A Framework for the Retrieval of Multimedia Objects Based on Four-Valued Fuzzy Description Logics

Umberto Straccia

I.E.I - C.N.R.,
Via S. Maria, 46,
I-56126 Pisa, Italy

Abstract. Knowledge representation, in particular logic, combined together with database and information retrieval techniques may play an important role in the development of so-called intelligent multimedia retrieval systems. In this paper we will present a logic-based framework in which multimedia objects' medium dependent properties (objects' low level features) and multimedia objects' medium independent properties (abstract objects' features, or objects' semantics) are addressed in a principled way. The framework is logic-based as it relies on the use of a four-valued fuzzy Description Logics for (*i*) representing the semantics of multimedia objects and (*ii*) for defining the retrieval process in terms of logical entailment. Low level features are not represented explicitly within the logic, but may be addressed by means of procedural attachments over a concrete domain. Description Logics are object-oriented representation formalisms capturing the most popular features of structured representation of knowledge. They are a good compromise between computational complexity and expressive power and, thus, may be seen as a promising tool within the context of logic-based multimedia information retrieval.

1 Introduction

In the last decade a substantial amount of work has been carried out in the context of *Description Logics* (DLs)[1]. DLs are a logical reconstruction of the so-called frame-based knowledge representation languages, with the aim of providing a simple well-established Tarski-style declarative semantics to capture the meaning of the most popular features of structured representation of knowledge. *Concepts, roles, individuals, assertions* and *specialisations* are the building blocks of these logics. *Concepts* are expressions which collects the properties, described by means of roles, of a set of individuals. From a logical point of view, concepts can be seen as unary predicates, interpreted as sets of elements over a domain. Examples of concepts are Team and Person. *Roles* are considered as binary predicates which are interpreted as binary relations between the elements over a domain. An example of role is Member which

[1] Description Logics have also been referred to as Terminological Logics, Concept Logics, KL-ONE-like languages. The web page of the description logic community is found at address `http://dl.kr.org/dl`.

may represent a relation between a team and the persons belonging to the team. *Individuals* are interpreted as elements in the domain. For instance, an individual represents a member of a team.

In order to build a knowledge base one starts with the definition and construction of the taxonomy of concepts, i.e. the schema, by means of *specialisations* (denoted by $C \Rightarrow D$, where C, D are concepts). An example of specialisation is $\mathsf{DreamTeam} \Rightarrow \mathsf{Team} \sqcap \forall \mathsf{Member}.\mathsf{SuperStar}$ which specifies that a DreamTeam is a Team such that each Member is a SuperStar. Information about individuals is stated through *assertions*. An assertion states either that an individual a is an instance of a concept C (denoted by $C(a)$) or that two individuals a and b are related by means of a role R (denoted by $R(a, b)$). Examples of assertions are DreamTeam(chicago) and Member(chicago,jordan). A DL *Knowledge Base* (KB) (denoted by $\mathcal{K}$) is a set of specialisations and assertions. A basic inference task with knowledge bases is *instance checking* and amounts to verify whether the individual a is an instance of the concept C with respect to the knowledge base $\mathcal{K}$, i.e. in symbols $\mathcal{K} \models C(a)$.

DL systems has been used for building a variety of applications including (see [14]) systems supporting software management [13], browsing and querying of networked information sources [16], knowledge mining [4], data archaeology [9], planning [35], learning [22], natural language understanding [7], clinical information system [18], digital libraries [36], software configuration management system [39] and web source integration [20]. DLs are considered as to be attractive logics in knowledge based applications as they are a good compromise between expressive power and computational complexity.

In this paper we will present a DL based *Multimedia Information Retrieval* (MIR) model. Any principled approach to the description of such a model requires the formal specification of three basic entities of retrieval: (i) the representation of multimedia objects; (ii) the representation (called query) of a user information need; and (iii) the retrieval function, returning a ranked list of objects for each information need. We believe that any MIR model should address the multidimensional aspect of multimedia objects: that is, their *form* and their *semantics* (or *meaning*). The form of a multimedia object is a collective name for all its *media dependent* features, like text index term weights (object of type text), colour distribution, shape, texture, spatial relationships (object of type image), mosaiced video-frame sequences and time relationships (object of type video). On the other hand, the semantics (or meaning) of a multimedia object is a collective name for those features that pertain to the slice of the real world being *represented*, which exists independently of the existence of a object referring to it. Unlike form, the semantics of a multimedia object is thus *media independent*. Corresponding to these two dimensions, there are three categories of retrieval: one for each dimension (*form-based retrieval* and *semantic-based retrieval*) and one concerns the combination of both of them. Form-based retrieval methods automatically create an object representation by extracting features from multimedia data. These features will then be used in order to satisfy a query like "find images with a texture like this". On the other hand, semantic-based retrieval methods rely on a symbolic representation (typically, constructed manually perhaps with the assistance of some automatic tool) of the semantics of a multimedia object, that is, a description formulated in some suitable formal language. These descriptions will then be used in order to satisfy a query like "find images about girls". User queries may also address both dimensions, e.g. "find images about girls wearing clothes with a texture like this". In this query, texture addresses an image (form) feature, whereas the aboutness ("girls wearing clothes") addresses the semantics of an image.

Our DL based MIR model is a model in which multimedia objects' form properties and multimedia objects' semantic properties can be represented and all three kind of retrieval are allowed. Indeed, we will rely on a four-valued fuzzy Description Logics for representing the semantics of multimedia objects. Form properties, i.e. the low-level features of multimedia objects, are not represented explicitly within the DL, but are addressed by means of procedural attachments over a concrete domain. The retrieval process is then defined in terms of logical entailment. Roughly, a KB $\mathcal{K}$ contains the representations, in terms of assertions, of the semantic content of the multimedia objects (images, audio streams, video frame sequences). A query is a concept C describing the set of objects to be retrieved, both in terms of the objects' semantic properties and in terms of the objects' form properties. The retrieval of a multimedia object identified by a is then determined by checking whether $\mathcal{K} \models C(a)$.

There has already been some work about DLs and MIR, see e.g. [24–26,31]. Unfortunately, none of these address all the above-mentioned three categories of retrieval in a satisfactory way. [25,26,31] do not allow form-based retrieval, but deal with semantic-based retrieval only; [24] addresses all categories of retrieval, but the framework is hardwired on a particular image representation formalism.

The rest of the paper is organised as follows. The next section sets up a simple framework in which both the form and the semantic properties of multimedia objects are represented. The model is parametric w.r.t. the logic used. Section 3 formally specifies our DL. Section 4 addresses the issue of automatic reasoning within the DL, while Section 5 concludes.

2 Multimedia Databases

To represent the various aspects of a multimedia object, our MIR model consists of two layers, compliant to [2,12]: the *object form layer* and the *object semantics layer*. In the following see Figure 1 as an example. The object form layer consists of *Multimedia Objects* (MOs). These are the objects of interest for retrieval. Roughly, MOs represent "regions" (such as image regions, text parts and video frame sequences) of raw multimedia data. A MO

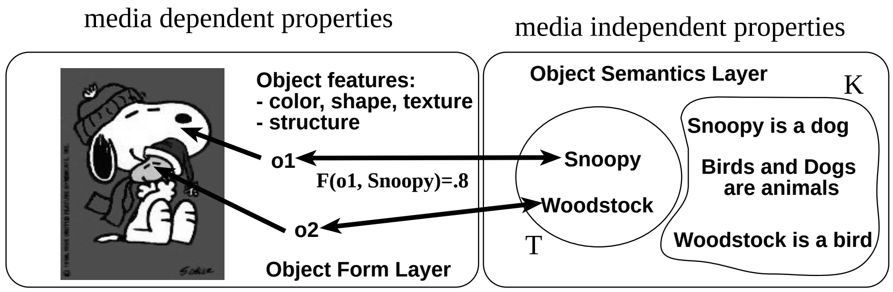

Fig. 1. MIR model layers and objects

may contain several feature attributes: each of them may be measured from the region the MO refers to. Of course, features are media dependent and, thus, for each media type there is a specific set of feature types. A feature is mainly characterised by a *feature extraction function* and by a *feature similarity function*. The feature extraction function extracts useful intrinsic properties of a region, like colour distributions, shapes, textures (media type image), mosaiced video-frame sequences (media type video) and text index term weights (media type text). A feature similarity function measures the similarity between two feature values of the same type. Typically, multimedia database systems already provide similarity functions for text, image, audio and video.

For the sake of our purpose, we model the object form layer as follows. Let $\mathcal{O}$ be a set of multimedia objects, which are assumed to be of the form $\langle o, v\rangle$, where o is an *object identifier* and v is a *value* having a certain type. If $\langle o, v\rangle$ and $\langle o, v'\rangle$ are two objects with the same identifier o then $v = v'$. We will not further specify the type of v. typically, v may be a string, an integer, an attribute tuple storing the features of the object (see, e.g. [1]). Further, let $\mathcal{M}$ be a set of similarity functions $s\colon \mathcal{O} \times \mathcal{O} \to [0, 1]$, determining the similarity between two objects according to some criteria. Typically, $s(o, o')$ depends on the features of o and o' (e.g., an image similarity function is obtained by combining appropriately the similarity functions for colour, texture and shape). We model an *Object Form Layer* (OFL) as a pair $OFL = \langle \mathcal{O}, \mathcal{M}\rangle$, where $\mathcal{O}$ is a set of multimedia objects and $\mathcal{M}$ is a set of similarity functions.

The object semantics layer describes the semantic properties of the slice of world the MOs are about. For instance, suppose that the *aboutness* of a certain MO, o, is Snoopy. The object semantics layer may describe the knowledge "Snoopy is a dog" and "dogs are animals". These descriptions can then be used in order to enhance the retrieval capabilities, i.e. we can infer that o is about a dog.

The semantic entities (or also events) which MOs can be about will be the logical individuals of our DL. We will call these individuals also *index terms*: as in information retrieval a text document may by associated to a keyword [29], we will associate a multimedia object to an individual. So, let

$\mathcal{T}$ be a set of index terms, i.e. individuals, and let $\mathcal{K}$ be a knowledge base, describing the properties of the individuals in $\mathcal{T}$ and the properties of the application domain. We model an *Object Semantics Layer* (OSL) as a pair $OSL = \langle \mathcal{T}, \mathcal{K} \rangle$, where $\mathcal{T}$ is a set of index terms (individuals) and $\mathcal{K}$ is a knowledge base.

According to the fuzzy information retrieval model [8,19,27,21], let F be a membership function describing the correlation between multimedia objects and index terms. The function F may be defined as $F: \mathcal{O} \times \mathcal{T} \to [0,1]$ in which $\mathcal{O}$ is a set of multimedia objects and $\mathcal{T}$ is the set of all index terms (individuals). The value $F(o,a)$ indicates to which degree the multimedia object o deals with the individual a; this value is also called index term weight. The meaning of an index term a may in this context be represented as a fuzzy subset of multimedia objects in $\mathcal{O}$, $m(a)$, with the quantitative measure of aboutness being the values of function F for a given index term a: $m(a) = \{\langle o, \mu_a(o) \rangle : o \in \mathcal{O}\}$, in which $\mu_a(o) = F(o,a)$. $m(a)$ is the meaning of individual a. The function F acts as the membership function of $m(a)$. Finally, a *multimedia object database about form and semantics*, or simply *Multimedia Database* (DB) is a tuple

$$DB = \langle OFL, OSL, F \rangle \tag{1}$$

where $OFL = \langle \mathcal{O}, \mathcal{M} \rangle$ is a object form layer, $OSL = \langle \mathcal{T}, \mathcal{K} \rangle$ is a object semantics layer and $F: \mathcal{O} \times \mathcal{T} \to [0,1]$ is a fuzzy membership function. This completes this part. In the following sections we will describe the DL by means of which we may describe the OSL.

