SIVP manuscript No.
(will be inserted by the editor)

Image Indexing and Retrieval using Expressive Fuzzy Description Logics

N. Simou · Th. Athanasiadis · G. Stoilos · S. Kollias

Received: date / Accepted: date

Abstract The effective management and exploitation of multimedia documents requires the extraction of the underlying semantics. Multimedia analysis algorithms can produce fairly rich, though imprecise information about a multimedia document which most of the times remains unexploited. In this paper we propose a methodology for semantic indexing and retrieval of images, based on techniques of image segmentation and classification combined with fuzzy reasoning. In the proposed knowledge-assisted analysis architecture a segmentation algorithm firstly generates a set of over-segmented regions. After that, a region classification process is employed to assign semantic labels using a confidence degree and simultaneously merge regions based on their semantic similarity. This information comprises the assertional component of a fuzzy knowledge base which is used for the refinement of mistakenly classified regions and also for the extraction of rich implicit knowledge used for global image classification. This knowledge about images is stored in a semantic repository permitting image retrieval and ranking.

1 Introduction

During the last decade a dramatically large increase of digital multimedia content has occurred. The main reason that led to this change was the broad availability and use of digital devices, not only by professional or very experienced users, but almost by everyone. As a consequence new research interests have emerged having as primary aim,

This research was supported by the European Commission under contract FP6-027026 K-SPACE.

N. Simou · Th. Athanasiadis · G. Stoilos · S. Kollias
Image Video and Multimedia Systems Laboratory,
School of Electrical and Computer Engineering,
National Technical University of Athens,
Iroon Polytexneiou 9, 15780 Zografou, Greece
Tel.: +30-210-7722521
Fax: +30-210-7722492
E-mail: nsimou,thanos,gstoil,stefanos@image.ntua.gr

among others, the efficient management of multimedia content providing mechanisms for indexing and retrieval.

To meet this objective, various approaches have been proposed during the years. The first attempts were based on metadata generation for the multimedia documents [26]. Different types of metadata were used which can be separated based on their context into low level like MPEG-7 descriptors and higher level like thematic categorization [19]. Furthermore different methods for metadata generation have been examined such as automatic, semi-automatic and manual. The research community has focused on the automatic extraction of semantic information, which is desired for interaction by most users. Clearly semantic indexing and retrieval is a very challenging task that requires significant effort in various fields. Firstly, efficient multimedia analysis and processing techniques are required for the extraction of semantic information. Additionally, artificial intelligence algorithms need to effectively manage the provided information by using it for the extraction of new implicit information. Finally, the extracted information, implicit and explicit, has to be stored in such a way that it is accessible for retrieval in a user-friendly way.

Most approaches related to semantic-based analysis and indexing are grounded on the implementation of semantic concepts detectors. Evaluation of these techniques indicates that a good level of maturity has been reached [25, 20, 28]. At the same time, image segmentation and object recognition have been used simultaneously, instead of sequentially, aiming at improvement in both domains. Such approaches include graph-cuts minimization algorithms [15], fuzzy region labeling for semantic region growing [4] or integrating bottom-up and top-down approaches [8]. On the other hand the majority of classification techniques employs statistical modeling, associating low-level visual features with mid-level concepts [20, 9]. Popular machine learning techniques used for region-based image classification include Neural Networks, Support Vector Machines (SVMs) [33] and AdaBoost [32].

This extracted information by multimedia analysis algorithms requires effective management capable of inferring complicated concepts. In this context various attempts have been examined using taxonomies and ontologies. Hollink [16] attempts to specify the necessary requirements for a visual ontology for video annotation and proposes the use of a WordNet/MPEG-7 ontology combination towards that scope. Hoogs et al. [2] couple a classical image analysis objects and events recognition approach with WordNets semantics, taking advantage of its hierarchical relationships structure. WordNet was also used to include lexical relationships between abstract and detected mid-level concepts in [27], while in [5] authors focus on the use of a multimedia ontology infrastructure for analysis and semantic annotation of multimedia content. Finally, in [21] the issue of scalability of multimedia ontologies is discussed.

Ontologies are based on Description Logics (DLs) [6] that are a family of knowledge representation languages. However, despite the rich expressiveness of DLs, they lack the ability to deal with vague and uncertain information which is very common in multimedia content. This was the reason that a variety of DLs capable of handling imprecise information, like probabilistic [12] and fuzzy [30, 29] have been proposed. In any of these variations the main role of an ontology remains the same: proper representation of semantic information in a readable format, permitting to software agents to use it in order to find, share and integrate information more easily. The great evolution of ontologies during the last decade, bred the need of data storage systems

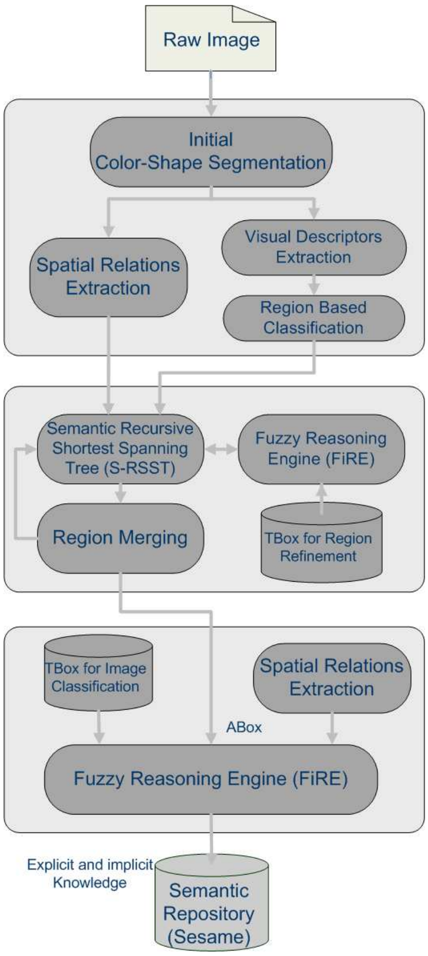


Fig. 1 Overview of the proposed architecture.

that were built specifically for storing and querying ontologies, like Sesame[1], Kowari[2], Jena[3] and more.

In this paper we present a knowledge assisted image analysis and automatic semantic annotation methodology consisting of several novel and state-of-the-art techniques. The architecture of our proposal is shown in Figure 1. As can be seen, we initially segment an image based on color and shape criteria. It is well known that with such criteria image segmentation algorithms fail to extract semantically meaningful objects. For that reason we introduce a novel semantic region growing methodology which incorporates object detection simultaneously with region merging, providing better input for region-based classification. To further improve this operation we use the fuzzy reasoning engine FiRE together with a terminology (TBox) defined in order to improve

[1] `http://www.openrdf.org/`
[2] `http://www.kowari.org/`
[3] `http://jena.sourceforge.net/`

region-based classification by incorporating spatial relations and neighborhood information (Role Assertions). The previous step provides a list of concepts (together with degrees of confidence for each one) that have been linked to the image (Concept Assertions). Our purpose is to use this information in order to extract additional, implicit knowledge, and also infer abstract concepts on a global image basis. Towards this aim, FiRE is employed using this time a different terminology especially for this purpose. The final results are stored in an online semantic repository, in a strictly structured format, allowing query mechanisms for semantic retrieval and ranking. The main contributions of the proposed architecture are:

- It presents a semantic-aware segmentation algorithm together with a region-based classification.
- It incorporates expressive fuzzy description logics together with the obtained results from the classification algorithms for segmentation refinement and extraction of higher implicit information.
- It presents a novel way for ranking and retrieval of images based on querying of expressive fuzzy knowledge bases.

The rest of the paper is organized as follows. Section 2 presents the region-based classification and the semantic-aware segmentation algorithm. Section 3 presents the overall architecture for fuzzy reasoning storage and querying. Section 3.1 shortly introduces the fuzzy DL f -$\mathcal{SHIN}$, while the fuzzy reasoning engine FiRE, which is based on the fuzzy DL f -$\mathcal{SHIN}$, along with its syntax and reasoning services, are presented in Section 3.2. Next, Section 3.3 presents the integration of FiRE with the RDF store Sesame and Section 3.4 describes the implementation of expressive queries to exploit fuzziness which are used in this case for the retrieval and ranking of images. Finally, we provide experimental results of the overall approach in Section 4 and we draw our conclusions in Section 5.

2 Semantic Segmentation and Region Classification

Knowledge-assisted analysis, in the context of this work, deals with the very important and difficult task of the instantiation of a fuzzy knowledge base by processing an image. The optimum operation of this task is crucial for our architecture since mistaken instantiation of the fuzzy knowledge base would lead to mistaken inferred knowledge. In order to achieve this objective, a semantic variation of the Recursive Shortest Spanning Tree (RSST) segmentation algorithm [18] is employed together with the fuzzy reasoning engine FiRE. We first describe the graph representation of an image used in our approach and the way it is initialized. Then, the Semantic RSST (S-RSST) algorithm is detailed and its operation on this graph together with FiRE.

