The Terminological Image Retrieval Model*

Carlo Meghini, Fabrizio Sebastiani and Umberto Straccia

Consiglio Nazionale delle Ricerche
Istituto di Elaborazione dell'Informazione
Via S. Maria 46, I-56126 Pisa, Italy,
E-mail: ⟨*lastname*⟩@iei.pi.cnr.it

Abstract. We present a model for image retrieval in which images are represented both at the *form* level, as sets of physical features of the *representing* objects, and at the content level, as sets of logical assertions about the *represented* entities as well as about facts of the subject matter that are deemed as relevant for retrieval. A uniform and powerful query language allows queries to be issued that transparently combine features pertaining to form and content. Queries are expressions of a fuzzy logical language. While that part of the query that pertains to (medium-independent) content is "directly" processed by an inferential engine, that part that pertains to (medium-dependent) form is entrusted to specialised signal processing procedures linked to the logical language by a *procedural attachment* mechanism.

1 Introduction

Due to the pervasive role of images in nowadays information systems, a vast amount of research has been carried out in the last few years on methods for retrieving images by content from large repositories. This research has produced many theoretical results, on top of which a first generation of image retrieval systems (IRSs, for short) have been built [7] and, in some cases, even turned into commercial products [2, 5]. The distinguishing feature of these systems, and of the related research prototypes, is their total disregard for a proper representation and use of *image semantics*.

This study addresses the problem of injecting semantics into image retrieval by presenting an image retrieval model in which images are represented both at the form level, as sets of physical features of the objects *representing* a slice of the world, and at the content level, as sets of properties of the real-world objects *being represented*. This model is logic-based, in the sense that the representation of image content is based on a *description logic*. Features of images pertaining to form are not represented explicitly in the description logic, as they are best dealt with outside it, i.e. by means of some digital signal processing technique. However, they impact on logical reasoning through a mechanism of "procedural

* This work has been carried out in the context of the project FERMI (n. 8134): "Formalization and Experimentation in the Retrieval of Multimedia Information", funded by the European Union under the ESPRIT Basic Research scheme.

attachments" [1], which implements the connection between (logical) reasoning about content and (non-logical) reasoning about form, thus allowing a unified query language capable of addressing both dimensions.

The resulting retrieval capability thus extends that of current IRSs with the use of semantic information processing and reasoning about image content. So far, the only attempts in this direction had been based on textual annotations to images ("captions": see e.g. [13]) or their regions, in some cases supported by the use of thesauri to semantically connect the terms occurring in the text [8]. These models permit the expression of image contents, but are weak in exploiting them, due to the well-known limitations of keyword-based text retrieval [14].

2 Representing image form

Let $\mathbb{N}$ be the set of natural numbers. A *region* is any subset of $\mathbb{N}^2$, i.e. a set of *points*. A region S is *aligned* if $min\{x \mid (x,y) \in S\} = 0$ and $min\{y \mid (x,y) \in S\} = 0$. We assume familiarity with the basic notions of digital geometry, such as neighborhood and connectedness (for details, see e.g. [11, Chapter 11]). A connected set with no "holes" is called *simply connected.*

Given a set of colours $\mathcal{C}$, a *layout* is a triple $i = \langle A^i, \pi^i, f^i \rangle$, where A^i, the *domain,* is a finite, aligned, rectangular region; π^i is a partition of A^i into non-empty connected regions $\{T_1, ..., T_n\}$, called *atomic regions;* f^i is a total function from π^i to $\mathcal{C}$, assigning a colour to each atomic region (and therefore called the *colour function*) such that no two neighbour atomic regions have the same colour; formally:

$$\forall T, T' \in \pi^i, \text{ if } T \text{ is a neighbour of } T' \text{ then } f^i(T) \neq f^i(T')$$

For notational convenience, we make explicit some of the information carried by a layout: given the layout $i = \langle A^i, \pi^i, f^i \rangle$,