3 DLs and the Retrieval of Multimedia Objects

Our DL combines and extents the works [25,26,32,34]. In [34] a fuzzy extension of a quite general DL, $\mathcal{ALC}$ [30], in the context of two-valued semantics has been described; in [32] a fuzzy extension of propositional logic within a four-valued semantics has been described and [25] briefly formalises a four-valued fuzzy DL. The rational behind the combination of fuzzy semantics and four-valued semantics is twofold. In [26] a four-valued semantics [3,6,23,28] has been proposed for DLs in order to enforce a notion of logical entailment in which premises (i.e. the knowledge base) need be relevant to conclusions (i.e. the query) to a stronger extent than classical "material" logical implication does. The logic specified in [26] is still insufficient for describing retrieval situations. Retrieval is usually not only a yes-no question, as classical logical entailment is, but rather the answer should be graded. This is similar to switching from the boolean $F(o,a) \in \{0,1\}$ case to the fuzzy case $F(o,a) \in [0,1]$. Because of this, [25] proposes a logic in which, rather than deciding *tout court* whether a multimedia object satisfies a query or not, we are able to *rank* the retrieved objects according to how strongly the system believes in their relevance to a query. The logic we will propose here, starts from [25], but substantial modifications will be made from a semantics point of view in order to deal with all three categories of retrieval mentioned previously. In the following, we will proceed step by step until our final logic is specified.

3.1 A Classical DL

The specific basic DL we employ is the logic $\mathcal{ALC}$ [30] ($\mathcal{A}$ttributive $\mathcal{L}$anguage with $\mathcal{C}$omplements), a significant representative of DLs $\mathcal{ALC}$ is universally considered the "standard" DL (as much as **K** is considered the "standard" modal logic). Reverting to one's DL of choice may be taken as the very last step within an application domain.

Consider a new alphabet of symbols called *individuals* (denoted by a and b). Let $OFL = \langle \mathcal{O}, \mathcal{M} \rangle$ be an object form layer such that all similarity functions $s \in \mathcal{M}$ are boolean, i.e. $s: \mathcal{O} \times \mathcal{O} \to \{0,1\}$ and let $F: \mathcal{O} \times \mathcal{T} \to \{0,1\}$ be a boolean membership function, where $\mathcal{T}$ is a set of individuals. We now specify the DL $\mathcal{ALC}(OFL, F)$ which "uses" both OFL and F based on classical two-valued semantics. We will rely on the notion of concrete domain [5], but rather than to rely on the quite general case, as presented in [5], we will restrict the concrete domain to our special case. A generalisation is straightforward, but complicates our presentation unnecessarily.

To begin, we need *concrete individual* symbols representing the elements $\langle o, v \rangle \in \mathcal{O}$. For ease of notation, we will use o as concrete individual in $\mathcal{ALC}(OFL, F)$, denoting $\langle o, v \rangle$. Furthermore, for each $s \in \mathcal{M}$ we assume that s is a two-place *concrete role* symbol in $\mathcal{ALC}(OFL, F)$, $\mathsf{IsAbout}$ is a two-place concrete role symbol denoting F, whereas O is an one-place concrete concept symbol in $\mathcal{ALC}(OFL, F)$ denoting $\mathcal{O}$.

Let us assume two new alphabets of symbols, called *primitive concepts* (denoted by A) and *primitive roles* (denoted by P). A *role* is either a primitive role or a concrete role. The *concepts* (denoted by C and D) of the language $\mathcal{ALC}(OFL, F)$ are formed out of primitive concepts according to the following syntax rules:

$$
\begin{array}{rrl}
C, D \longrightarrow & A \mid & \text{(primitive concept)} \\
& \mathsf{O} \mid & \text{(class of multimedia objects)} \\
& C \sqcap D \mid & \text{(concept conjunction)} \\
& C \sqcup D \mid & \text{(concept disjunction)} \\
& \neg C \mid & \text{(concept negation)} \\
& \forall R.C \mid & \text{(universal quantification)} \\
& \exists R.C & \text{(existential quantification)} \quad .
\end{array}
\tag{2}
$$

An *interpretation* $\mathcal{I}$ for $\mathcal{ALC}(OFL, F)$ consists of a non empty set $\Delta^{\mathcal{I}} = {\Delta_A}^{\mathcal{I}} \cup \mathcal{O}$ (called the *domain*), where ${\Delta_A}^{\mathcal{I}}$ is the abstract domain and $\mathcal{O}$ is the concrete domain such that ${\Delta_A}^{\mathcal{I}} \cap \mathcal{O} = \emptyset$. The *interpretation function* $\cdot^{\mathcal{I}}$ is such that

1. $\cdot^{\mathcal{I}}$ maps every individual into ${\Delta_A}^{\mathcal{I}}$;
2. $a^{\mathcal{I}} \neq b^{\mathcal{I}}$, if $a \neq b$. (*Unique Name Assumption*);
3. $o^{\mathcal{I}} = \langle o, v \rangle \in \mathcal{O}$, for all concrete individuals o;
4. $\cdot^{\mathcal{I}}$ maps every primitive concept into a function from ${\Delta_A}^{\mathcal{I}}$ to $\{t, f\}$;
5. $\cdot^{\mathcal{I}}$ maps O into a function from $\mathcal{O}$ to $\{t, f\}$ such that $\mathsf{O}^{\mathcal{I}}(d) = t$ if $d \in \mathcal{O}$;
6. $\cdot^{\mathcal{I}}$ maps every primitive role into a function from ${\Delta_A}^{\mathcal{I}} \times {\Delta_A}^{\mathcal{I}}$ to $\{t, f\}$;
7. $\cdot^{\mathcal{I}}$ maps $\mathsf{IsAbout}$ into a function from $\mathcal{O} \times {\Delta_A}^{\mathcal{I}}$ to $\{t, f\}$ such that $\mathsf{IsAbout}^{\mathcal{I}}(d, d') = t$ if $\exists a \in \mathcal{T}.a^{\mathcal{I}} = d' \wedge F(d, a) = 1$;
8. $\cdot^{\mathcal{I}}$ maps $s \in \mathcal{M}$ into a function from $\mathcal{O} \times \mathcal{O}$ to $\{t, f\}$ such that $s^{\mathcal{I}}(d, d') = t$ if $s(d, d') = 1$.

An interpretation is extended to complex concepts by appropriately combining the interpretations of their components:

$$
\begin{aligned}
(C \sqcap D)^{\mathcal{I}}(d) &= t \text{ iff } C^{\mathcal{I}}(d) = t \text{ and } D^{\mathcal{I}}(d) = t \\
(C \sqcup D)^{\mathcal{I}}(d) &= t \text{ iff } C^{\mathcal{I}}(d) = t \text{ or } D^{\mathcal{I}}(d) = t \\
(\neg C)^{\mathcal{I}}(d) &= t \text{ iff } C^{\mathcal{I}}(d) = f \\
(\forall R.C)^{\mathcal{I}}(d) &= t \text{ iff for all } d' \in \Delta^{\mathcal{I}}, \text{ if } R^{\mathcal{I}}(d, d') = t \text{ then } C^{\mathcal{I}}(d') = t \\
(\exists R.C)^{\mathcal{I}}(d) &= t \text{ iff for some } d' \in \Delta^{\mathcal{I}}, R^{\mathcal{I}}(d, d') = t \text{ and } C^{\mathcal{I}}(d') = t \ .
\end{aligned}
$$

Note that each concept C and role R can be mapped into an equivalent open first-order formula $F_C(x)$ and $F_R(x, y)$, respectively:

$$F_A(x) = A(x) \tag{3a}$$
$$F_R(x, y) = R(x, y) \tag{3b}$$
$$F_{C \sqcap D}(x) = F_C(x) \wedge F_D(x) \tag{3c}$$
$$F_{C \sqcup D}(x) = F_C(x) \vee F_D(x) \tag{3d}$$
$$F_{\neg C}(x) = \neg F_C(x) \tag{3e}$$
$$F_{\forall R.C}(x) = \forall y. F_R(x, y) \rightarrow F_C(y) \tag{3f}$$
$$F_{\exists R.C}(x) = \exists y. F_R(x, y) \wedge F_C(y) \ . \tag{3g}$$

An *assertion* (denoted by $\mathcal{A}$) is an expression of type $C(a)$ (meaning that a is an instance of concept C), or an expression of type $R(a, b)$ (meaning that a is related to b by means of role R). An interpretation $\mathcal{I}$ *satisfies* $C(a)$ (resp. $R(a, b)$) iff $C^{\mathcal{I}}(a^{\mathcal{I}}) = t$ (resp. $R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) = t$).

A *specialisation* (denoted by $\mathcal{T}$) is an expression of the form $C \Rightarrow D$, where C, D are concepts. Specialisations allow to state the existence of a specialisation ("more specific than") relation between concepts. For instance, the specialisation $\mathsf{Father} \Rightarrow \mathsf{Man} \sqcap \exists \mathsf{HasChild}.\mathsf{Child}$ states that a father is a man having at least a child. An interpretation $\mathcal{I}$ *satisfies* $C \Rightarrow D$ is iff $C^{\mathcal{I}}(d) = t$ implies that $D^{\mathcal{I}}(d) = t$, for all $d \in \Delta^{\mathcal{I}}$. As a consequence, a specialisation may be seen as the first-order formula $F_{C \Rightarrow D}$

$$F_{C \Rightarrow D} = \forall x. F_C(x) \rightarrow F_D(x) \ . \quad (4)$$

A set $\mathcal{K}$ of assertions and specialisations will be called a *Knowledge Base* (KB). With $\mathcal{K}_{\mathcal{A}}$ we will denote the set of assertions in $\mathcal{K}$, whereas with $\mathcal{K}_{\mathcal{T}}$ we will denote the set of specialisations in $\mathcal{K}$.

Often, DLs support only a special form of specialisations. The form of specialisations they allow is defined as follows. A *concept definition* is an expression of the form $A{:=}\, C$, where A is a primitive concept, called *concept name*, and C is a concept. The goal of a concept definition $A{:=}\, C$ is to define concept A as being "equivalent" to concept C, i.e. A is the *name* of concept C. Essentially, a concept definition $A{:=}\, C$ is a macro which can be expressed in terms of specialisations: each occurrence of $A{:=}\, C$ can be replaced by considering both specialisations $A \Rightarrow C$ and $C \Rightarrow A$.