2.1 Graph Representation of an Image

An image can be described as a structured set of individual objects, allowing thus a straightforward mapping to a graph structure. In this fashion, many image analysis problems can be considered as graph theory problems, inheriting the solid theoretical grounds of the latter. Attributed Relation Graph (ARG) [7] is a type of graph often used in computer vision and image analysis for the representation of structured objects.

Formally, an ARG is defined by spatial entities represented as a set of vertices V and binary spatial relationships represented as a set of edges E: $ARG \equiv \langle V, E\rangle$. Letting G be the set of all connected, non-overlapping regions/segments of an image, then a region $a \in G$ of the image is represented in the graph by vertex $v_a \in V$, where $v_a \equiv \langle a, D_a, \mathcal{L}_a\rangle$. D_a is the ordered set of MPEG-7 Visual Descriptors characterizing the region in terms of low-level features, while $\mathcal{L}_a = \sum_{i=1}^{|\mathbf{C}|} C_i/\mu_a(C_i)$ is the fuzzy set of concepts assigned to the region, defined over the crisp set of concepts $\mathbf{C} = \{C_i\}$. Following the sum notation for fuzzy sets [14], μ_a is the membership function of the fuzzy set and $\mu_a(C_i)$ represents the degree of confidence that region a depicts concept C_i. The adjacency relation between two neighbor regions $a, b \in G$ of the image is represented by graph's edge $e_{ab} \equiv \langle (v_a, v_b), s_{ab}\rangle$. s_{ab} is a similarity value for the two adjacent regions represented by the pair (v_a, v_b). This value is calculated based on the semantic similarity of the two regions as described by the two fuzzy sets $\mathcal{L}_a$ and $\mathcal{L}_b$:

$$s_{ab} := \max_{C_i \in \mathbf{C}} \{\top(\mu_a(C), \mu_b(C))\}, \quad a \in G, b \in N_a \tag{1}$$

where $\top$ is a t-norm and N_a is the set of neighbors of a defined below. The above formula states that the similarity of two regions is the supremum of all common concepts of the t-norm $\top$ of $\mu_a(C_i)$ and $\mu_b(C_i)$ for the specific concept of the two regions a and b.

Next, we consider two regions $a, b \in G$ to be connected when at least one pixel of one region is 4-connected to one pixel of the other. In an ARG, a neighborhood N_a of a vertex $v_a \in V$ is the set of vertices whose corresponding regions are connected to a: $N_a = \{v_b : e_{ab} \neq \varnothing\}$, $a, b \in G$. Finally, the subset of ARG's edges that are incident to region a can be defined as: $E_a \equiv \{e_{ab} : v_b \in N_a\}$.

2.2 Image Graph Initialization

The graph representation of an image, as detailed in the previous subsection, is the basis of all further analysis tasks. Here, we discuss briefly how this graph is constructed, i.e. compute the vertices' attributes D_a and $\mathcal{L}_a$, as well as the semantic similarity s_{ab} for all graph's edges. Graph initialization consists mainly of four processes:

- Initial (over)segmentation.
- Extraction of spatial relations between regions.
- Extraction of visual descriptors.
- Region-based classification.

Initially, a segmentation algorithm, based on low-level features such as color and texture [3], is applied in order to divide the given image into regions. Based on the relative location of each region, binary spatial relations are extracted. Further on, corresponding low-level visual descriptions (such as MPEG-7 visual descriptors) are computed for every resulting region. The latter are employed so as to form a compound low-level descriptor vector for every image region, based on a simple concatenation mechanism. The computed feature vector is employed for generating an initial set of the region's candidate semantic labels $\mathcal{L}_a$ together with a degree of confidence for each of them using a trained Support Vector Machine (SVM) [11]. Based on the fuzzy sets of labels of each region and the neighborhood information as extracted from the spatial relations, we calculate the semantic similarity for every graph's edge.

2.3 Semantic RSST

As previously argued, segmentation based on cues such as color and texture fails to extract semantic objects and thus region-based classification also fails to recognize objects of interest within an image (material-like concepts like *sky*, *sand*, etc. being an exception). Here, we present a methodology to improve both image segmentation and detection of concepts at the same time, providing a more robust mechanism for semantic indexing.

RSST [18] is a bottom-up segmentation algorithm that begins from the pixel level and iteratively merges similar neighbor regions until certain termination criteria are satisfied. RSST may use internally a graph representation of image regions, like the ARG described in Section 2.1. In the beginning, all edges of the graph are sorted according to a criterion, e.g. color dissimilarity of the two connected regions using Euclidean distance of the color components. The edge with the least weight is found and the two regions connected by that edge are merged. After each step, the merged region's attributes (e.g. region's mean color) are re-calculated. Traditional RSST will also re-calculate weights of related edges as well and re-sort them, so that in every step the edge with the least weight will be selected. This process goes on recursively until termination criteria are met. Such criteria may vary, but usually these are either the final number of regions or a threshold on the weight.

Following the conventions and notation used so far, we introduce here a modified version of RSST, called Semantic RSST (S-RSST) with novel (dis)similarity and termination criteria. The criterion for ordering the edges is not a distance metric based on low-level features, but the semantic similarity measure s_{ab} defined in Equation (1). For an edge between two adjacent regions a and b we define its weight as follows:

$$w(e_{ab}) = 1 - s_{ab} \tag{2}$$

Considering that an edge's weight should represent the degree of dissimilarity between the two joined regions and since $s_{ab} \in [0 \ldots 1]$, we simply subtract the estimated value from one. Commutativity and associativity axioms of all fuzzy set operations (thus including default fuzzy union and default fuzzy intersection) ensure that the ordering of the arguments is indifferent. In this way all graph's edges are sorted by their weight.

Let us now examine in detail one iteration of the S-RSST algorithm. Firstly, the edge with the least weight is selected as:

$$e^*_{ab} = \underset{e_{ab} \in E}{\operatorname{argmin}}(w(e_{ab})), \quad a \in G, b \in N_a \tag{3}$$

Then regions a and b are merged to form a new region $\hat{a}$. Region b is removed completely from the ARG, whereas a is updated appropriately. Region merging, apart from the effect to the graph, it actually changes the segmentation map of the image, by joining the two sets of pixels of the two regions a and b. The update procedure at the graph consists of the following two actions:

1. Update graph's vertices: Consulting FiRE to re-evaluate the degrees of confidence of each region for every concept taking into account the current region labels and their spatial relations (see Section 4.2).
2. Update graph's edges, which is subdivided into two tasks:
   (a) Removal of edge e_{ab}, i.e. update the Role assertions

(b) Re-evaluation of the weight according to equation (2) of all affected edges $\hat{e}$, i.e. the union of those incident to region a and of those incident to region b: $\hat{e} \in E_a \cup E_b$

This procedure continues until the edge e^* with the least weight in the ARG is above a threshold: $w(e^*) > T_w$. This threshold is calculated in the beginning of the algorithm (similarly to the traditional RSST), based on the cumulative histogram of the weights of all edges E. It should be pointed out that the interaction of the S-RSST algorithm with FiRE for the refinement of the degrees of detected concepts corrects erroneous classifications of regions and assists the region merging process towards the segmentation of semantic objects (see Sections 4.2, 4.3).

3 Fuzzy Reasoning, Storing and Querying

This section gives a brief introduction to the fuzzy DL f-$\mathcal{SHIN}$ also presenting the fuzzy reasoning engine FiRE that currently supports it. The graphical user interface, the syntax and the inference services of FiRE are presented. Furthermore FiRE integration with the RDF store Sesame, which is an open source Java framework for storing and querying RDF/RDFS data, is presented. The main benefit from this integration is the implementation of *Conjunctive Threshold Queries* and *General Fuzzy Conjunctive Queries*, based on which semantic ranking and retrieval of images is achieved.

3.1 The Fuzzy DL f-$\mathcal{SHIN}$

The Description Logic (DL) f-$\mathcal{SHIN}$ is a fuzzy extension of the DL $\mathcal{SHIN}$ [13] and it similarly consists of an alphabet of distinct concept names ($\mathbf{C}$), role names ($\mathbf{R}$) and individual names ($\mathbf{I}$). Using DLs, the construction of new concepts and roles is possible. For that purpose DLs include a set of constructors to construct concept and role descriptions. These constructors specify the name of the DL language [6] and in the case of f -$\mathcal{SHIN}$ these are the $\mathcal{ALC}$ constructors (i.e. negation $\neg$, conjunction $\sqcap$, disjunction $\sqcup$, full existential quantification $\exists$ and value restriction $\forall$) extended by transitive roles ($\mathcal{S}$), role hierarchy ($\mathcal{H}$), inverse roles ($\mathcal{I}$), and number restrictions ($\mathcal{N}$ $\leq$, $\geq$). Hence, if R is a role then R^- is also a role, namely the inverse of R. f-$\mathcal{SHIN}$ concepts are inductively defined as follows:

1. If $C \in \mathbf{C}$, then C is a f-$\mathcal{SHIN}$ concept
2. If C and D are concepts, R is a role, S is a simple role and $n \in \mathbb{N}$, then $(\neg C)$, $(C \sqcup D)$, $(C \sqcap D)$, $(\forall R.C)$, $(\exists R.C)$, $(\geq nS)$ and $(\leq nS)$ are also f-$\mathcal{SHIN}$ concepts.

