate a proper part relationship with the entities involved into the analysis. (To perform this reasoning procedure it is necessary to know the context information related with the camera visual field.)

Some other conditions can be used to tune the behavior of the system; for instance, to require a minimum track size to carry out the analysis. Notice that cameras must be calibrated, in such a way that local objects coordinates can be transformed into the global coordinates space.

The images below show a situation where the overhead camera (5a) might suffer a lost of accuracy, as the labelled tracks are grouped. In this conditions, the confidence of the lateral camera (5b) could be increased, since it meets the conditions mentioned above.

A simple rule can detect the conditions and update the confidence value of each camera. The rule antecedent identifies situations with two tracks and a lateral point of view camera without a confidence value. If both tracks are in a proper part relationship with the field view area of the lateral camera and, at the same time, the tracks are in a disconnected relationship (dr), then the rule consequent generates a new assertion which assigns a confidence value to the lateral camera.

```
(firerule
  (and  //Antecedent
    (?track1 #!tren:Track)
    (?track2 #!tren:Track)
    (?lateralCam #!scob:CameraFieldView)
    (?lateralCam #!scob:conf #!scob:IsNotReliable))
    (?*lateralCam ?*track1 :pp)   //First criterion
    (?*lateralCam ?*track2 :pp)   //First criterion
    (?*track1 ?*track2 :dr)       //Second criterion
  )
  (   //Consequent
  (related ?lateralCam #!scob:conf  #!scob:IsReliable))
)
```

For the sake of simplicity, the second condition has been directly reduced to a disconnected relationship between the entities. Identification statements have been deleted, and RCC-5 relations has been used instead of RCC-8 .

Scenarios with a high number of cameras could use a mixed approach to detect the orientation of the topology relationships. There are several approaches to manage orientation as qualitative relationships. The best known are the cone-based and projection-based techniques, which model the cardinal directions (n, ne, e, se, s, sw, w, nw, equal) depending on the located object, the reference object, and the frame of reference.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have discussed some of the challenges that must be faced to incorporate spatial and topological representation and reasoning features to ontological formalisms. We have reviewed some current approaches and have discussed their features and limitations. In addition, we have proposed a hybrid approach that is used in a framework for ontology-based interpretation of multi-camera data. Topological relations among the individuals of a scene are discovered thanks to a combination of standard tools for spatial representations and an ontology reasoner powered with a RCC substrate. Some optimizations techniques have been applied to improve the performance of the prototype; namely, the use of specialized data structures to minimize model update.

(a) Situation showed from the overhead point of view. Frame 135.

(b) Situation showed from the lateral point of view. Frame 135.

Fig. 5: Overlapping situation

Future works will include a complete study about which spatial structures may be more appropriate for our problem. Data structure implementation and performance results of the approach are also pending for future research, as well as the applicability of the current implementation to more complex scene recognition problems.

REFERENCES

[1] D. Randell, Z. Cui, and A. Cohn, "A spatial logic based on regions and connection," in *Principles of knowledge representation and reasoning: Proceedings of the 4th International Conference (KR'94)*, Bonn, Germany, May 1994, pp. 165–176.

[2] J. Gómez-Romero, M. Patricio, J. García, and J. Molina, "Ontology-based context representation and reasoning for object tracking and scene interpretation in video," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7494–7510, 2011.

[3] J. Llinas, C. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White, "Revisiting the JDL data fusion model II," in *Proceedings of the 7th International Conference on Information Fusion (Fusion2004)*, Stockholm, Sweden, July 2004, pp. 1218–1230.

[4] B. Clarke, "A calculus of individuals based on 'connection'," *Notre Dame Journal of Formal Logic*, vol. 22, no. 3, pp. 204–218, 1981.

[5] ——, "Individuals and points," *Notre Dame Journal of Formal Logic*, vol. 26, no. 1, pp. 61–75, 1985.

[6] B. Bennett, "Spatial reasoning with propositional logics," in *Principles of knowledge representation and reasoning: Proceedings of the 4th International Conference (KR'94)*, Bonn, Germany, May 1994, pp. 51–62.

[7] M. Egenhofer, "Reasoning about binary topological relations," in *Proceedings of the Second International Symposium on Advances in Spatial Databases (SSD'91)*, Zürich, Switzerland, August 1991, pp. 143–160.

[8] Y. Katz and B. Grau, "Representing qualitative spatial information in OWL-DL," in *Proceedings of OWL: Experiences and Directions (OWLED 2005)*, Galway, Ireland, November 2005.

[9] F. Baader, I. Horrocks, and U. Sattler, "Description logics as ontology languages for the semantic web," *Mechanizing Mathematical Reasoning*, pp. 228–248, 2005.

[10] M. Stocker and E. Sirin, "PelletSpatial: A hybrid RCC-8 and RDF/OWL reasoning and query engine," in *Proceedings of OWL: Experiences and Directions (OWLED 2009)*, Chantilly, VA, USA, October 2009, pp. 2–31.

[11] I. Horrocks, O. Kutz, and U. Sattler, "The even more irresistible $\mathcal{SROIQ}$," in *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR'06)*, Lake District, UK, June 2006, pp. 57–67.

[12] R. Grütter and B. Bauer-Messmer, "Combining OWL with RCC for spatioterminological reasoning on environmental data," in *Proceedings of OWL: Experiences and Directions (OWLED 2007)*. Innsbruck, Austria: June, 2007.

[13] ——, "Towards spatial reasoning in the semantic web: A hybrid knowledge representation system architecture," *The European Information Society*, pp. 349–364, 2007.

[14] F. Hogenboom, B. Borgman, F. Frasincar, and U. Kaymak, "Spatial knowledge representation on the semantic web," in *Proceedings of the IEEE 4th International Conference on Semantic Computing (ICSC 2010)*, Pittsburgh, PA, USA, September 2010, pp. 252–259.

[15] S. Rudolph, M. Krötzsch, and P. Hitzler, "Cheap boolean role constructors for description logics," *Logics in Artificial Intelligence*, pp. 362–374, 2008.

[16] V. Haarslev and R. Möller, "Description of the RACER system and its applications," in *Proceedings International Workshop on Description Logics (DL-2001)*, Stanford, CA, USA, July 2001, pp. 131–141. [Online]. Available: http://www.racer-systems.com/

[17] E. Klien, M. Lutz, and W. Kuhn, "Ontology-based discovery of geographic information services–An application in disaster management," *Computers, environment and urban systems*, vol. 30, no. 1, pp. 102–123, 2006.

[18] N. Wiegand and C. García, "A task-based ontology approach to automate geospatial data retrieval," *Transactions in GIS*, vol. 11, no. 3, pp. 355–376, 2007.

[19] (2011) Geospatial tutorial for AllegroGraph. [Online]. Available: http://www.franz.com/agraph/support/documentation/current/geospatial-tutorial.html

[20] (2011) GeoSpatialWeb. [Online]. Available: http://code.google.com/p/geospatialweb/

[21] W. Van Hage, J. Wielemaker, and G. Schreiber, "The space package: Tight integration between space and semantics," *Transactions in GIS*, vol. 14, no. 2, pp. 131–146, 2010.

[22] M. Schuele and P. Karaenke, "Qualitative spatial reasoning with topological information in BDI agents," in *Proceedings of the 2nd Workshop on Artificial Intelligence and Logistics (AILog)*, Lisbon, Portugal, August 2010, pp. 7–12.