Consider a KB $\mathcal{K}$ such that $\mathcal{K}_{\mathcal{T}} \neq \emptyset$. Suppose $\mathcal{K}_{\mathcal{T}}$ contains only concept definitions $A{:=}\, C$ and specialisations of the form $A \Rightarrow C$, where A is a primitive concept. We will say that A *directly uses* primitive concept B, if either (i) there is a concept definition $A{:=}\, C \in \mathcal{K}_{\mathcal{T}}$ such that B occurs in C; or (ii) there is a specialisation $A \Rightarrow C \in \mathcal{K}_{\mathcal{T}}$ such that B occurs in C. Let *uses* be the transitive closure of the relation directly uses in $\mathcal{K}$. $\mathcal{K}_{\mathcal{T}}$ is *cyclic* iff there is A such that A uses A through $\mathcal{K}_{\mathcal{T}}$. In the rest of the paper we will restrict our attention to well formed KBs: we will say that a KB $\mathcal{K}$ is *well formed* iff if $\mathcal{K}_{\mathcal{T}} \neq \emptyset$ then

1. $\mathcal{K}_{\mathcal{T}}$ contains only concept definitions $A{:=}\, C$ and specialisations of the form $A \Rightarrow C$, where A is a primitive concept;
2. no A appears more than once on the left hand side of concept definitions and specialisations in $\mathcal{K}_{\mathcal{T}}$; and
3. $\mathcal{K}_{\mathcal{T}}$ is not cyclic.

An interpretation $\mathcal{I}$ *satisfies* (*is a model of*) $\mathcal{K}$ iff $\mathcal{I}$ satisfies each element in $\mathcal{K}$. $\mathcal{K}$ *entails* an assertion $\mathcal{A}$ (denoted by $\mathcal{K} \models_2 \mathcal{A}$) iff every model of $\mathcal{K}$ satisfies $\mathcal{A}$. Notice that $\mathcal{K} \models_2 R(a,b)$ iff $R(a,b) \in \mathcal{K}$. $\mathcal{K}$ *entails* a specialisation $\mathcal{T}$ (denoted by $\mathcal{K} \models_2 \mathcal{T}$) iff every model of $\mathcal{K}$ satisfies $\mathcal{T}$. The notion of entailment is easily extended to DBs: a multimedia database $DB = \langle OFL, OSL, F \rangle$, where $OSL = \langle \mathcal{T}, \mathcal{K} \rangle$, *entails* an assertion $C(a)$, written $DB \models_2 C(a)$ iff each interpretation for $\mathcal{ALC}(OFL, F)$ satisfying $\mathcal{K}$ satisfies $C(a)$ too.

The problem of determining whether $\mathcal{K} \models_2 \mathcal{T}$ is called *hybrid subsumption problem*; the problem of determining whether $\mathcal{K}$ is satisfiable is called *knowledge base satisfiability problem*, and the problem of determining whether $\mathcal{K} \models_2 \mathcal{A}$ is called *instance checking problem*. It can easily be verified that the following relations hold:

$$\mathcal{K} \models_2 C \Rightarrow D \ \text{ iff } \ \mathcal{K} \cup \{C(a), \neg D(a)\} \text{ is not satisfiable} \quad (5a)$$

$$\mathcal{K} \models_2 C(a) \ \text{ iff } \ \mathcal{K} \cup \{\neg C(a)\} \text{ is not satisfiable} \ , \quad (5b)$$

where a is a new individual. As a consequence, all the above problems can be reduced to the knowledge base satisfiability problem. There exists a well known technique, based on constraint propagation, which solves this problem. The interested reader can consult e.g., [5,10].

Example 1. Consider the KB $\mathcal{K}$ containing the following taxonomy.

$$\begin{array}{lcl}
\mathsf{IndividualSport} & \Rightarrow & \mathsf{Sport} \\
\mathsf{TeamSport} & \Rightarrow & \mathsf{Sport} \\
\mathsf{Basketball} & \Rightarrow & \mathsf{SportTool} \\
\mathsf{TennisRacket} & \Rightarrow & \mathsf{SportTool} \\
\mathsf{TeamSport} & := & \mathsf{Sport} \sqcap \\
 & & (\exists \mathsf{KindOfSport}.\mathsf{Sport}) \sqcap \\
 & & (\forall \mathsf{KindOfSport}.\mathsf{TeamSport}) \\
\mathsf{IndividualSport} & := & \mathsf{Sport} \sqcap \\
 & & (\exists \mathsf{KindOfSport}.\mathsf{Sport}) \sqcap \\
 & & (\forall \mathsf{KindOfSport}.\mathsf{IndividualSport}) \\
\mathsf{Basket} & := & \mathsf{TeamSport} \sqcap \\
 & & (\exists \mathsf{HasSportTool}.\mathsf{SportTool}) \sqcap \\
 & & (\forall \mathsf{HasSportTool}.\mathsf{Basketball}) \\
\mathsf{Tennis} & := & \mathsf{IndividualSport} \sqcap \\
 & & (\exists \mathsf{HasSportTool}.\mathsf{SportTool}) \sqcap \\
 & & (\forall \mathsf{HasSportTool}.\mathsf{TennisRacket})
\end{array}$$

Furthermore, assume that $\mathcal{K}$ contains $\mathsf{Basket(b)}$ and $\mathsf{Tennis(t)}$. Let $DB = \langle OFL, OSL, F \rangle$ be a DB where $OFL = \langle \mathcal{O}, \mathcal{M} \rangle$ is an object form layer containing two images $\mathsf{i1} = \langle \mathsf{o1}, \mathsf{v1} \rangle$ and $\mathsf{i2} = \langle \mathsf{o2}, \mathsf{v2} \rangle$, $OSL = \langle \mathcal{T}, \mathcal{K} \rangle$ is the object semantics layer such that the set of index terms is $\mathcal{T} = \{\mathsf{b}, \mathsf{t}\}$ and $F\colon \mathcal{O} \times \mathcal{T} \rightarrow \{0, 1\}$ is a membership function such that $F(\mathsf{i1}, \mathsf{b}) = 1$ and $F(\mathsf{i2}, \mathsf{t}) = 1$. That is, F specifies that the aboutness of images $\mathsf{i1}$ and $\mathsf{i2}$ is basket and tennis, respectively.

It is quite easy to verify that $\mathcal{K} \models_2 \mathsf{Basket} \Rightarrow \mathsf{Sport}$ stating that a Basket is a Sport. Similarly, $\mathcal{K} \models_2 \mathsf{Tennis} \Rightarrow \mathsf{Sport}$ holds.

Semantic-based retrieval: an example of semantic-based retrieval is the following: "find images about sport" We may query DB by means of the query concept

$$\mathsf{Q} = \mathsf{O} \sqcap \exists \mathsf{IsAbout}.\mathsf{Sport} \; .$$

The answer will be the list containing both $\mathsf{i1}$ and $\mathsf{i2}$, as $DB \models_2 \mathsf{Q(o1)}$ and $DB \models_2 \mathsf{Q(o2)}$ hold. Another example of semantic-based retrieval is: "find images about individual sports". The query concept can be

$$\mathsf{Q}' = \mathsf{O} \sqcap \exists \mathsf{IsAbout}.(\mathsf{Sport} \sqcap \exists \mathsf{KindOfSport}.\mathsf{IndividualSport}) \; .$$

It follows that only image $\mathsf{i2}$ will be retrieved. In fact, $DB \not\models_2 \mathsf{Q}'(\mathsf{o1})$ and $DB \models_2 \mathsf{Q}'(\mathsf{o2})$ hold.

Form-based retrieval: a typical case of form-based retrieval is: "find images which are similar to a given image $\mathsf{i3} = \langle \mathsf{o3}, \mathsf{v3} \rangle$". Here, we are looking for images i which at the form level are similar to $\mathsf{i3}$, i.e. the medium dependent properties (features) of i match those of $\mathsf{i3}$. Before submitting our query to the DB, we assume that $\mathsf{i3}$ is in $\mathcal{O}$ (if not so, put $\mathsf{i3}$ into $\mathcal{O}$) and we assume also that there is a similarity function $\mathsf{Sim}_{\mathsf{i3}}: \mathcal{O} \times \{\mathsf{i3}\} \to \{0,1\}$ in $\mathcal{M}$ establishing that image $\mathsf{i1}$ is similar to image $\mathsf{i3}$, i.e. $\mathsf{Sim}_{\mathsf{i3}}(\mathsf{i1}, \mathsf{i3}) = 1$. We can now formalise our request by means of the concept

$$\mathsf{Q}'' = \mathsf{O} \sqcap (\exists \mathsf{Sim}_{\mathsf{i3}}.\mathsf{O}) \ .$$

It follows easily that only image $\mathsf{i1}$ will be retrieved. In fact, $DB \models_2 \mathsf{Q}''(\mathsf{o1})$ and $DB \not\models_2 \mathsf{Q}''(\mathsf{o2})$ hold.

Combination of form-based and semantic-based retrieval: let us further expand the last example, illustrating a typical combination of form-based retrieval and semantic-based retrieval. Suppose our information request is "find images which are similar to a given image $\mathsf{i3} = \langle \mathsf{o3}, \mathsf{v3} \rangle$ and which are about sport". We can formalise our request by means of the concept

$$\mathsf{Q}''' = \mathsf{Q}'' \sqcap \mathsf{Q} \ .$$

As before, only image $\mathsf{i1}$ will be retrieved, as $DB \models_2 \mathsf{Q}'''(\mathsf{o1})$ and $DB \not\models_2 \mathsf{Q}'''(\mathsf{o2})$ hold.

■

3.2 A Four-Valued DL

The four-valued semantics for $\mathcal{ALC}$ is described in [25,26,33]. The extension to $\mathcal{ALC}(OFL, F)$ is straightforward which we briefly resume below.

Four-valued Semantics

The key difference between a classical logic and our four-valued logic is that, while the former relies on the classical set of truth values $\{t, f\}$, the latter relies on its *powerset* $2^{\{t,f\}}$, i.e. the four values are $\{t\}$, $\{f\}$, $\{t, f\}$ and $\emptyset$. These values may be understood as representing the status of a sentence in the epistemic state of a reasoning agent. Under this view, if the value of a sentence contains t, then the agent has evidence to the effect – or beliefs – that the sentence is true. Similarly, if it contains f, then the agent has evidence to the effect that the sentence is false. The value $\emptyset$ corresponds to a lack of evidence, while the truth value $\{t, f\}$ corresponds to the possession of contradictory evidence.

A four-valued *interpretation* for $\mathcal{ALC}(OFL, F)$, $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, in the following called interpretation, is as before, except that the *interpretation function* of $\mathcal{I}$) is now such that

1. $\cdot^{\mathcal{I}}$ maps every primitive concept into a function from $\Delta_A{}^{\mathcal{I}}$ to $2^{\{t,f\}}$;
2. $\cdot^{\mathcal{I}}$ maps O into a function from $\mathcal{O}$ to $2^{\{t,f\}}$ such that $t \in \mathsf{O}^{\mathcal{I}}(d)$ if $d \in \mathcal{O}$;
3. $\cdot^{\mathcal{I}}$ maps every primitive role into a function from $\Delta_A{}^{\mathcal{I}} \times \Delta_A{}^{\mathcal{I}}$ to $2^{\{t,f\}}$;
4. $\cdot^{\mathcal{I}}$ maps $\mathsf{IsAbout}$ into a function from $\mathcal{O} \times \Delta_A{}^{\mathcal{I}}$ to $2^{\{t,f\}}$ such that $t \in \mathsf{IsAbout}^{\mathcal{I}}(d, d')$ if $\exists a \in \mathcal{T}.a^{\mathcal{I}} = d' \wedge F(d, a) = 1$;
5. $\cdot^{\mathcal{I}}$ maps $s \in \mathcal{M}$ into a function from $\mathcal{O} \times \mathcal{O}$ to $2^{\{t,f\}}$ such that $t \in s^{\mathcal{I}}(d, d')$ if $s(d, d') = 1$.