Differently to crisp DLs, the semantics of fuzzy DLs are given by a *fuzzy interpretation* [30]. A fuzzy interpretation is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ where $\Delta^{\mathcal{I}}$ is a non-empty set of objects and $\cdot^{\mathcal{I}}$ is a fuzzy interpretation function, which maps an individual name a to elements of $\mathsf{a}^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ and a concept name A (role name R) to a membership function $\mathsf{A}^{\mathcal{I}} : \Delta^{\mathcal{I}} \to [0,1]$ ($R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0,1]$).

By using fuzzy set theoretic operations the fuzzy interpretation function can be extended to give semantics to complex concepts, roles and axioms [14]. In this case the standard fuzzy operators of $1-x$ (c) for fuzzy negation and max (u), min (t) for fuzzy union and intersection are used, respectively. The complete set of semantics is depicted in Table 1.

Table 1 Semantics of concepts and roles

Constructor	Syntax	Semantics
top	$\top$	$\top^{\mathcal{I}}(a) = 1$
bottom	$\bot$	$\bot^{\mathcal{I}}(a) = 0$
general negation	$\neg C$	$(\neg C)^{\mathcal{I}}(a) = c(C^{\mathcal{I}}(a))$
conjunction	$C \sqcap D$	$(C \sqcap D)^{\mathcal{I}}(a) = t(C^{\mathcal{I}}(a), D^{\mathcal{I}}(a))$
disjunction	$C \sqcup D$	$(C \sqcup D)^{\mathcal{I}}(a) = u(C^{\mathcal{I}}(a), D^{\mathcal{I}}(a))$
exists restriction	$\exists R.C$	$(\exists R.C)^{\mathcal{I}}(a) = \sup_{b \in \Delta^{\mathcal{I}}}\{t(R^{\mathcal{I}}(a,b), C^{\mathcal{I}}(b))\}$
value restriction	$\forall R.C$	$(\forall R.C)^{\mathcal{I}}(a) = \inf_{b \in \Delta^{\mathcal{I}}}\{\mathcal{J}(R^{\mathcal{I}}(a,b), C^{\mathcal{I}}(b))\}$
at-most	$\leq pR$	$\inf_{b_1,\ldots,b_{p+1} \in \Delta^{\mathcal{I}}} \mathcal{J}(t_{i=1}^{p+1} R^{\mathcal{I}}(a,b_i), u_{i<j}\{b_i = b_j\})$
at-least	$\geq pR$	$\sup_{b_1,\ldots,b_p \in \Delta^{\mathcal{I}}} t(t_{i=1}^{p} R^{\mathcal{I}}(a,b_i), t_{i<j}\{b_i \neq b_j\})$
inverse role	R^-	$(R^-)^{\mathcal{I}}(b,a) = R^{\mathcal{I}}(a,b)$
equivalence	$C \equiv D$	$\forall a \in \Delta^{\mathcal{I}}.C^{\mathcal{I}}(a) = D^{\mathcal{I}}(a)$
sub-concept	$C \sqsubseteq D$	$\forall a \in \Delta^{\mathcal{I}}.C^{\mathcal{I}}(a) \leq D^{\mathcal{I}}(a)$
transitive role	$\mathsf{Trans}(R)$	$\forall a,b \in \Delta^{\mathcal{I}}.R^{\mathcal{I}}(a,b) \geq \sup_{c \in \Delta^{\mathcal{I}}}\{t(R^{\mathcal{I}}(a,c), R^{\mathcal{I}}(c,b))\}$
sub-role	$R \sqsubseteq S$	$\forall a,b \in \Delta^{\mathcal{I}}.R^{\mathcal{I}}(a,b) \leq S^{\mathcal{I}}(a,b)$
concept assertions	$\langle a : C \bowtie n\rangle$	$C^{\mathcal{I}}(a^{\mathcal{I}}) \bowtie n$
role assertions	$\langle (a,b) : R \bowtie n\rangle$	$R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \bowtie n$

An f-$\mathcal{SHIN}$ knowledge base Σ is a triple $\langle \mathcal{T}, \mathcal{R}, \mathcal{A}\rangle$, where $\mathcal{T}$ is a fuzzy $TBox$, $\mathcal{R}$ is a fuzzy $RBox$ and $\mathcal{A}$ is a fuzzy $ABox$. $TBox$ is a finite set of fuzzy concept axioms which are of the form $C \equiv D$ called fuzzy concept equivalence axioms or $C \sqsubseteq D$ called fuzzy concept inclusion axioms saying that C is equivalent or C is a sub-concept of D, respectively. Similarly, $RBox$ is a finite set of fuzzy role axioms of the form $\mathsf{Trans}(R)$ called fuzzy transitive role axioms and $R \sqsubseteq S$ called fuzzy role inclusion axioms saying that R is transitive and R is a sub-role of S, respectively. Finally, $ABox$ is as finite set of fuzzy assertions of the form $\langle a : C \bowtie n\rangle$, $\langle (a,b) : R \bowtie n\rangle$, where $\bowtie$ stands for $\geq, >, \leq, <$ or $a \neq b$, for $a, b \in \mathbf{I}$. Fuzzy representation enriches expressiveness, so a fuzzy assertion of the form $\langle a : C \geq n\rangle$ means that a participates in the concept C with a membership degree that is at least equal to n.

3.2 Fuzzy Reasoning Engine FiRE

3.2.1 FiRE interface

FiRE[4] is a Java based fuzzy reasoning engine currently supporting f -$\mathcal{SHIN}$ that can be used either as an API by another application or through its graphical user interface. The graphical user interface of FiRE consists of the editor panel, the inference services panel and the output panel (Figure 2). Hence the user has the ability to create or edit an existing fuzzy knowledge base using the editor panel, and to use the inference services panel to make different kinds of queries to the fuzzy knowledge base. Finally, the output panel consists of four different tabs, each one displaying feedback depending on the user operation.

[4] FiRE can be found at `http://www.image.ece.ntua.gr/~nsimou/FiRE/` together with installation instructions and examples

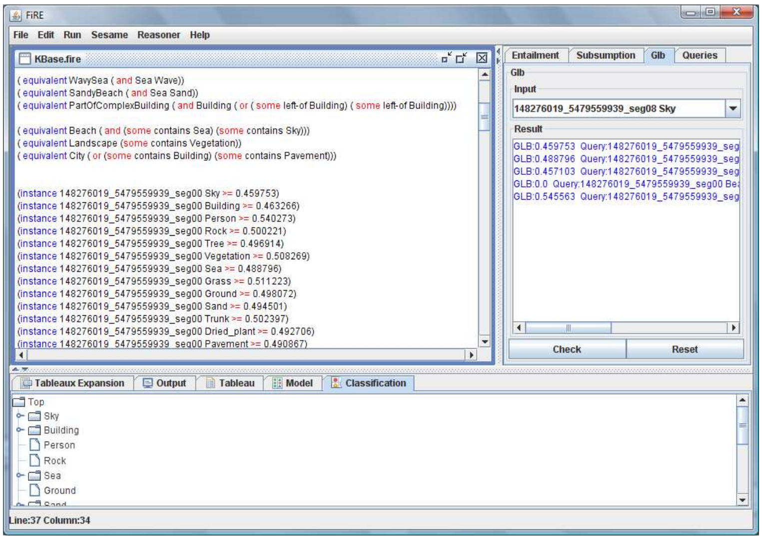


Fig. 2 The FiRE user interface consists of the editor panel (upper left), the inference services panel (upper right) and the output panel (bottom)

3.2.2 FiRE syntax

As previously mentioned, a fuzzy knowledge base consists of three components $TBox$, $RBox$ and $ABox$. $TBox$ and $RBox$ are defined using the Knowledge Representation System Specification [1] proposal since they do not include uncertainty. So, transitive roles or the sub-role of another role can be defined by using the keywords **transitive** and **parent** respectively and concept axioms by the keywords **implies** and **equivalent**. (Please refer to [1] for a full specification.)

On the contrary, since the assertions are extended to fit imperfect knowledge, ABox in fuzzy DLs is different. Instances in FiRE are defined using the keyword **instance** followed by the individual, the concept in which the individual participates, the inequality type (one of $<, <=, >, >=$) and the degree of confidence $degree \in [0, 1]$. Similarly, role assertions are defined by using the keyword **related** followed by subject and object individuals, the inequality type and the degree of confidence. In both cases the inequality type and the degree of confidence are required only for fuzzy assertions, if these are not mentioned then the assertions are assumed as crisp (i.e $>= 1$).

Example 1 The syntax of the assertions $\langle region1 : Sky \rangle$, $\langle region2 : (Sand \sqcap Sea) \geq 0.8 \rangle$, $\langle (region1, region2) : isAboveOf \geq 0.7 \rangle$ are shown below in FiRE syntax.

```
(instance region1 Sky)
(instance region2 (and Sea Sand) >= 0.8)
(related region1 region2 isAboveOf >= 0.7)
```

3.2.3 FiRE reasoning services

One of the main advantages of DLs compared with other formal representation languages is their sound and complete algorithms that can be used for reasoning. The main reasoning services offered by crisp reasoners are satisfiability checking, subsumption and entailment of concepts and axioms with respect to an ontology. In other words, using these reasoning services someone is capable of answering questions like "Can the concept C have any instances in models of the ontology T?" (satisfiability of C), "Is the concept D more general than the concept C in models of the ontology T?" (subsumption $C \sqsubseteq D$) "Does axiom Ψ logically follow from the ontology?" (entailment of Ψ).

These reasoning services are also available by f-$\mathcal{SHIN}$ together with *greatest lower bound queries* which are specific to fuzzy assertions. In the case of fuzzy DL, satisfiability questions become of the form "Can the concept C have any instances with degree of participation $\bowtie n$ in models of the ontology T?". Furthermore, the incorporation of degrees in assertions makes the evaluation of the best lower and upper truth-value bounds of a fuzzy assertion vital. The term of *greatest lower bound* of a fuzzy assertion with respect to Σ was defined in [30]. Informally, greatest lower bound queries are queries like "What is the greatest degree n that our ontology entails an individual a to participate in a concept C?".

FiRE uses the tableau algorithm of f-$\mathcal{SHIN}$, presented by Stoilos et al. [29], in order to decide the key inference problems of a fuzzy ontology. Hence entailment queries that ask whether our knowledge base logically entails the membership of an individual to a specific concept to a certain degree, are specified in the *Entailment* inference tab (see Figure 2). Their syntax is the same as the one used for the definition of a fuzzy instance. So, for example a statement of the form:

```
(instance region1 (and Sea Sky) > 0.8)
```

would ask whether `region1` is Sea and Sky to a degree greater than or equal to 0.8. If there are assertions in the *ABox* of our Σ that satisfy this query (i.e. there is a model for our ontology) then FiRE will return true.

On the other hand subsumption queries that are specified in the *Subsumption* inference tab evaluate whether a concept is more general than another concept. Their syntax is of the following form:

```
(concept1) (concept2)
```

where `concept1` and `concept2` are f -$\mathcal{SHIN}$ concepts. Let's assume that the first concept is Beach while the second concept is Sea. Since Beach is a more complex concept let's assume that it has been defined in the $TBox$ using an equivalence axiom as a conjunction of concepts Sea and Sand, i.e. $\mathsf{Beach} \equiv \mathsf{Sea} \sqcap \mathsf{Sand}$. Then the following subsumption query will always return true since Beach will always be (i.e. in all models) a sub-concept of Sea.

```
(Beach) (Sea)
```

Additionally, the user can perform a global concept classification procedure presenting the concept hierarchy tree in the *Classification* tab of the output panel.

Finally, FiRE permits the user to make greatest lower bound queries (GLB), which are evaluated by FiRE performing entailment queries. During this procedure a set of entailment queries is constructed consisting of an entailment query for every degree contained in the ABox, using the individual and the concept of interest. These queries are performed using the binary search algorithm to reduce the degrees search space, resulting the GLB. The syntax of GLB queries is of the form:

```
individual (concept)
```

where `concept` can be either an atomic concept or a result of f -$\mathcal{SHIN}$ constructors. In order to illustrate the operation of the GLB service we will present a trivial example using an atomic concept. Let's assume the following *ABox* (FiRE syntax) for which we want to evaluate the GLB of individual `region1` participating in concept Sand. (i.e. the query `region1` Sand)

```
(instance region1 Sand > 0.8)
(instance region2 Sea > 0.6)
(instance region3 Person > 0.5)
(instance region4 Tree > 1)
```

Firstly all the degrees used in *ABox* are sorted. FiRE then performs entailment queries for the `region1` (region of interest) participating in Sand (concept of interest) using the binary search algorithm. This procedure is repeated until the entailment query is unsatisfiable. The greatest degree found before unsatisfiability is the greatest lower bound. In this example the following entailment queries are performed with the indicated results in order to evaluate that the greatest lower bound of `region1` participating in concept Sand is 0.8.

`(instance region1 Sand > 0.5)` TRUE
`(instance region1 Sand > 0.8)` TRUE
`(instance region1 Sand > 1)` FALSE

(More complicated examples that use complex concepts are presented in Sections 4.2 and 4.4.)

Finally the user can perform global GLB for a fuzzy knowledge base. Global GLB service of FiRE evaluates the greatest lower bound degree of all the concepts of Σ participating in all the individuals of Σ.

3.3 Storage of a Fuzzy Knowledge Base

FiRE was enhanced by the functionalities of the RDF store Sesame (Sesame 2 beta 6). Sesame is an open source Java framework for storing and querying RDF/RDFS data. It supports two ways of storing RDF, called RDF Repositories. The first is the In-memory RDF Repository, which stores all the RDF triples in main memory, while the Native RDF Repository stores the RDF triples on the hard disk and uses B-Trees to index and access them.

In the proposed architecture the RDF store is used as a back end for storing and querying RDF triples in a sufficient and convenient way while the reasoner is the front end, which the user can use in order to store and query a fuzzy knowledge base. In that way, a user is able to access data from a repository, apply any of the available reasoning services on this data and then store back in the repository the implicit knowledge extracted from them.

In order to use Sesame for storing a fuzzy knowledge without enforcing any extensions to it, a way of serializing fuzzy knowledge into RDF triples was neccesary. For that purpose a fuzzy-OWL to RDF mapping was required, similar to the one provided in the OWL abstract syntax and semantics document [23]. In [17] the authors use RDF *reification*, in order to store membership degrees. However it is well-known that reification has weak, ill-defined model theoretic semantics and its support by RDF tools is doubtful. Another approach [31] suggests the use of datatypes but the use of a concrete feature like datatypes to represent abstract information such as fuzzy assertions is not appropriate.

These limitations lead us to propose a cleaner way that is based on the use of blank nodes. First, we define three new entities, namely `frdf:membership`, `frdf:degree` and `frdf:ineqType` as types (i.e. `rdf:type`) of `rdf:Property`.

Using these new properties together with blank nodes we can represent fuzzy instances. Let's assume for example that we want to represent the assertion $\langle (region1 : Sky) \geq 0.74\rangle$. The RDF triples representing this information are the following:

`region1`	`frdf:membership`	`_:region1membSky` .
`_:region1membSky`	`rdf:type`	`Sky` .
`_:region1membSky`	`frdf:degree`	"`0.74^^xsd:float`" .
`_:region1membSky`	`frdf:ineqType`	"=" .

where `_:regionmembSky` is a blank node used to represent the fuzzy assertion of `region1` with the concept Sky.

On the other hand mapping fuzzy role assertions is more tricky since RDF does not allow for blank nodes in the predicate position. Thus, new properties are used for each assertion, such that assertion $\langle (region1, region2) : aboveOf \geq 0.8\rangle$ is mapped to

`region1`	`frdf:r1r2above-Of`	`region2` .
`frdf:r1r2above-of`	`rdf:type`	`above-of` .
`frdf:r1r2above-of`	`frdf:degree`	"`0.8^^xsd:float`" .
`frdf:r1r2above-of`	`frdf:ineqType`	"=" .

3.4 Querying an Expressive Fuzzy Knowledge Base

3.4.1 Fuzzy queries

One of the main advantages of persistent storage systems, like relational databases and RDF storing systems, is their ability to support *conjunctive queries*. Conjunctive queries generalize the classical inference problem of *realization* of Description Logics [6], i.e. "get me all individuals of a given concept C", by allowing for the combination (conjunction) of concepts and roles. Formally, a conjunctive query is of the following form:

$$q(X) \leftarrow \exists Y.conj(X, Y) \tag{4}$$

or simply $q(X) \leftarrow conj(X, Y)$, where $q(X)$ is called the head, $conj(X, Y)$ is called the body, X are called the *distinguished variables*, Y are existentially quantified variables called the *non-distinguished variables*, and $conj(X, Y)$ is a conjunction of atoms of the form $\mathsf{A}(v)$, $R(v_1, v_2)$, where A, R are respectively concept and role names, v, v_1 and v_2 are *individual* variables in X and Y or individuals from the ontology.