The interpretation function is extended to complex concepts as follows:

$$
\begin{array}{ll}
t \in (C \sqcap D)^{\mathcal{I}}(d) & \text{iff } t \in C^{\mathcal{I}}(d) \text{ and } t \in D^{\mathcal{I}}(d) \\
f \in (C \sqcap D)^{\mathcal{I}}(d) & \text{iff } f \in C^{\mathcal{I}}(d) \text{ or } f \in D^{\mathcal{I}}(d) \\
\\
t \in (C \sqcup D)^{\mathcal{I}}(d) & \text{iff } t \in C^{\mathcal{I}}(d) \text{ or } t \in D^{\mathcal{I}}(d) \\
f \in (C \sqcup D)^{\mathcal{I}}(d) & \text{iff } f \in C^{\mathcal{I}}(d) \text{ and } f \in D^{\mathcal{I}}(d) \\
\\
t \in (\neg C)^{\mathcal{I}}(d) & \text{iff } f \in C^{\mathcal{I}}(d) \\
f \in (\neg C)^{\mathcal{I}}(d) & \text{iff } t \in C^{\mathcal{I}}(d) \\
\\
t \in (\forall R.C)^{\mathcal{I}}(d) & \text{iff } \forall\, d' \in \Delta^{\mathcal{I}}, t \in R^{\mathcal{I}}(d, d') \text{ implies } t \in C^{\mathcal{I}}(d') \\
f \in (\forall R.C)^{\mathcal{I}}(d) & \text{iff } \exists\, d' \in \Delta^{\mathcal{I}}, t \in R^{\mathcal{I}}(d, d') \text{ and } f \in C^{\mathcal{I}}(d') \\
\\
t \in (\exists R.C)^{\mathcal{I}}(d) & \text{iff } \exists\, d' \in \Delta^{\mathcal{I}}, t \in R^{\mathcal{I}}(d, d') \text{ and } t \in C^{\mathcal{I}}(d') \\
f \in (\exists R.C)^{\mathcal{I}}(d) & \text{iff } \forall\, d' \in \Delta^{\mathcal{I}}, t \in R^{\mathcal{I}}(d, d') \text{ implies } f \in C^{\mathcal{I}}(d') \;.
\end{array}
$$

Let $\mathcal{I}$ be an interpretation and let $\mathcal{K}$ be a KB. $\mathcal{I}$ *satisfies* an assertion $C(a)$, iff $t \in C^{\mathcal{I}}(a^{\mathcal{I}})$, whereas $\mathcal{I}$ *satisfies* $R(a, b)$ iff $t \in R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}})$. $\mathcal{I}$ *satisfies* a specialisation $C \Rightarrow D$ iff $t \in C^{\mathcal{I}}(d)$ implies $t \in D^{\mathcal{I}}(d)$, for all $d \in \Delta^{\mathcal{I}}$. From this, the definition of satisfiable knowledge base and entailment (denoted by $\models_4$) easily follows. As for the classical two-valued case, $\mathcal{K} \models_4 R(a, b)$ iff $R(a, b) \in \mathcal{K}$.

We will briefly resume some main properties of our four-valued semantics (see also [26,33] for details). Let $DB = \langle OFL, OSL, F\rangle$ be a multimedia database. At first, it can easily be verified that $\mathcal{K} \models_4 C \Rightarrow D$ iff $\mathcal{K} \cup \{C(a)\} \models_4 D(a)$ holds (a is a new individual), i.e., the hybrid subsumption problem can be reduced to the instance checking problem. Therefore, we will restrict our attention to the instance checking problem only. For it, there exists a decision algorithm, which is described in [33].

Satisfiability: Each KB is (four-valued) satisfiable. Consequently, unlike classical DLs, ($5a$) and ($5b$) do not apply. Note that, as described in [26], the satisfiability property is important in retrieval, as we cannot assume that the aboutness of a MO is described in a consistent manner.

Soundness of the semantics: Our logic is *sound* with respect to two-valued semantics. In fact, $DB \models_4 \mathcal{A}$ implies $DB \models_2 \mathcal{A}$. In particular, $\models_4 \subset \models_2$ holds. This means that if a MO is retrieved according to our four-valued semantics then it will be retrieved according to the classical semantics as well.

Paradoxes: The so-called paradoxes of classical logical implication do not hold in our four-valued semantics: e.g. $\{C(a), \neg C(a)\} \not\models_4 (\exists \mathsf{IsAbout}.D)(o)$ reflects a "relevance semantics" in the sense that both $C(a)$ and $\neg C(a)$ together do not give any argument in supporting that "the MO o is about D". Similarly, $\emptyset \not\models_4 (\forall \mathsf{IsAbout}.(C \sqcup \neg C))(o)$ states that ([26]) the absence of knowledge ($\mathcal{K} = \emptyset$) does not give any argument in supporting that "if the MO o is about something then o is about $C \sqcup \neg C$". Indeed, in our four-valued semantics, it might be the case of lack of evidence w.r.t. $C \sqcup \neg C$: for an interpretation $\mathcal{I}$, it might be the case that $t \in \mathsf{IsAbout}^{\mathcal{I}}(o^{\mathcal{I}}, d')$ but $C^{\mathcal{I}}(d') = (\neg C)^{\mathcal{I}}(d') = \emptyset$. That is, we know that the MO o is about something, but we do not know what o is about.

Example 2. Consider the multimedia database DB and the query concepts Q, Q', Q'' and Q''' in the previous Example 1. As for the two-valued case, it is quite easy to verify that $DB \models_4 \mathsf{Basket} \Rightarrow \mathsf{Sports}$ and $DB \models_4 \mathsf{Tennis} \Rightarrow \mathsf{Sports}$ hold. Moreover, it can be verified that $DB \models_4 \mathsf{Q}(\mathsf{o1})$, $DB \models_4 \mathsf{Q}(\mathsf{o2})$, $DB \not\models_4 \mathsf{Q}'(\mathsf{o1})$, $DB \models_4 \mathsf{Q}'(\mathsf{o2})$, $DB \models_4 \mathsf{Q}''(\mathsf{o1})$, $DB \not\models_4 \mathsf{Q}''(\mathsf{o2})$, $DB \models_4 \mathsf{Q}'''(\mathsf{o1})$ and $DB \not\models_4 \mathsf{Q}'''(\mathsf{o2})$ hold. Note also that e.g. $DB \not\models_4 (\exists \mathsf{IsAbout}.(\mathsf{Car} \cup \neg \mathsf{Car}))(\mathsf{o1})$, is a consequence of the fact that we do know nothing about whether the image $\mathsf{i1}$ is about Car or is about $\neg\mathsf{Car}$. On the other hand, according to classical semantics, $DB \models_2 (\exists \mathsf{IsAbout}.(\mathsf{Car} \cup \neg \mathsf{Car}))(\mathsf{o1})$ holds and, thus, reduces the "precision degree" of retrieval. ■

3.3 A Four-Valued Fuzzy DL

Until now, we assumed that each similarity function $s \in \mathcal{M}$ of an object form layer $OFL = \langle \mathcal{O}, \mathcal{M} \rangle$ and the membership function F are boolean, i.e. the values are in $\{0, 1\}$. Our last step is to generalise the logic $\mathcal{ALC}(OFL, F)$ to the case where these values can be in $[0, 1]$.

Our fuzzy extension takes inspiration from Zadeh's work on fuzzy sets [38]. A *fuzzy set* A with respect to a set X is characterised by a *membership function* $|A| : X \to [0, 1]$, assigning an A-membership degree, $|A|(x)$, to each element x in X. This degree gives us an estimation of the belonging of x to A. Typically, if $|A|(x) = 1$ then x definitely belongs to A, while $|A|(x) = .8$ means that x is an element of A, but to a lesser extent. Moreover, according to Zadeh, the membership function has to satisfy three well-known restrictions, for all $x \in X$ and for all fuzzy sets A, B with respect to X:

$$
\begin{aligned}
|A \cap B|(x) &= \min\{|A|(x), |B|(x)\} \\
|A \cup B|(x) &= \max\{|A|(x), |B|(x)\} \\
|\overline{A}|(x) &= 1 - |A|(x) \ ,
\end{aligned}
$$

where $\overline{A}$ is the complement of A in X. Conjunction and disjunction of fuzzy sets can be defined by any t-norm and t-conorm operators, respectively; usually min and max are adopted (see e.g. [37]).

When we switch to logic, and to DLs in particular, we have concepts C which are interpreted as fuzzy sets and, thus, speak about C-membership degrees, i.e., $|C|$ is the membership function of C. For instance, the assertion that individual a is an instance of concept C may have as a membership degree any real number in between 0 and 1: if the degree of membership of $\mathsf{C(a)}$ is 1, then a is definitely an instance of C, while if the degree of membership of $\mathsf{C(a)}$ is .8 then a is an instance of C, but to a lesser extent. Hence, in a fuzzy DL, concepts become *imprecise*. As a consequence, given e.g. a query concept Q, the retrieval process produces a ranking of concrete individuals: the rank of o, for each concrete individual o, is the degree of membership of $\mathsf{Q(o)}$, and will be interpreted as the degree of aboutness of the multimedia object identified by o to the concept Q.

In the following, let $OFL = \langle \mathcal{O}, \mathcal{M} \rangle$ be a generic object form layer and let F be a membership function.

Fuzzy assertions

A *fuzzy assertion* (denoted by α) is an expression of type $\langle \mathcal{A} \geq n \rangle$, where $\mathcal{A}$ is an assertion in $\mathcal{ALC}(OFL, F)$ and $n \in (0, 1]$.

In fuzzy $\mathcal{ALC}(OFL, F)$, a concept is interpreted as a fuzzy set. According to this view, the intended meaning of $\langle C(a) \geq n \rangle$ we will adopt is: "the membership degree of individual a being an instance of concept C is at least n". Similarly for roles: the intended meaning of $\langle R(a, b) \geq n \rangle$ we will adopt is: "the membership degree of individual a being related to b by means of role R is at least n".

In the two-valued case (see [34]) w.r.t. $\mathcal{ALC}$, a *fuzzy valuation* is a function $|\cdot|$ mapping (i) $\mathcal{ALC}$ concepts into a membership function $\Delta \to [0, 1]$ (Δ is the domain); and (ii) $\mathcal{ALC}$ roles into a membership function $\Delta \times \Delta \to [0, 1]$. If C is a concept then $|C|$ will naturally be interpreted as the *membership function* of the fuzzy concept (set) C, i.e. if $d \in \Delta$ is an object of the domain Δ then $|C|(d)$ gives us the degree of being the object d an element of the fuzzy concept C. Similar arguments holds for roles.