Since in our case we extend classical assertions to fuzzy assertions, new methods of querying such fuzzy information are possible. More precisely, in [22] the authors extend ordinary conjunctive queries to a family of significantly more expressive query languages, which are borrowed from the fields of fuzzy information retrieval [10]. These languages exploit the membership degrees of fuzzy assertions by introducing weights or thresholds in query atoms. In particular, the authors first define *conjunctive threshold queries* (CTQs) as:

$$q(X) \leftarrow \exists Y. \bigwedge_{i=1}^{n} (atom_i(X,Y) \geq k_i) \tag{5}$$

where $k_i \in [0,1]$, $atom_i(X,Y)$ represents either a fuzzy-DL concept or role and all $k_i \in (0,1]$ are thresholds. As it is obvious those answers of CTQs are a matter of true or false, in other words an evaluation either is or is not a solution to a query. The authors also propose *General Fuzzy Conjunctive Queries* (GFCQs) that further exploit fuzziness and support degrees in query results. The syntax of a GFCQ is the following:

$$q(X) \leftarrow \exists Y. \bigwedge_{i=1}^{n} (atom_i(X,Y) : k_i) \tag{6}$$

where $atom_i(X,Y)$ is as above while k_i is the degree associated weight. As shown in [22], this syntax is general enough to allow various choices of semantics, which emerge by interpreting differently the degree of each fuzzy-DL atom $(atom_i(X,Y))$ with the associated weight (k_i).

A straightforward extension will be the use of fuzzy threshold with the aid of fuzzy R-implications instead of the crisp one. Hence, a t-norm (t) as the semantic function for conjunctions and R-implications (ω_t) as the semantic function for degree-associated atoms, we get fuzzy threshold queries, in which the degree of truth of q_F under $\mathcal{I}$ is

$$d = \sup_{S' \in \Delta^{\mathcal{I}} \times \cdots \times \Delta^{\mathcal{I}}} \{ t_{i=1}^{n}\ \omega_t(k_i, atom_i^{\mathcal{I}}(\bar{v})_{[X \mapsto S, Y \mapsto S']}) \}. \tag{7}$$

Given some S', if for all atoms we have $atom_i^{\mathcal{I}}(\bar{v})_{[X \mapsto S, Y \mapsto S']} \geq k_i$, since $\omega_t(x,y) = 1$ when $y \geq x$ [14], we have $d = 1$; this corresponds to threshold queries introduced earlier.

Another example of semantics for GFCQs would be to use fuzzy aggregation functions [14]. For example, let $G(x) = \sum_{i=1}^{n}\ x_i$ as a function for conjunctions and $a(k_i, y) = \frac{k_i}{\sum_{i=1}^{n}\ k_i} * y$ as the semantic function for degree-associated atoms. Then we get an instance of fuzzy aggregation queries, in which the degree of truth of q_F under $\mathcal{I}$ is

$$d = \sup_{S' \in \Delta^{\mathcal{I}} \times \cdots \times \Delta^{\mathcal{I}}} \frac{\sum_{i=1}^{n}\ k_i * atom_i^{\mathcal{I}}(\bar{v})_{[X \mapsto S, Y \mapsto S']}}{\sum_{i=1}^{n}\ k_i}. \tag{8}$$

Example of such queries are given in Section 4, while more details on their semantics is beyond the scope of this paper and the interested reader should refer to [22].

3.4.2 Fuzzy queries using FiRE

These queries were implemented by the use of the query language SPARQL [24] that is supported by Sesame. The user can perform them using the *Queries* inference tab, and in the case of generalized fuzzy conjunctive queries a choice of fuzzy threshold or fuzzy aggregation semantics is possible.

Example 2 A threshold query that reveals its FiRE syntax follows:

```
x,y <- Beach(x) >= 0.4 ^ contains(x,y) >= 1.0
                                ^ Person(y) >= 0.8
```

Queries consist of two parts: the first one specifies the individuals that will be evaluated while the second one states the condition that has to be fulfilled for the individuals. This query asks for individuals x and y; x has to participate in concept Beach to at least the given degree and it also has to be the subject of a `contains` assertion with participation greater than 1.0, having as a role-filler individual y that has to participate in concept Person to at least the given degree.

The query is firstly converted from the FiRE conjunctive query syntax to the SPARQL query language. Based on the fuzzy OWL syntax in triples that we have defined in Section 3.3 the query of Example 2 is as follows in SPARQL. The query results are evaluated by the Sesame engine and visualized by FiRE.

```
SELECT ?x WHERE {
    ?x ns5:membership ?Node1 .
    ?Node1 rdf:type ?Concept1 .
    ?Node1 ns5:ineqType ?IneqType1 .
    ?Node1 ns5:degree ?Degree1 .
    FILTER regex (?Concept1 , "CONCEPTS#Beach")
    FILTER regex (?IneqType1 ,">")
    FILTER (?Degree1 >= "0.8^^xsd:float")

    ?BlankRole2 ns5:ineqType ?IneqType2 .
    ?BlankRole2 ns5:degree ?Degree2 .
    ?BlankRole2 rdf:type ?Role2 .
    ?x BlankRole2 ?y .
    FILTER regex (?Role2 , "ROLES#contains")
    FILTER regex (?IneqType1 ,">")
    FILTER (?Degree2 >= "1^^xsd:float")
    ...
}
```

Example 3 We can issue a GFCQ by using the symbol ":" followed by the importance of participation for each condition statement instead of inequality types. Hence we can get all beach images that contain the concepts Beach and Person and rank higher those with larger degree for the latter:

```
x <- Beach(x):0.6 ^ contains(x,y) : 1 ^ Person(y) : 0.8
```

In the case of GGCQs the operation is different. The SPARQL query is constructed in a way that retrieves the participation degrees of every Role or Concept used in the atoms criteria, for the results that satisfy all of the atoms. The participation degrees retrieved for each query atom together with its weight are then used by FiRE for the ranking procedure of the results based on the selected semantics. An excerpt of the SPARQL query for Example 3 follows.

```
SELECT ?x ?Degree1...
WHERE {
    ?x ns5:membership ?Node1 .
    ?Node1 rdf:type ?Concept1 .
    ?Node1 ns5:ineqType ?IneqType1 .
    ?Node1 ns5:degree ?Degree1 .
```

```
        FILTER regex (?Concept1 , "CONCEPTS#Beach")
        FILTER regex (?IneqType1 ,">")
        FILTER (?Degree1 >= "0.0^^xsd:float")

        ...
    }
```

4 Experimental Evaluation

4.1 Image Dataset and Evaluation Procedure

In this section an examination of the proposed architecture is presented. The dataset consists of 1000 images from authors' personal collection and the Internet. All these images in general are characterized as personal content, i.e. photos that usually everyone may have taken from vacations, varying from beach to mountainous landscapes and to city touring. The dataset is accompanied by region-based ground truth, i.e. for every region we know which concept it depicts (such type of ground truth requires significant labour effort to produce it, even for 1000 images). Firstly, every image is processed by the semantic segmentation algorithm, the output of which comprise the ABox of a domain specific fuzzy knowledge base. After that, FiRE using an expressive domain terminology extracts implicit information and also refines the region labels assigned by analysis. The resulting information for image regions together with the inferred abstract concepts on the global image are stored in a Sesame repository. In that way a user can perform the various queries presented in Section 3.4 for ranking and retrieval of specific personalized content. We first go through the fuzzy knowledge base presenting the refinement operation on region classification. After that an evaluation of the semantic segmentation is made and finally the usage of semantic queries is illustrated by a set of examples.

4.2 The Fuzzy Knowledge Base

In order to effectively categorize images and also improve the semantic segmentation process we have implemented an expressive terminology. The terminology includes definition of new concepts that characterize an image and also of concepts that refine information extracted by the classification module considering the spatial relations of the labeled regions.

For better comprehension of the defined terminology, we first present the input used as the assertional part of the fuzzy knowledge base provided by the analysis module. The possible extracted concepts and roles that comprise the alphabet of concepts **C** and roles **R** are shown below:

C = {*Sky Building Person Rock Tree Vegetation Sea Grass Ground Sand Trunk Dried-plant Pavement Boat Wave*}

R ={*above-of below-of left-of right-of contains*}

The set of individuals consist of the amount of segments obtained for each image together with the global image.

Example 4 An excerpt of ABox for an image is:

$$\langle image1\_seg01 : Sky \geq 0.744\rangle,$$
$$\langle image1\_seg01 : Person \geq 0.514\rangle$$
$$\langle image1\_seg03 : Sky \geq 0.831\rangle$$
$$\langle image1\_seg09 : Sea \geq 0.557\rangle,$$
$$\langle image1\_seg09 : Person \geq 0.510\rangle,$$
$$\ldots$$
$$\langle (image1, image1\_seg01) : contains \geq 1\rangle,$$
$$\langle (image1, image1\_seg03) : contains \geq 1\rangle,$$
$$\langle (image1, image1\_seg09) : contains \geq 1\rangle,$$
$$\ldots$$
$$\langle (image1\_seg01, image1\_seg09) : above\text{-}of \geq 1\rangle,$$
$$\ldots$$

An excerpt of $TBox$ can be found in Table 2. The knowledge base consists of two layers. The first, or lower layer, includes concepts like Sea that are extracted by the classification module and have been re-defined using spatial relations. This part of the knowledge is only used together with the S-RSST for region refinement (those concepts are shown in capitals).