In the four-valued case, consistently with our approach of distinguishing explicit from implicit falsehood (e.g. distinguishing $f \in C^{\mathcal{I}}(d)$ from $t \notin C^{\mathcal{I}}(d)$), we will use rather two fuzzy valuations, $|\cdot|^t$ and $|\cdot|^f$: $|C|^t$ will naturally be interpreted as the *membership function* of C, whereas $|C|^f$ will analogously be interpreted as the *non-membership function* of C. For instance, $|\mathsf{Tall}|^t(d)$ gives us the degree of being d Tall, whereas $|\mathsf{Tall}|^f(d)$ gives us the degree of being d not Tall. While in the classical "two-valued" fuzzy case, as usual, $|C|^f = 1 - |C|^t$, for each concept C, we might well have $|C|^t(d) = .6$ and $|C|^f(d) = .8$. This is a natural consequence of our four-valued approach, where $f \in C^{\mathcal{I}}(d)$ and $t \in C^{\mathcal{I}}(d)$ is allowed. The case of roles is similar.

A *fuzzy interpretation*[2] for fuzzy $\mathcal{ALC}(OFL, F)$ is a tuple $\mathcal{I} = ((\cdot)^{\mathcal{I}}, |\cdot|^t, |\cdot|^f, \Delta^{\mathcal{I}})$, where $\cdot^{\mathcal{I}}$ maps individuals and concrete individuals as usual and

1. $|\cdot|^t$ and $|\cdot|^f$ are fuzzy valuations, i.e. $|\cdot|^t$ and $|\cdot|^f$
   (a) map every primitive concept into a function from ${\Delta_A}^{\mathcal{I}}$ to $[0,1]$;
   (b) map O into a function from $\mathcal{O}$ to $[0,1]$ such that $|\mathsf{O}|^t(d) = 1$ if $d \in \mathcal{O}$;
   (c) map every primitive role into a function from ${\Delta_A}^{\mathcal{I}} \times {\Delta_A}^{\mathcal{I}}$ to $[0,1]$;
   (d) map $\mathsf{IsAbout}$ into a membership function from $\mathcal{O} \times {\Delta_A}^{\mathcal{I}}$ to $[0,1]$ such that $|\mathsf{IsAbout}|^t(d,d') = n$ if $\exists a \in \mathcal{T}.a^{\mathcal{I}} = d' \wedge F(d,a) = n$;
   (e) map $s \in \mathcal{M}$ into a membership function from $\mathcal{O} \times \mathcal{O}$ to $[0,1]$ such that $|s|^t(d,d') = n$ if $s(d,d') = n$;
2. $\cdot^{\mathcal{I}}$ maps every fuzzy assertion into an element of $2^{\{t,f\}}$.

Moreover, $|\cdot|^t$ and $|\cdot|^f$ are extended to complex concepts as follows: for all $d \in \Delta^{\mathcal{I}}$

$$\begin{aligned}
|C \sqcap D|^t(d) &= \min\{|C|^t(d), |D|^t(d)\} \\
|C \sqcap D|^f(d) &= \max\{|C|^t(d), |D|^t(d)\} \\[1ex]
|C \sqcup D|^t(d) &= \max\{|C|^t(d), |D|^t(d)\} \\
|C \sqcup D|^f(d) &= \min\{|C|^t(d), |D|^t(d)\} \\[1ex]
|\neg C|^t(d) &= |C|^f(d) \\
|\neg C|^f(d) &= |C|^t(d) \\[1ex]
|\forall R.C|^t(d) &= \min_{d' \in \Delta^{\mathcal{I}}}\{\max\{1 - |R|^t(d,d'), |C|^t(d')\}\} \\
|\forall R.C|^f(d) &= \max_{d' \in \Delta^{\mathcal{I}}}\{\min\{|R|^t(d,d'), |C|^f(d')\}\}
\end{aligned}$$

$$\begin{aligned}
|\exists R.C|^t(d) &= \max_{d' \in \Delta^{\mathcal{I}}}\{\min\{|R|^t(d,d'), |C|^t(d')\}\} \\
|\exists R.C|^f(d) &= \min_{d' \in \Delta^{\mathcal{I}}}\{\max\{1 - |R|^t(d,d'), |C|^f(d')\}\} \ .
\end{aligned}$$

These equations are the standard interpretation of conjunction, disjunction, and negation. Just note that that the semantics for the $\forall$ connective,

$$|\forall R.C|^t(d) = \min_{d' \in \Delta^{\mathcal{I}}}\{\max\{1 - |R|^t(d,d'), |C|^t(d')\}\} \tag{6}$$

is the result of viewing $\forall R.C$ as the open first order formula $\forall y.F_R(x,y) \to F_C(y)$ (see $(3f)$). Now, the universal quantifier $\forall$ is viewed as a conjunction over the elements of the domain and, thus, $|\forall y.P(y)|^t = \min_{d' \in \Delta^{\mathcal{I}}}\{|P|^t(d')\}$, where P is an unary predicate, whereas the implication $F_R(x,y) \to F_C(y)$ is Zadeh's fuzzy implication connective [11] and, thus, $|F_R(x,y) \to F_C(y)|^t = \max\{1 - |F_R|^t(x,y), |F_C|^t(y)\}$. The combination of these two points yields (6). Concerning the $\exists$ connective, by definition, $|\exists R.C|^t(d)$ is

[2] In the following called interpretation.

$$|\exists R.C|^t(d) = \max_{d' \in \Delta^{\mathcal{I}}}\{\min\{|R|^t(d,d'), |C|^t(d')\}\} \ . \tag{7}$$

The above equation is the result of viewing $\exists R.C$ as the open first order formula $\exists y.F_R(x,y) \wedge F_C(y)$ (see (3g)). Now, the existential quantifier $\exists$ is viewed as a disjunction over the elements of the domain and, thus, $|\exists y.P(y)|^t = \max_{d' \in \Delta^{\mathcal{I}}}\{|P|^t(d')\}$, where P is an unary predicate. Hence, this view yields to (7).

Finally, the interpretation function $(\cdot)^{\mathcal{I}}$ has to satisfy

$$\begin{array}{ll} t \in \langle C(a) \geq n \rangle^{\mathcal{I}} & \text{iff } |C|^t(a^{\mathcal{I}}) \geq n \\ f \in \langle C(a) \geq n \rangle^{\mathcal{I}} & \text{iff } |C|^f(a^{\mathcal{I}}) \geq n \\ t \in \langle R(a,b) \geq n \rangle^{\mathcal{I}} & \text{iff } |R|^t(a^{\mathcal{I}}, b^{\mathcal{I}}) \geq n \\ f \in \langle R(a,b) \geq n \rangle^{\mathcal{I}} & \text{iff } |R|^f(a^{\mathcal{I}}, b^{\mathcal{I}}) \geq n \ . \end{array}$$

Let $\mathcal{I}$ be an interpretation and let $\mathcal{K}$ be a fuzzy KB (a set of fuzzy assertions). $\mathcal{I}$ *satisfies* a fuzzy assertion α, iff $t \in \alpha^{\mathcal{I}}$. $\mathcal{I}$ *satisfies* (*is a model of*) $\mathcal{K}$ iff $\mathcal{I}$ satisfies all elements in $\mathcal{K}$. $\mathcal{K}$ *entails* a fuzzy assertion α (denoted by $\mathcal{K} \approx_4 \alpha$) iff all models of $\mathcal{K}$ satisfy α. The extension of this definition to the case $DB \approx_4 \alpha$ is straightforward. Given $\mathcal{K}$ and an assertion $\mathcal{A}$, we define the *maximal membership degree* of $\mathcal{A}$ w.r.t. $\mathcal{K}$ (denoted by $Maxdeg(\mathcal{K}, \mathcal{A})$) to be $\max\{n > 0 : \mathcal{K} \approx_4 \langle \mathcal{A} \geq n \rangle\}$ ($\max \emptyset = 0$). Notice that $\mathcal{K} \approx_4 \langle \mathcal{A} \geq n \rangle$ iff $Maxdeg(\mathcal{K}, \mathcal{A}) \geq n$. Finally, we define $Maxdeg(DB, \mathcal{A})$ as $\max\{n > 0 : DB \approx_4 \langle \mathcal{A} \geq n \rangle\}$.

Fuzzy specialisations

A *fuzzy specialisation* (denoted by τ) is an expression of type $C \mapsto D$, where C and D are fuzzy $\mathcal{ALC}(OFL, F)$ concepts. A *fuzzy concept definition* is an expression of type $A{:}\approx C$, where A is a primitive concept and C is a concept.

The intended meaning of $C \mapsto D$ is: for all instances a of concept C, if $C(a)$ has membership degree n then $D(a)$ has membership degree n too. That is, we allow a simple bottom up propagation of the membership degree. Formally, given a fuzzy interpretation $\mathcal{I}$, additionally $(\cdot)^{\mathcal{I}}$ has to satisfy

$$\begin{array}{l} t \in (C \mapsto D)^{\mathcal{I}} \text{ iff } \forall d \in \Delta^{\mathcal{I}} \forall n \in (0,1].|C|^t(d) \geq n \text{ implies } |D|^t(d) \geq n \\ f \in (C \mapsto D)^{\mathcal{I}} \text{ iff } \exists d \in \Delta^{\mathcal{I}} \exists n \in (0,1].|C|^t(d) \geq n \text{ and } |D|^t(d) < n \ . \end{array} \tag{8}$$

The definitions of *satisfiability* and entailment are extended to the case we are considering specialisations in the usual way. Finally, an interpretation $\mathcal{I}$ satisfies a fuzzy concept definition $A{:}\approx C$ iff $\mathcal{I}$ satisfies both $A \mapsto C$ and $C \mapsto A$, i.e. $A{:}\approx C$ is a macro for$A \mapsto C$ and $C \mapsto A$.

3.4 Some Properties

We discuss now some properties of our four-valued fuzzy semantics.

Satisfiability: as for the four-valued case, each fuzzy KB is satisfiable. For instance, $\{\langle A(a) \geq .6\rangle, \langle \neg A(a) \geq .8\rangle\}$ is satisfiable, whereas in the classical case (see [34]) it is not.

Soundness of the semantics: our logic is *sound* w.r.t. four-valued semantics and, thus, w.r.t. two-valued semantics. In fact, let $\mathcal{K}$ be a fuzzy KB. With $\mathcal{K}_C$ we indicate the (crisp) KB

$$\mathcal{K}_C = \{\mathcal{A} : \langle \mathcal{A} \geq n\rangle \in \mathcal{K}\} \cup \{C \Rightarrow D : C \mapsto D \in \mathcal{K}\} \ .$$

It is quite easily to verify that if $\mathcal{K} \approx_4 \langle \mathcal{A} \geq n\rangle$ then $\mathcal{K}_C \models_4 \mathcal{A}$, i.e. there cannot be fuzzy entailment without entailment.

Bottom up propagation: bottom up propagation of membership degrees through a taxonomy is supported: for $n > 0$

$$\begin{array}{l} \{\langle C(a) \geq n\rangle, C \mapsto D\} \models\!\!\approx_4 \langle D(a) \geq n\rangle \\ \{\langle A(a) \geq n\rangle, A{:}\approx C\} \models\!\!\approx_4 \langle C(a) \geq n\rangle \\ \{\langle C(a) \geq n\rangle, A{:}\approx C\} \models\!\!\approx_4 \langle A(a) \geq n\rangle \ . \end{array}$$

Contraposition does not hold, i.e.

$$\{\langle \neg D(a) \geq n\rangle, C \mapsto D\} \not\models\!\!\approx_4 \langle C(a) \geq n\rangle \ .$$

Paradoxes: the so-called paradoxes of logical implication do not hold. For $k, m, n > 0$

$$\begin{array}{l} \{\langle C(a) \geq n\rangle, \langle \neg C(a) \geq m\rangle\} \not\models\!\!\approx_4 \langle D(b) \geq k\rangle \\ \emptyset \not\models\!\!\approx_4 \langle (C \sqcup \neg C)(a) \geq k\rangle \ . \end{array}$$

Example 3. Let us consider Example 1. Consider the query concepts Q ("find images about sport"), Q'' ("find images similar to image i3"), $\mathsf{Q}''' = \mathsf{Q}'' \sqcap \mathsf{Q}$ ("find images similar to image i3 and about sport"), and the KB $\mathcal{K}$ in Example 1 in which (*i*) each specialisation $C \Rightarrow D$ has been replaced with the fuzzy specialisation $C \mapsto D$ and each concept definition $A{:}= C$ has been replaced with the fuzzy concept definition $A{:}\approx C$; and (*ii*) $\mathsf{Basket}(\mathsf{b})$ and $\mathsf{Tennis}(\mathsf{t})$ have been replaced with $\langle \mathsf{Basket}(\mathsf{b}) \geq 1\rangle$ and $\langle \mathsf{Tennis}(\mathsf{t}) \geq 1\rangle$, respectively.

This time, we assume that $F{:}\,\mathcal{O} \times \mathcal{T} \rightarrow [0,1]$ is a membership function, which, relying on the image features of image i1 and i2, establishes that $F(\mathsf{i1}, \mathsf{b}) = .9$ and $F(\mathsf{i2}, \mathsf{t}) = .6$. Additionally, we assume that the image similarity function $\mathsf{Sim}_{\mathsf{i3}} \in \mathcal{M}$ is such that $\mathsf{Sim}_{\mathsf{i3}}{:}\,\mathcal{O} \times \{\mathsf{i3}\} \rightarrow [0,1]$ with $\mathsf{Sim}_{\mathsf{i3}}(\mathsf{i1}, \mathsf{i3}) = .4$ and $\mathsf{Sim}_{\mathsf{i3}}(\mathsf{i2}, \mathsf{i3}) = .5$. That is, image i2 is "more" similar to image i3 than image i1 is.

Semantic-based retrieval: it can be verified that both

$$DB \approx_4 \langle \mathsf{Q}(\mathsf{o1}) \geq .9 \rangle$$
$$DB \approx_4 \langle \mathsf{Q}(\mathsf{o2}) \geq .6 \rangle$$

hold. Moreover, $Maxdeg(DB, \mathsf{Q}(\mathsf{o1}))) = .9$, whereas $Maxdeg(DB, \mathsf{Q}(\mathsf{o2}))) = .6$ hold. Therefore, both images i1 and i2 will be retrieved, but image i1 will be ranked before image i2.

Form-based retrieval: it can be verified that both

$$DB \approx_4 \langle \mathsf{Q}''(\mathsf{o1}) \geq .4 \rangle$$
$$DB \approx_4 \langle \mathsf{Q}''(\mathsf{o2}) \geq .5 \rangle$$

hold. Thus, image i2 is ranked before image i1.

Combination of form-based and semantic-based retrieval: it can be verified that both

$$DB \approx_4 \langle \mathsf{Q}'''(\mathsf{o1}) \geq \min\{.9, .4\} \rangle$$
$$DB \approx_4 \langle \mathsf{Q}'''(\mathsf{o2}) \geq \min\{.6, .5\} \rangle$$

hold. Thus, image i2 is ranked before image i1.

■

4 Automatic Reasoning

Deciding whether, given a multimedia database DB and a fuzzy assertion α, $DB \approx_4 \alpha$, requires a calculus. The one we present here is a *sequent calculus* [17]. The calculus is essentially an extension of [33] to the fuzzy case. The main idea behind our calculus for fuzzy entailment is that in order to prove $DB \approx_4 \alpha$ we transform DB into a fuzzy KB, $\mathcal{K}_{DB}$, and then try to prove whether the "sequent" $\mathcal{K}_{DB} \to \alpha$ is valid.

So, let $DB = \langle OFL, OSL, F \rangle$ be a multimedia database, where $OFL = \langle \mathcal{O}, \mathcal{M} \rangle$ is a object form layer, $OSL = \langle \mathcal{T}, \mathcal{K} \rangle$ is a object semantics layer and $F \colon \mathcal{O} \times \mathcal{T} \to [0, 1]$ is a fuzzy membership function. Without loss of generality we will restrict our attention to concepts in *Negation Normal Form* (NNF), where the negation symbol does not appear in front of a complex concept[3] and suppose that all fuzzy concept definitions $A {:}\approx C$ have been replaced with $A \mapsto C$ and $C \mapsto A$.

At first, we transform DB into a fuzzy KB, $\mathcal{K}_{DB}$, such that $DB \approx_4 \alpha$ iff $\mathcal{K}_{DB} \approx_4 \alpha$:

$$\begin{aligned}\mathcal{K}_{DB} = \mathcal{K} \cup\ & \{\langle \mathsf{O}(o) \geq 1 \rangle \colon \langle o, v \rangle \in \mathcal{O}\} \cup \\ & \{\langle \mathsf{IsAbout}(o, a) \geq n \rangle \colon \langle o, v \rangle \in \mathcal{O}, a \in \mathcal{T}, n = F(\langle o, v \rangle, a)\} \cup \\ & \{\langle \mathsf{s}(o_1, o_2) \geq n \rangle \colon \langle o_i, v_i \rangle \in \mathcal{O}, s \in \mathcal{M}, n = s(\langle o_1, v_1 \rangle, \langle o_2, v_2 \rangle)\}\ .\end{aligned}$$

[3] For instance, $\neg(A_1 \sqcap A_2)$ is replaced with $\neg A_1 \sqcup \neg A_2$ and $\neg \exists R.A$ is replaced with $\forall R.\neg A$.

Furthermore, we consider a new alphabet of symbols, called *variables* (denoted by x and y). The alphabet of *individual symbols* is the union of the alphabets of variables, individuals and concrete individuals (individual symbols are denoted by v and w). An interpretation $\mathcal{I}$ is extended to variables by mapping them into an element of its domain $\Delta^{\mathcal{I}}$.

A *sequent* is an expression of the form $\Gamma \rightarrow \Delta$, where $\Gamma = \{\alpha_1, \ldots, \alpha_n, \tau_1, \ldots, \tau_m\}$ and $\Delta = \{\alpha_{n+1}, \ldots, \alpha_{n+k}\}$ are finite sets of fuzzy assertions and specialisations, with $n + k \geq 1, m \geq 0$. Moreover, in these fuzzy assertions may appear both individual symbols and fuzzy assertions of the form $\langle R(v, w) > n\rangle$ (with obvious semantics). Γ is called the *antecedent* and Δ is called the *consequent*. A sequent $\Gamma \rightarrow \Delta$ is *satisfiable* iff there is an interpretation $\mathcal{I}$ such that if $\mathcal{I}$ satisfies *all* elements in Γ then $\mathcal{I}$ satisfies *some* element in Δ. Note that the elements in the antecedent are considered in *and*, whereas the elements in the succedent are considered in *or*. A sequent $\Gamma \rightarrow \Delta$ is *valid* iff all interpretations satisfy $\Gamma \rightarrow \Delta$. A sequent $\Gamma \rightarrow \Delta$ is *falsifiable* iff it is not valid. Please, note that $\mathcal{K} \models\!\!\approx_4 \alpha$ iff the sequent $\mathcal{K} \rightarrow \alpha$ is valid, thus, $DB \models\!\!\approx_4 \alpha$ iff $\mathcal{K}_{DB} \rightarrow \alpha$ valid. For ease of notation we will often omit braces and operations of set-theoretic union, thus writing e.g. $\alpha_1, \Gamma \rightarrow \Delta, \alpha_2$ in place of $\{\alpha_1\} \cup \Gamma \rightarrow \Delta \cup \{\alpha_2\}$.

An *axiom* is a sequent of the form

$$\langle \mathcal{A} \geq n\rangle, \Gamma \rightarrow \Delta, \langle \mathcal{A} \geq m\rangle \ , \tag{9}$$

where $n \geq m$. It is immediate to see that all axioms are valid. A sequent calculus is based on a number of *rules of inference* operating on sequents. Rules fall naturally into two categories: those operating on fuzzy assertions and fuzzy specialisations occurring in the antecedent, and those operating on fuzzy assertions occurring in the consequent. Every rule consists of one "upper" sequent called *premise* and of one or two "lower" sequents called *conclusions*. The rules of the calculus for fuzzy entailment are defined as follows:

$$\begin{array}{ll}
(\sqcap \rightarrow) & \dfrac{\langle (C \sqcap D)(v) \geq n\rangle, \Gamma \rightarrow \Delta}{\langle C(v) \geq n\rangle, \langle D(v) \geq n\rangle, \Gamma \rightarrow \Delta} \\[2ex]
(\rightarrow \sqcap) & \dfrac{\Gamma \rightarrow \Delta, \langle (C \sqcap D)(v) \geq n\rangle}{\Gamma \rightarrow \Delta, \langle C(v) \geq n\rangle \mid \Gamma \rightarrow \Delta, \langle D(v) \geq n\rangle} \\[2ex]
(\sqcup \rightarrow) & \dfrac{\langle (C \sqcup D)(v) \geq n\rangle, \Gamma \rightarrow \Delta}{\langle C(v) \geq n\rangle, \Gamma \rightarrow \Delta \mid \langle D(v) \geq n\rangle, \Gamma \rightarrow \Delta} \\[2ex]
(\rightarrow \sqcup) & \dfrac{\Gamma \rightarrow \Delta, \langle (C \sqcup D)(v) \geq n\rangle}{\Gamma \rightarrow \Delta, \langle C(v) \geq n\rangle, \langle D(v) \geq n\rangle}
\end{array} \tag{10}$$

$$(\forall \rightarrow)\ \frac{\alpha, \langle (\forall R.C)(v) \geq n\rangle, \Gamma \rightarrow \Delta}{\alpha, \langle (\forall R.C)(v) \geq n\rangle, \langle C(w) \geq n\rangle, \Gamma \rightarrow \Delta}$$

$$\text{where } \alpha = \begin{cases} \langle R(v,w) \geq m\rangle \text{ with } m > 1-n, \text{ or} \\ \langle R(v,w) > m\rangle \text{ with } m \geq 1-n \end{cases}$$

$$(\rightarrow \forall)\ \frac{\Gamma \rightarrow \Delta, \langle (\forall R.D)(v) \geq n\rangle}{\langle R(v,x) > 1-n\rangle, \Gamma \rightarrow \Delta, \langle D(x) \geq n\rangle}$$

$$(\exists \rightarrow)\ \frac{\langle (\exists R.C)(v) \geq n\rangle, \Gamma \rightarrow \Delta}{\langle R(v,x) \geq n\rangle, \langle C(x) \geq n\rangle, \Gamma \rightarrow \Delta}$$

$$(\rightarrow \exists)\ \frac{\alpha, \Gamma \rightarrow \Delta, \langle (\exists R.C)(v) \geq n\rangle}{\alpha, \Gamma \rightarrow \Delta, \langle (\exists R.C)(v) \geq n\rangle, \langle C(w) \geq n\rangle}$$

$$\text{where } m \geq n \text{ and } \alpha = \begin{cases} \langle R(v,w) \geq m\rangle, \text{ or} \\ \langle R(v,w) > m\rangle \end{cases}$$

$$(\mapsto\rightarrow)\ \frac{\langle A(v) \geq n\rangle, A \mapsto C, \Gamma \rightarrow \Delta}{\langle A(v) \geq n\rangle, A \mapsto C, \langle C(v) \geq n\rangle, \Gamma \rightarrow \Delta}$$

$$(\rightarrow\mapsto)\ \frac{C \mapsto A, \Gamma \rightarrow \Delta, \langle A(v) \geq n\rangle}{C \mapsto A, \Gamma \rightarrow \Delta, \langle A(v) \geq n\rangle, \langle C(v) \geq n\rangle}$$

where x is a new variable (called also *eigenvariable*) and v, w are individual symbols. Of course, in order to prevent infinite application of the $(\forall \rightarrow)$, $(\rightarrow \exists)$, $(\mapsto\rightarrow)$ and $(\rightarrow\mapsto)$ rules, we assume that each instantiation of the rules is applied only once. Please note that rules $(\rightarrow \sqcap)$ and $(\rightarrow \sqcup)$ introduce a branching.