Since the classifier is trained, we assume a correct estimation of the label but with a possibly mistaken confidence value. So, SEA has been defined as a segment that was originally assigned as Sea by the SVM classifier, while its neighboring regions can be either one of Wave, Sea or Sky. In this case, a variation of the greatest lower bound (GLB) reasoning service is used for the semantic refinement of the labels provided by the SVM classifier. Hence, we first compute the GLB of the region of interest to the concept of interest (i.e. SEA). We then evaluate the GLB of the region of interest to the neighbor criterion concept of the concept of interest (if SEA is the concept of interest then neighbor criterion concept is $((\exists \textit{right-of}.(\mathsf{Sand} \sqcup \mathsf{Wave})) \sqcup (\exists \textit{left-of}.(\mathsf{Sand} \sqcup \mathsf{Wave})) \sqcup ...)$. If this bound is greater than the initial GLB, (i.e the value that was originally assigned to that concept) then the region value is refined to the neighbor criterion concept GLB, otherwise it remains as assigned.

Let's assume the ABox of Example 4. The segment of examination is $image1\_seg09$ that has been classified as Sea with degree greater or equal to 0.557. Segment $image1\_seg01$ is *above-of* segment $image1\_seg09$ but since role *above-of* has as inverse role *below-of* (Table 4.2) it is equivalent saying that $image1\_seg09$ is *below-of* segment $image1\_seg01$. Furthermore $image1\_seg01$ that has been classified as Sky with degree greater or equal to 0.744. Clearly the criteria for $image1\_seg09$ to participate in SEA are satisfied and its GLB is 0.557, equal to the degree that was originally assigned by the classifier. This fact means that the neighbor criteria concepts (i.e. $\sqcap((\exists \textit{right-of}.(\mathsf{Sea} \sqcup \mathsf{Wave}))...)$ of the axiom defined for concept SEA are also fulfilled with a degree at least equal to 0.557. Hence we evaluate the GLB of $image1\_seg09$ to the neighbor criterion concept that is 0.744 and we refine participation of segment $image1\_seg09$ in concept SEA to this value (note that if the Sky value was 0.3 the Sea value would have remained as originally assigned).

The refined results comprise the ABox of the second layer knowledge base, which includes higher-level concepts. Such concepts refer to a segment like WavySea and SandyBeach but also to global image classification like Beach. Concerning notation, concepts here are not in capital because this knowledge is used after region-refinement and hence the refined ABox component is again given by S-RSST.

Table 2 An excerpt of the terminology $TBox$.

$\mathcal{T} = \{\mathsf{SEA}$	$\equiv$	$\mathsf{Sea} \sqcap ((\exists right\text{-}of.(\mathsf{Sea} \sqcup \mathsf{Wave})) \sqcup (\exists left\text{-}of.(\mathsf{Sea} \sqcup \mathsf{Wave}))$
		$\sqcup(\exists above\text{-}of.(\mathsf{Sea} \sqcup \mathsf{Wave})) \sqcup (\exists below\text{-}of.(\mathsf{Sea} \sqcup \mathsf{Wave} \sqcup \mathsf{Sky}))),$
SAND	$\equiv$	$\mathsf{Sand} \sqcap ((\exists right\text{-}of.(\mathsf{Sand} \sqcup \mathsf{Wave})) \sqcup (\exists left\text{-}of.(\mathsf{Sand} \sqcup \mathsf{Wave}))$
		$\sqcup(\exists above\text{-}of.(\mathsf{Sand} \sqcup \mathsf{Wave})) \sqcup (\exists below\text{-}of.(\mathsf{Sand} \sqcup \mathsf{Wave} \sqcup \mathsf{Sea}))),$
WAVE	$\equiv$	$\mathsf{Wave} \sqcap (\exists right\text{-}of.(\mathsf{Sea} \sqcup \mathsf{Wave})) \sqcup (\exists left\text{-}of.(\mathsf{Sea} \sqcup \mathsf{Wave}))$
		$\sqcup(\exists above\text{-}of.(\mathsf{Sea} \sqcup \mathsf{Wave})) \sqcup (\exists below\text{-}of.(\mathsf{Sea} \sqcup \mathsf{Wave}))),$
$\mathsf{WavySea}$	$\equiv$	$\mathsf{Sea} \sqcap \mathsf{Wave},$
$\mathsf{SandyBeach}$	$\equiv$	$\mathsf{Sea} \sqcap \mathsf{Sand},$
$\mathsf{PartOfComplexBuilding}$	$\equiv$	$\mathsf{Building} \sqcap (\exists left\text{-}of.\mathsf{Building} \sqcup \exists right\text{-}of.\mathsf{Building}),$
Beach	$\equiv$	$\exists \mathsf{contains}.\mathsf{Sea} \sqcap \exists \mathsf{contains}.\mathsf{Sky},$
$\mathsf{Landscape}$	$\equiv$	$\exists \mathsf{contains}.\mathsf{Vegetation},$
City	$\equiv$	$\exists \mathsf{contains}.\mathsf{Building} \sqcup \exists \mathsf{contains}.\mathsf{Pavement}\}$
$\mathcal{R} = \{\mathsf{contains}, left\text{-}of^{-} = right\text{-}of, above\text{-}of^{-} = below\text{-}of\}$		

According to the defined terminology implicit knowledge is extracted. For every image the GLB reasoning service is used for the defined concepts of the terminology. The obtained implicit results together with the explicit information provided by the classifier (i.e. $ABox$) are stored to a Sesame repository. To better understand the operation of GLB let's assume the refined $ABox$ previously described. After the region refinement procedure we have assertions $\langle image1\_seg01 : Sky \geq 0.744\rangle$, $\langle image1\_seg03 : Sky \geq 0.831\rangle$, and $\langle image1\_seg09 : Sea \geq 0.744\rangle$. Also regarding role assertions we have $\langle (image1, image1\_seg01) : contains \geq 1\rangle$, $\langle (image1, image1\_seg03) : contains \geq 1\rangle$ and $\langle (image1, image1\_seg09) : contains \geq 1\rangle$. Within our knowledge base, Beach has been defined as an image that contains segments labeled as Sky and Sea. Clearly the criteria for $image1$ to participate in concept Beach are fulfilled. Hence $image1$ is classified as Beach with the greatest degree that satisfies these criteria, being 0.744. This is because Beach was defined using conjunction operator that is interpreted with min (Table 1). So despite $\langle image1\_seg03 : Sky \geq 0.831\rangle$ the minimum limit is set by$\langle image1_seg09 : Sea \geq 0.744\rangle$ so we end up with $\langle image1 : Beach \geq 0.744\rangle$.

Figure 3 presents the obtained results for some images, the initial output of the S-RSST and the semantically refined regions merged by S-RSST using FiRE.

4.3 Evaluation of Semantic Segmentation

In order to evaluate the performance of the semantic segmentation (S-RSST), we compare the recognition rates against a baseline technique: We used a color-based RSST segmentation to partition the image and trained an SVM classifier for each concept $C_i \in \mathbf{C}$. Each classifier returns a degree of confidence for every region and precision/recall metrics are calculated. The same metrics are computed for the output of the semantic segmentation, which as already discussed, performs image segmentation and region classification at the same time. Some indicative examples of this evaluation scenario are illustrated in Figure 3, where column (b) displays the result of the baseline technique and column (c) the results of semantic segmentation. The latter, in most cases, continues merging of regions based on their attached semantic labels, resulting in a less segmented output and refined degrees of those labels.