A deduction can easily be represented as a tree (growing downwards): a *deduction tree* is a tree whose nodes are each labelled with a sequent and in which a sequent labelling a node may be obtained through the application of a rule of inference to the sequent labelling its parent node. The sequent labelling the root of a deduction tree is called *premise* of the deduction tree. A *proof tree* is a deduction tree whose leaves are labelled with an axiom. A sequent $\Gamma \rightarrow \Delta$ is *provable*, written $\Gamma \vdash \Delta$, iff there is a proof tree of which it is the premise. A proof of a sequent $\Gamma \rightarrow \Delta$ proceeds top-down, by constructing a proof tree with root $\Gamma \rightarrow \Delta$ and applying the rules until each branch reaches an axiom.

Example 4. Consider a multimedia database $DB = \langle OFL, OSL, F\rangle$, where $OFL = \langle \mathcal{O}, \mathcal{M}\rangle$, $OSL = \langle \mathcal{T}, \mathcal{K}\rangle$, $\mathcal{O}$ contains an image object $\mathsf{i} = \langle \mathsf{o}, \mathsf{v}\rangle$, $\mathcal{M} = \emptyset$, $\mathcal{T} = \{\mathsf{a}\}$, $\mathsf{i} \in \mathcal{O}$, $F(\mathsf{i}, \mathsf{a}) = .6$ and $\mathcal{K}$ is,

$$\mathcal{K} = \{\langle (\mathsf{Ferrari} \sqcup \mathsf{Porsche})(\mathsf{a}) \geq 1\rangle, \mathsf{Ferrari} \mapsto \mathsf{Car}, \mathsf{Porsche} \mapsto \mathsf{Car}\}\ .$$

Consider the query $\alpha = \langle (\exists \mathsf{IsAbout.Car})(\mathsf{o}) \geq .3 \rangle$, that is, we are asking whether the image i is about a car with degree at least .3. At first, we transform DB into a fuzzy KB, $\mathcal{K}_{DB}$:

$$\mathcal{K}_{DB} = \mathcal{K} \cup \{\langle \mathsf{O}(\mathsf{o}) \geq 1 \rangle, \langle \mathsf{IsAbout}(\mathsf{o}, \mathsf{a}) \geq .6 \rangle\}$$

Straightforwardly, $\mathcal{K}_{DB} \models\!\!\approx_4 \alpha$ holds. The following is a proof tree[4] of $\mathcal{K}_{DB} \rightarrow \alpha$.

$$\begin{array}{c}
\mathcal{K}_{DB} \rightarrow \alpha \\
\hline
\begin{array}{c|c}
\langle \mathsf{F}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_{DB} \rightarrow \alpha & \langle \mathsf{P}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_{DB} \rightarrow \alpha \\
\hline
\langle \mathsf{C}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_1, \mathcal{K}_{DB} \rightarrow \alpha & \langle \mathsf{C}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_2, \mathcal{K}_{DB} \rightarrow \alpha \\
\hline
\langle \mathsf{C}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_1, \mathcal{K}_{DB} \rightarrow \alpha, \langle \mathsf{C}(\mathsf{a}) \geq .3 \rangle & \langle \mathsf{C}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_2, \mathcal{K}_{DB} \rightarrow \alpha, \langle \mathsf{C}(\mathsf{a}) \geq .3 \rangle
\end{array}
\end{array}$$

where $\mathcal{K}_1 = \{\langle \mathsf{F}(\mathsf{a}) \geq 1 \rangle\}$, $\mathcal{K}_2 = \{\langle \mathsf{P}(\mathsf{a}) \geq 1 \rangle\}$.

On the other hand, by considering $\alpha' = \langle (\exists \mathsf{About.Ferrari})(\mathsf{o}) \geq .3 \rangle$, we have $\mathcal{K}_{DB} \not\models\!\!\approx_4 \alpha'$. The following is a deduction tree of $\mathcal{K}_{DB} \rightarrow \alpha'$ not being a proof tree.

$$\begin{array}{c}
\mathcal{K}_{DB} \rightarrow \alpha' \\
\hline
\begin{array}{c|c}
\langle \mathsf{F}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_{DB} \rightarrow \alpha' & \langle \mathsf{P}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_1, \mathcal{K}_{DB} \rightarrow \alpha' \\
\hline
\langle \mathsf{C}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_1, \mathcal{K}_{DB} \rightarrow \alpha' & \langle \mathsf{C}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_2, \mathcal{K}_{DB} \rightarrow \alpha' \\
\hline
\langle \mathsf{C}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_1, \mathcal{K}_{DB} \rightarrow \alpha', \langle \mathsf{F}(\mathsf{a}) \geq .3 \rangle & \langle \mathsf{C}(\mathsf{a}) \geq 1 \rangle, \mathcal{K}_2, \mathcal{K}_{DB} \rightarrow \alpha', \langle \mathsf{F}(\mathsf{a}) \geq .3 \rangle
\end{array}
\end{array}$$

Note that the branch on the right does not end up with an axiom. ■

Theorem 1. *A sequent $\Gamma \rightarrow \Delta$ is valid iff $\Gamma \vdash \Delta$. Moreover, validity is decidable.* ⊣

Proof. It can be verified that any deduction tree is finite. Therefore, the deduction of a sequent terminates after a finite number of rule applications.

Soundness of the calculus, i.e. if the sequent $\Gamma \rightarrow \Delta$ is provable then $\Gamma \rightarrow \Delta$ is valid, can easily be proven by observing that every axiom is valid and that for each of the rules, a conclusion of a rule is valid iff the premise of the rule is valid.

Now, if there is a proof tree for $\Gamma \rightarrow \Delta$ then, from the correctness of the rules we have that $\Gamma \rightarrow \Delta$ is valid. Otherwise, pick up a deduction tree not being a proof tree, and which cannot be expanded any more. Therefore, there is a path from the premise to a non axiom leaf of the tree. Let LHS be the union of all fuzzy assertions occurring in the left hand side of each sequent along that path and RHS be the union of all fuzzy assertions occurring in the right hand side of any such sequent. Let $S = LHS \cup RHS$. From S an

[4] A, F, P and C stand for IsAbout, Ferrari, Porsche and Car, respectively.

interpretation $\mathcal{I}$ can be build satisfying LHS and not satisfying RHS (see below). Since $\Gamma \subseteq LHS$ and $\Delta \subseteq RHS$, it follows that $\Gamma \rightarrow \Delta$ is falsifiable.

Let us build $\mathcal{I}$. At first, define

$$\begin{array}{l} S_{\geq} = \{\langle \mathcal{A} \geq n \rangle \in LHS\}, \\ S_{>} = \{\langle \mathcal{A} > n \rangle \in LHS\} \ . \end{array}$$

Please note that $S_{>}$ contains fuzzy assertions of the form $\langle R(v,w) > n \rangle$ only. Define, for $\epsilon > 0$

$$n^{\geq}_{\mathcal{A}} = \max\{n : \langle \mathcal{A} \geq n \rangle \in S_{\geq}\}, \text{ and}$$

$$n^{>}_{\mathcal{A}} = \max\{n : \langle \mathcal{A} > n \rangle \in S_{>}\} + \epsilon \ .$$

Let $\mathcal{I}$ be a relation such that the domain $\Delta^{\mathcal{I}}$ of $\mathcal{I}$ is the set of individuals or variables appearing in S union $\mathcal{O} \cup \{\langle \mathsf{o}, \mathsf{nil} \rangle\}$ (nil is the null value, o is a new object identifier). For all individual symbols w, $w^{\mathcal{I}} = w$ if w individual or variable, $o^{\mathcal{I}} = \langle o, v \rangle$ if $\langle o, v \rangle \in \mathcal{O}$, $\mathsf{o}^{\mathcal{I}} = \langle \mathsf{o}, \mathsf{nil} \rangle$ otherwise. For each atomic concept A, for each role R, for all $w, v \in \Delta^{\mathcal{I}}$ define ($\max \emptyset = 0$)

$$\begin{array}{l} |A|^t(w) = \max\{n^{\geq}_{A(w)}, n^{>}_{A(w)}\}, \\ |A|^f(w) = 0, \\ |R|^t(w,v) = \max\{n^{\geq}_{R(w,v)}, n^{>}_{R(w,v)}\}, \text{ and} \\ |R|^f(w,v) = 0 \ . \end{array}$$

It can be verified that $\mathcal{I}$ is a four-valued fuzzy interpretation and there is $\epsilon > 0$ such that $\mathcal{I}$ satisfies all $\alpha \in S_{\geq}$, $\mathcal{I}$ satisfies all $\alpha \in S_{>}$ and, thus, LHS, whereas $\mathcal{I}$ does not satisfy any $\alpha \in RHS$. Q.E.D.

Example 5. Consider Example 4, case α'. We build an interpretation $\mathcal{I}$ falsifying $\mathcal{K}_{DB} \rightarrow \alpha'$ by relying on the right hand branch of the deduction tree. According to Theorem 1, the domain of $\mathcal{I}$ is $\Delta^{\mathcal{I}} = \{\mathsf{i}, \mathsf{a}, \langle \mathsf{o}, \mathsf{nil} \rangle\}$, $|\mathsf{A}|^t(\mathsf{o}, \mathsf{a}) = .6$, and $|\mathsf{P}|^t(\mathsf{a}) = 1$ and $|\mathsf{C}|^t(\mathsf{a}) = 1$. In all other cases the value is 0, both for $|\cdot|^t$ as well as for $|\cdot|^f$. Clearly, $\mathcal{I}$ does not satisfy α', as $|\mathsf{F}|^t(\mathsf{a}) = 0$. ■

We conclude the section by showing how to compute $Maxdeg(DB, C(a))$. The problem of determining $Maxdeg(DB, C(a))$ is important, as computing it is in fact the way to answer a query of type "to which degree is a a C, given the multimedia database DB?". An easy algorithm can be given in terms of a sequence of fuzzy entailment tests. Our algorithm is based on the observation that

$$\begin{array}{l} Maxdeg(DB, C(a)) \in N_{\mathcal{K}_{DB}}, \text{ where} \\ N_{\mathcal{K}_{DB}} = \{0, .5, 1\} \cup \{n : \langle \mathcal{A} \geq n \rangle \in \mathcal{K}_{DB}\} \ . \end{array} \quad (11)$$

The algorithm, a simple binary search on $N_{\mathcal{K}_{DB}}$, is described below.

Algorithm $Max(\mathcal{K}_{DB}, C(a))$
Set $Min := 0$ *and* $Max := 2$.

1. *Pick* $n \in N_{\mathcal{K}_{DB}} \cup \{.5, 1\}$ *such that* $Min < n < Max$. *If there is no such* n, *then set* $Maxdeg(\mathcal{K}_{DB}, C(a)) := Min$ *and exit.*
2. *If* $\mathcal{K}_{DB} \approx_4 \langle C(a) \geq n \rangle$ *then set* $Min = n$ *and go to Step 1, else set* $Max = n$ *and go to Step 1.* ∎

By a binary search on $N_{\mathcal{K}_{DB}}$ the value of $Maxdeg(DB, C(a))$ can be determined in at most $\log |N_{\mathcal{K}_{DB}} + 2|$ fuzzy entailment tests.

5 Conclusions

We have presented a logic-based MIR model in which all three kind of retrieval, form-based retrieval, semantic-based retrieval and their combination, are integrated in a principled way. We rely on a two-layer model for the representation of multimedia object's properties: the object form layer, which collects the medium dependent features of multimedia objects, and the object semantics layer, which collects symbolic representations of the slice of the real world being represented. The logic we have presented, for both representing the properties of the object semantics layer and to retrieve multimedia objects, is a four-valued fuzzy DL. The logic, is characterised by

- a description logic component which allows the representation of structured objects of the real world;
- a non-classical, four-valued, semantics which (*i*) allows us to deal with possible inconsistencies arising from the representation of multimedia object's content; (*ii*) enforces a notion of entailment, the pertinence of premises to conclusions, to a stronger extent than classical "material" logical implication does; and
- a fuzzy component which allows the treatment of the inherent imprecision on the aboutness of multimedia objects representation.

Compliant to [2], our work maybe understood as a contribution towards the development of intelligent multimedia retrieval systems, where the combination of database techniques, information retrieval, artificial intelligence (in particular, knowledge representation and machine learning) plays an important role.

We are aware that our logic, and logics in general, have a main drawback: their computational complexity. Reverting to an appropriate sublogic of our logic, may be taken as an interesting topic for further research. A good starting point may be those DLs for which the instance checking problem is known to be solvable in polynomial time (see e.g. [15]).

References

1. Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison Wesley Publ. Co., Reading, Massachussetts, 1995.
2. Wasfi Al-Khatib, Y. Francis Day, Arif Ghafoor, and P. Bruce Berra. Semantic modeling and knowledge representation in multimedia databases. *IEEE Transactions on Knowledge and Data Engineering*, 11(1):64–80, 1999.
3. Alan R. Anderson and Nuel D. Belnap. *Entailment - the logic of relevance and necessity*. Princeton University Press, Princeton, NJ, 1975.
4. T.M. Anwar, H.W. Beck, and S.B. Navathe. Knowledge mining by imprecise querying: A classification-based approach. In *Proceedings of the 8th International Conference on Data Engineering*, pages 622–630, 1992.
5. Franz Baader and Philipp Hanschke. A schema for integrating concrete domains into concept languages. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI-91)*, pages 452–457, Sydney, 1991.
6. Nuel D. Belnap. A useful four-valued logic. In Gunnar Epstein and J. Michael Dunn, editors, *Modern uses of multiple-valued logic*, pages 5–37. Reidel, Dordrecht, NL, 1977.
7. T. Bollinger and U. Pletat. The LILOG knowledge representation system. *ACM SIGART Bulletin*, 2(3):22–27, 1991.
8. G. Bordogna, P Carrara, and G. Pasi. Query term weights as constraints in fuzzy information retrieval. *Information Processing and Management*, 27(1):15–26, 1991.
9. Ronald J. Brachman, P. G. Selfridge, L. G. Terveen, B. Altman, Alex Borgida, F. Halpern, T. Kirk, A. Lazar, Deborah L. McGuinness, and Lori Alperin Resnick. Knowledge representation support for data archeology. In Y. Yesha, editor, *Proc. of the Int. Conf. on Information and Knowledge Management (CIKM-92)*, pages 457–464, 1992.
10. Martin Buchheit, Francesco M. Donini, and Andrea Schaerf. Decidable reasoning in terminological knowledge representation systems. *Journal of Artificial Intelligence Research*, 1:109–138, 1993.
11. Jianhua Chen and Sukhamany Kundu. A sound and complete fuzzy logic system using Zadeh's implication operator. In Zbigniew W. Ras and Michalewicz Maciek, editors, *Proc. of the 9th Int. Sym. on Methodologies for Intelligent Systems (ISMIS-96)*, number 1079 in Lecture Notes In Artificial Intelligence, pages 233–242. Springer-Verlag, 1996.
12. Wesley W. Chu, Alfonso F. Cardenas, and Ricky K. Taira. Knowledge-based image retrieval with spatial and temporal constructs. *IEEE Transactions on Knowledge and Data Engineering*, 10(6):872–888, 1998.
13. Premkumar Devanbu, Ronald J. Brachman, Peter J. Selfridge, and Bruce W. Ballard. LASSIE: A knowledge-based software information system. *Communications of the ACM*, 34(5):36–49, 1991.
14. DL. Description Logic Web Home Page: `http://dl.kr.org/dl`, WWW.
15. Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Werner Nutt. Tractable concept languages. In *Proceedings of IJCAI-91, 12th International Joint Conference on Artificial Intelligence*, pages 458–463, Sidney, Australia, 1991.
16. M. Olivier Duschka and Y. Alon Levy. Recursive plans for information gathering. In *Proc. of the 15th Int. Joint Conf. on Artificial Intelligence (IJCAI-97)*, pages 778–784, Nagoya, Japan, 1997.

17. Jean H. Gallier. *Logic for Computer Science: Foundations of Automatic Theorem Proving*. Harper & Row Publishers, New York, 1986.
18. C. A. Gobel, C. Haul, and S. Bechhofer. Describing and classifying multimedia using the description logic GRAIL. In *Proceedings of the SPIE Conference on Storage and Retrieval for Still Images and Video Databases IV (SPIE-96)*, pages 132–143, San Jose, CA, February 1996.
19. E.E. Kerre, R.B. Zenner, and R.M. De Caluwe. The use of fuzzy set theory in information retrieval and databases: a survey. *Journal of the American Society for Information Science*, 37(5):341–345, 1986.
20. Craig A. Knoblock, Steven Minton, Ambite Jose Luis, Naveen Ashish, Pragnesh Jay Modi, Muslea Ion, Andrew G. Philpot, and Sheila Tejada. Modeling web sources for information integration. In *Proc. of the 15th Nat. Conf. on Artificial Intelligence (AAAI-98)*, pages 211–218, 1998.
21. Donald H. Kraft and Duncan Buel. Fuzzy sets and generalised boolean retrieval systems. *Int. J. Man-Machine Studies*, 19:45–56, 1983.
22. P Lambrix and J. Maleki. Learning composite concepts in description logics: A first step. In *Proceedings of the 9th International Symposium on Methodologies for Intelligent Systems - ISMIS 96, LNAI 1079*, pages 68–77, Zakopane, Poland, 1996.
23. Hector J. Levesque. A logic of implicit and explicit belief. In *Proc. of the 3th Nat. Conf. on Artificial Intelligence (AAAI-84)*, pages 198–202, Austin, TX, 1984.
24. Carlo Meghini, Fabrizio Sebastiani, and Umberto Straccia. Modelling the retrieval of structured documents containing texts and images. In *Proceedings of the First European Conference on Research and Advanced Technology for Digital Libraries*, number 1324 in Lecture Notes in Computer Science, pages 325–344, Pisa, September 1997.
25. Carlo Meghini, Fabrizio Sebastiani, and Umberto Straccia. MIRLOG: A logic for multimedia information retrieval. In F. Crestani, M. Lalmas, and C.J. van Rijsbergen, editors, *Logic and Uncertainty in Information Retrieval: Advanced models for the representation and retrieval of information*, volume 4 of *The Kluwer International Series On Information Retrieval*, chapter 7, pages 151–185. Kluwer Academic Publishers, Boston, MA, 1998.
26. Carlo Meghini and Umberto Straccia. A relevance terminological logic for information retrieval. In *Proceedings of SIGIR-96, 19th International Conference on Research and Development in Information Retrieval*, pages 197–205, Zurich, Switzerland, 1996.
27. C.V. Negoita and P. Flondor. On fuzziness in information retrieval. *Int. J. Man-Machine Studies*, 8:711–716, 1976.
28. Peter F. Patel-Schneider. A four-valued semantics for terminological logics. *Artificial Intelligence*, 38:319–351, 1989.
29. Gerard Salton and J. Michael McGill. *Introduction to Modern Information Retrieval*. Addison Wesley Publ. Co., Reading, Massachussetts, 1983.
30. Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48:1–26, 1991.
31. Fabrizio Sebastiani. A probabilistic terminological logic for modelling information retrieval. In *Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval*, pages 122–130, Dublin, IRL, 1994. Published by Springer Verlag, Heidelberg, FRG.

32. Umberto Straccia. A four-valued fuzzy propositional logic. In *Proc. of the 15th Int. Joint Conf. on Artificial Intelligence (IJCAI-97)*, pages 128–133, Nagoya, Japan, 1997.
33. Umberto Straccia. A sequent calculus for reasoning in four-valued description logics. In *Proc. of the Int. Conf. on Analytic Tableaux and Related Methods (TABLEAUX-97)*, number 1227 in Lecture Notes In Artificial Intelligence, pages 343–357, Pont-à-Mousson, France, 1997.
34. Umberto Straccia. A fuzzy description logic. In *Proc. of the 15th Nat. Conf. on Artificial Intelligence (AAAI-98)*, pages 594–599, 1998.
35. Robert Weida and Diane Litman. Terminological reasoning with constraint networks and an application to plan recognition. In *Proc. of the 3rd Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR-92)*, pages 282–293. Morgan Kaufmann, Los Altos, 1992.
36. Cristopher Welty. Description logics for digital libraries. In *International Workshop on Description Logics*, Trento, Italy, 1998.
37. Ronald R. Yager. Fuzzy sets as a tool for modeling. In Jan van Leeuwen, editor, *Computer Science Today*, number 1000 in Lecture Notes in Computer Science, pages 536–548. Springer-Verlag, 1995.
38. L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.
39. A. Zeller. Versioning system models through description logic. In *Proc. 8th International Symposium on System Configuration Management (SCM-8)*, Brussels, 1988.