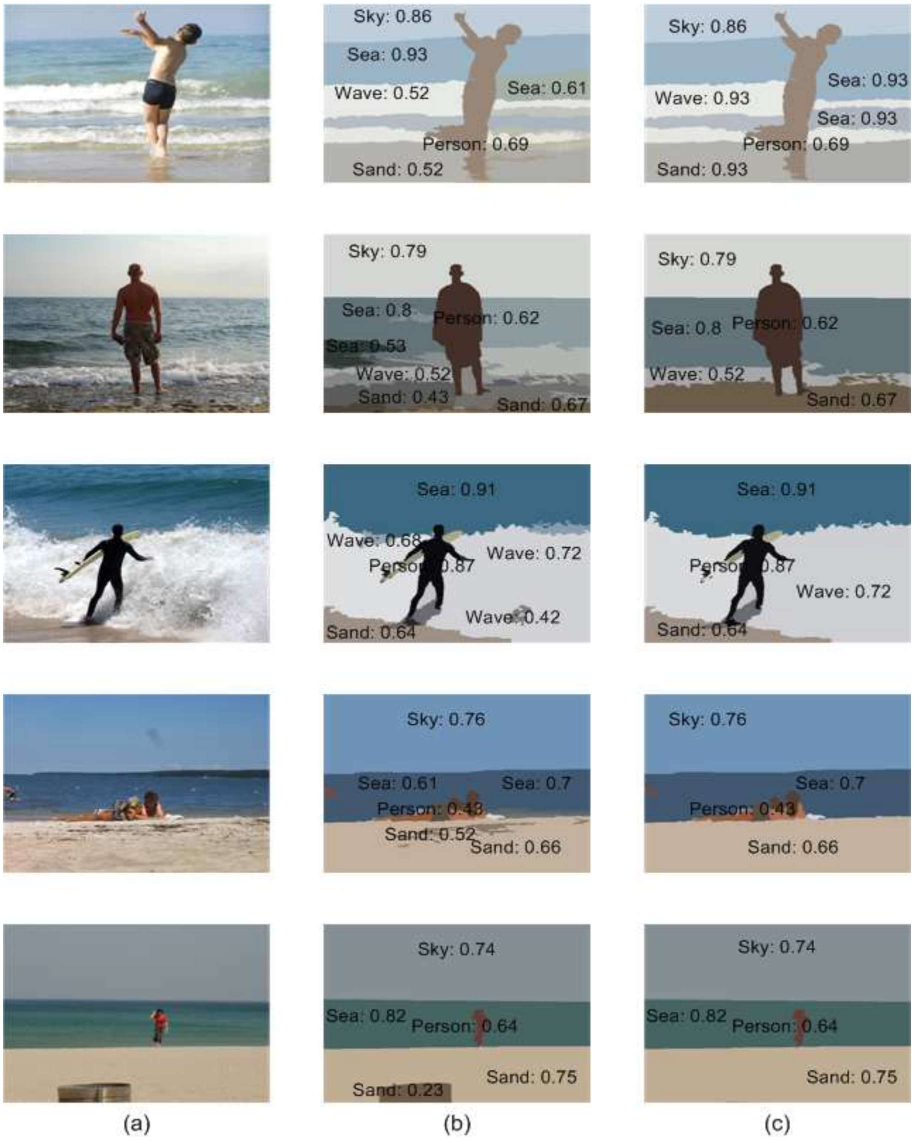
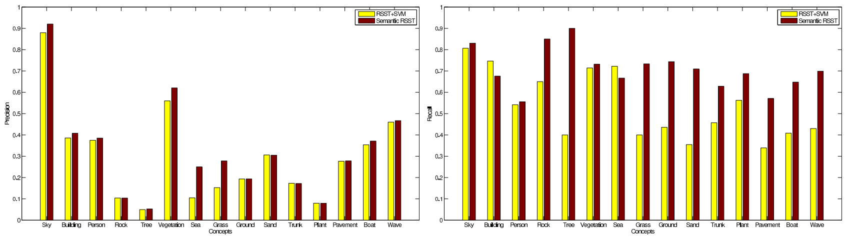


Fig. 3 (a) Original Image (b) Original Segmentation (c) Semantic segmentation using FiRE.

An important issue that was raised, was how to deal with the fuzzy degrees of confidence resulting from the semantic segmentation algorithm and how to compute the precision and recall figures given the available ground truth. We employed a simple idea, where for every concept a different defuzzification scheme was used, based on the histogram of confidence values for the whole dataset. In consequence, we considered that concepts were detected only when their degree of confidence was greater to that certain (concept-specific) threshold. Figure 4 illustrates the (a) precision and (b) recall values for all 15 concepts.



(a) (b)

Fig. 4 (a) Precision and (b) Recall for the different classification results for every concept.

Additionally we calculated the overall performance of the above two techniques (RSST + SVM, Semantic RSST), irrespectively of the concept, using a weighted average of precision and recall values of each concept. Each concept's weight depends on the frequency of appearance of that concept in the dataset according to the ground truth. Moreover, the weighted harmonic mean (F-measure) of precision and recall was calculated to measure the effectiveness of the classification (Table 3).

Table 3 Average classification results for all concepts.

Technique	Precision	Recall	F-measure
RSST+SVM	0.40	0.64	0.49
Semantic RSST	0.43	0.74	0.55

4.4 Retrieval Examples

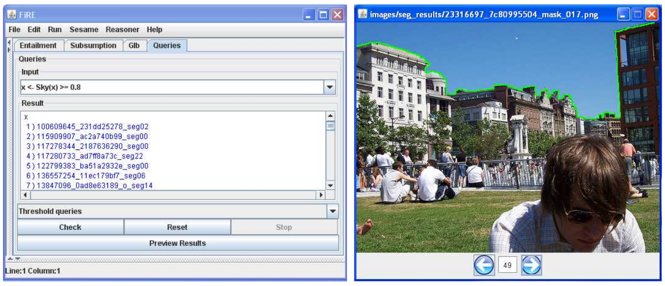
The main benefit of the FiRE integration with Sesame is the ability to perform Conjunctive Threshold Queries (CTQs) and General Fuzzy Conjunctive Queries (GFCQs). The extracted implicit knowledge together with refined region results are stored in a Sesame repository. The user can perform fuzzy conjunctive threshold queries like the one in Example 5 using the FiRE query interface shown in Figure 5.

Example 5

```
x,y <- Beach(x) >= 0.4 ^ contains(x,y) >= 1.0
                          ^ Person(y) >= 0.8
```

The semantic meaning of the degrees in such a query is inherited by their original production by the SVM classifier. In other words when the user requires that $\mathsf{Person}(y) >= 0.8$, he or she requires a segment y that was evaluated by the SVM classifier as Person with a degree of confidence greater than 0.8. Due to the inaccurate nature of most classifiers only few segments have been classified to a label with degree 1. Hence $\mathsf{Person}(y) >= 0.8$ is very likely to be a person while $\mathsf{Person}(y) >= 0.5$ is possible to be a blurry person.

It is important to note that the proposed representation allows not only image retrieval, but also retrieval of segments with a specific label. Hence, on the left of Figure 5 the query panel is shown that consists of the query input and the retrieved results, while on the right the user can browse the retrieved images or segments.



(a) (b)

Fig. 5 FiRE query interface.

Figure 6 presents the first 8 images that satisfy the query of Example 5. As mentioned in Section 3.4.1 the answers to CTQs queries are a matter of true or false, in other words an evaluation either is or is not a solution to a query. On the other hand these images can also be retrieved using GFCQs that offer a ranking of the results. The query syntax is of the form:

Example 6
```
x <- Beach(x):0.6 ^ contains(x,y) : 1
                  ^ Person(y) : 0.8
```

A weight criterion is assigned to every atom of the query, thus an image with a relevance rank is retrieved. The ranking varies on the chosen semantics.

Table 4 illustrates the ranking for these images. Beach has been defined as an image that contains Sea and Sky. For that purpose we present the participation degrees of these concepts and the resulting degree for Beach that is their minimum degree based on the semantics of fuzzy conjunction using GLB. In a similar manner we present the degrees of Person and since the relation assertions are crisp, (i.e. they have degree of participation equal to 1) we evaluate the minimum degree of Person and Beach as a ranking criterion for the CFQs while columns FT and FA illustrate the rankings of the query of Example 6 using *fuzzy threshold* and *fuzzy aggregation* GFCQs respectively.

Fuzzy threshold GFCQs use a fuzzy threshold with the aid of fuzzy R-implications, instead of a crisp threshold. Hence, the ranking evaluated by this kind of queries corresponds to the difference of the degrees of participation in the concepts/roles of the query atoms from the corresponding weight criteria. So, images (g), (b) and (d) in Table 4 that have degrees of participation in the atoms greater than the corresponding weight criteria are ranked with 1.

On the other hand, *fuzzy aggregation* GFCQs rank images according to fuzzy aggregation functions. Hence the weight criterion in this case operates as a coefficient which is used together with the degrees of participation in the concepts-roles of the query atoms to rank the results. Observing fuzzy aggregation ranking for images (b) and (d) we can see a significant difference in the Beach values. However, their ranking values are close due to the Person values that are assumed to be more important since they are weighted higher in the query of Example 6.

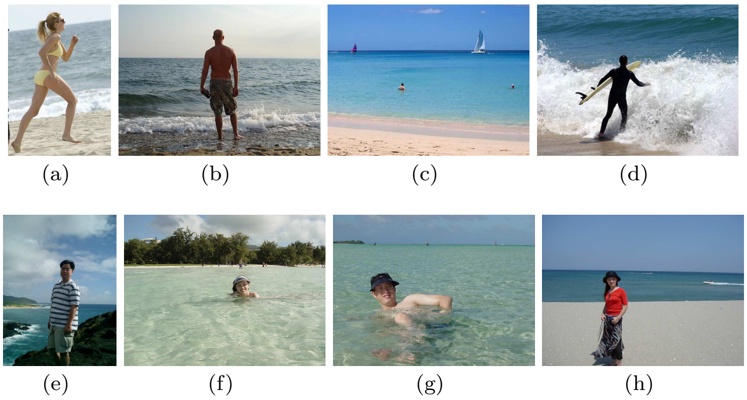


Fig. 6 Images retrieved for a ”person in beach” query.

Table 4 ”Person in beach” query ranking of Examples 5 and 6.

Image	Sea	Sky	Beach	Person	**CTQ**	**GFCQ**	
						FT	**FA**
(g)	0.90	0.87	0.87	0.86	0.87	1.00	0.92
(b)	0.83	0.91	0.83	0.80	0.80	1.00	0.89
(d)	0.69	0.66	0.66	0.81	0.66	1.00	0.85
(h)	0.60	0.83	0.60	0.85	0.60	1.00	0.84
(a)	0.47	0.92	0.47	0.84	0.47	0.87	0.81
(f)	0.63	0.47	0.47	0.85	0.47	0.87	0.81
(c)	0.46	0.91	0.46	0.86	0.46	0.86	0.81
(e)	0.44	0.73	0.44	0.83	0.44	0.84	0.80

5 Conclusions

In this paper we have presented an automatic semantic annotation methodology consisting of several novel and state of the art techniques. An image is initially processed by a knowledge-assisted analysis module which produces a segmentation mask and region-associated concepts. A semantic segmentation methodology incorporating object detection simultaneously with region merging, provides input for region-based classification resulting in a list of concepts (together with degrees of confidence for each one) that have been linked to the image. The expressive fuzzy DL f -$\mathcal{SHIN}$ and the reasoning engine FiRE that supports it, can use this information for the extraction of additional, implicit knowledge, improvement of region-based classification by incorporating spatial relations and neighborhood information and finally inferring of abstract concepts on a global image basis. The final results are stored in an online semantic repository and in cooperation with FiRE user can perform semantic retrieval and ranking of images.

Experiments confirmed the usefulness and potential of our approach and based on the satisfactory evaluation results we believe that further improvements towards this direction can be made for efficient and effective multimedia indexing. Furthermore the use of expressive fuzzy description logics together with fuzzy queries reveals semantic ranking of images. The various rankings provided depend on the semantics used by fuzzy queries and can be interpreted in different ways according to the application scenario. Finally the proposed architecture can be easily adapted to video sequences permitting semantic browsing of scenes.

References

1. Description-logic knowledge representation system specification from the krss group of the arpa knowledge sharing effort. `http://dl.kr.org/krss-spec.ps`.
2. Hoogs A., Rittscher J., Stein G., and Schmiederer J. Video content annotation using visual analysis and a large semantic knowledgebase. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2003.
3. T. Adamek, N.O'Connor, and N.Murphy. Region-based segmentation of images using syntactic visual features. In *Proc. Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS 2005*, Montreux, Switzerland, April 13-15 2005.
4. Th. Athanasiadis, Ph. Mylonas, Y. Avrithis, and S. Kollias. Semantic image segmentation and object labeling. *IEEE Trans. on Circuits and Systems for Video Technology*, 17(3):298–312, 2007.
5. Th. Athanasiadis, V. Tzouvaras, K. Petridis, F. Precioso, Y. Avrithis, and Y. Kompatsiaris. Using a multimedia ontology infrastructure for semantic annotation of multimedia content. In *Proceedings of the 5th International Workshop on Knowledge Markup and Semantic Annotation*, 2005.
6. F. Baader, D. McGuinness, D. Nardi, and P.F. Patel-Schneider. *The Description Logic Handbook: Theory, implementation and applications.* Cambridge University Press, 2002.
7. S. Berretti, A. Del Bimbo, and E. Vicario. Efficient matching and indexing of graph models in content-based retrieval. *IEEE Trans. on Circuits and Systems for Video Technology*, 11(12):1089–1105, Dec. 2001.
8. Eran Borenstein and Eltan Sharon. Combining top-down and bottom-up segmentation. In *in 8th Conference on Computer Vision and Pattern Recognition Workshop, CVPR 2004*, 2004.
9. M. G. Christel and A. G. Hauptmann. The use and utility of high-level semantic features in video retrieval. In *In Proceedings of 4th International Conference on Image and Video Retrieval, CIVR 2005*, Singapore, July 2005.
10. V. Cross. Fuzzy information retrieval. *Journal of Intelligent Information Systems*, 3:29–56, 1994.
11. I. Kompatsiaris G. Papadopoulos, V. Mezaris and M. Strintzis. Combining global and local information for knowledge-assisted image analysis and classification. *EURASIP Journal on Advances in Signal Processing, Special Issue on Knowledge-Assisted Media Analysis for Interactive Multimedia Applications*, accepted for publication, 2007.
12. Rosalba Giugno and Thomas Lukasiewicz. P-shoq(d): A probabilistic extension of shoq(d) for probabilistic ontologies in the semantic web. In *JELIA '02: Proceedings of the European Conference on Logics in Artificial Intelligence*, pages 86–97, London, UK, 2002. Springer-Verlag.
13. I. Horrocks, U. Sattler, and S. Tobies. Reasoning with Individuals for the Description Logic $\mathcal{SHIQ}$. In David MacAllester, editor, *CADE-2000*, number 1831 in LNAI, pages 482–496. Springer-Verlag, 2000.
14. G. J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications.* Prentice-Hall, 1995.
15. M. P. Kumar, P. H. S. Torr, and A. Zisserman. OBJ CUT. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego*, 2005.
16. Hollink L., Worring M., and Schreiber G. Building a visual ontology for video retrieval. In *Proceedings of the ACM Multimedia*, 2005.
17. Mauro Mazzieri and Aldo Franco Dragoni. A fuzzy semantics for semantic web languages. In *ISWC-URSW*, pages 12–22, 2005.
18. O.J. Morris, M.J. Lee, and A.G. Constantinides. Graph theory for image analysis: An approach based on the shortest spanning tree. In *Inst. Elect. Eng., vol. 133, pp. 146-152*, 1986.
19. Phivos Mylonas, Thanos Athanasiadis, Manolis Wallace, Yannis Avrithis, and Stefanos Kollias. Semantic representation of multimedia content: Knowledge representation and semantic indexing. *Multimedia Tools and Applications*, in press.

20. M. Naphade and T. S. Huang. A probabilistic framework for semantic video indexing, filtering and retrieval. *IEEE Trans. on Multimedia*, 3(1):144–151, March 2001.
21. M. Naphade, J. Smith, J. Tesic, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis. Large-scale concept ontology for multimedia. *IEEE Multimedia*, 13(3):86–91, March 2006.
22. J.Z. Pan, G. Stamou, G. Stoilos, and E. Thomas. Expressive querying over fuzzy DL-Lite ontologies. In *Proceedings of the International Workshop on Description Logics (DL 2007)*, 2007.
23. Peter F. Patel-Schneider, Patrick Hayes, and Ian Horrocks. OWL Web Ontology Language Semantics and Abstract Syntax. Technical report, W3C, Feb. 2004. W3C Recommendation, URL http://www.w3.org/TR/2004/REC-owl-semantics-20040210/.
24. E. Prud'hommeaux and A. Seaborne. SPARQL query language for RDF, 2006. W3C Working Draft, http://www.w3.org/TR/rdf-sparql-query/.
25. Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and trecvid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.
26. John R. Smith. *Video indexing and retrieval using MPEG-7.* CRC Press, 2004.
27. C. Snoek, B. Huurninkm, L. Hollink, M. de Rijke, G. Schreiber, and M. Worring. Adding semantics to detectors for video retrieval. *IEEE Trans. on Multimedia*, 9(5):144–151, August 2007.
28. Cees G. M. Snoek, Bouke Huurnink, Laura Hollink, Maarten de Rijke, Guus Schreiber, and Marcel Worring. Adding semantics to detectors for video retrieval. *IEEE Transactions on Multimedia*, 9(5):975986, August 2007.
29. Giorgos Stoilos, Giorgos Stamou, Vassilis Tzouvaras, Jeff Z. Pan, and Ian Horrocks. Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research*, 30(5):273–320, 2007.
30. U. Straccia. Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.
31. V. Vaneková, J. Bella, P. Gurský, and T. Horváth. Fuzzy RDF in the semantic web: Deduction and induction. In *Proceedings of Workshop on Data Analysis (WDA 2005)*, pages 16–29, 2005.
32. Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001.
33. L. Zhang, F. Lin, and B. Zhang. Support vector machine learning for image retrieval. *Image Processing, 2001. Proceedings. 2001 International Conference on*, 2, 2001.