- the *extended regions* π^i_e of i are defined as

  $$\pi^i_e = \{S \mid \exists T_1, ..., T_k \in \pi^i, k \geq 1,\ S = \cup_{i=1}^k T_i,\ S \text{ connected}\}$$

  The fact that we do not require S to be *simply* connected allows some interesting visual objects (e.g. the figure of a goalkeeper partly covered by an approaching ball) to be classified as extended regions;
- the *extended colour function* f^i_e of a layout i is defined as the function that assigns to each extended region S a *colour distribution* $f^i_e(S)$ (i.e. a mapping from $\mathcal{C}$ to [0,1] such that $\sum_{\{c \in \mathcal{C}\}} f^i_e(S)(c) = 1$) as follows: $\forall c \in \mathcal{C}$, $\forall S \in \pi^i_e$ such that $S = \cup_{j=1}^k T_j$ and each T_j is an atomic region:

  $$f^i_e(S)(c) = \frac{\sum_{T_j \in Z} |T_j|}{|S|}$$

  where Z is the set containing all and only the atomic regions T_j in $\{T_1, \ldots, T_k\}$ that have colour c, *i.e.* $f^i(T_j) = c$, and $|S|$ refers to the cardinality of a region S viewed as a set of points.

In general, a region S is not bound to a particular layout. This binding is realized in the notion of *grounded region*, which we define as a pair $\langle i, S\rangle$, where $i = \langle A^i, \pi^i, f^i\rangle$ is a layout and $S \in \pi^i_e$.

Let $[k]$ denote the set of the first k natural numbers. Given $m, n \in \mathbb{N}$, the *image space* $\mathbf{M}(m, n)$ is given by the set of all possible layouts of domain $[m]\times[n]$. The *image universe* $\mathcal{U} = \cup_{(i,j)\in\mathbb{N}^2}\mathbf{M}(i, j)$ is the union of all possible image spaces.

3 Representing image contents

We take the content of an image to be a *scene*, i.e. a set of possible situations indistinguishable from the visual point of view. The formalism we have chosen for representing and reasoning on image contents is a *Description Logic* (DL), namely the logic is $\mathcal{ALC}$ [12], a significant representative of the DLs family; however, our model is not tied in any way to this particular choice, and any other DL would easily fit in it. The language of $\mathcal{ALC}$ includes unary and binary predicate symbols, called *primitive concepts* (indicated by the metavariable A with optional subscripts) and *primitive roles* (metavariable R), respectively. These are the basic constituents by means of which *concepts* (i.e. "non-primitive predicate symbols") are built via *concept constructors*, according to the following syntactic rule:

$$C \longrightarrow A \mid C_1 \sqcap C_2 \mid \neg C \mid \forall R.C$$

A *crisp assertion* is an expression having one of the following forms:

- $C(a)$, where a is an *individual* and C is a concept, means that a is an instance of C; for example, `(Musician ⊓ Teacher)(tim)` makes the individual `tim` a `Person` and a `Teacher`;
- $R(a_1, a_2)$, where a_1 and a_2 are individuals and R is a role, means that a_1 is related to a_2 by means of R (e.g. `Friend(tim,tom)`);
- $T \sqsubseteq T'$, where T and T' are both concepts or both roles, means that T is a specialization of T' (e.g. `PianoPlayer` $\sqsubseteq$ (`Musician` $\sqcap$ ($\exists$`Plays.Keyboard`))).

The first two kinds of assertions are called *simple assertions,* while any instance of the last kind is said to be an *axiom.* In order to deal with the uncertainty inherent in similarity-based retrieval, we introduce in the logic *fuzzy assertions* (see e.g. [4]), *i.e.* expressions of the form $\langle\alpha, n\rangle$ where α is a crisp assertion and $n \in [0, 1]$, meaning that α is true "to degree n". We will use the terms *fuzzy simple assertion* and *fuzzy axiom*, with the obvious meaning.

The semantics of the resulting logic relies on *fuzzy interpretations*, i.e. pairs $\mathcal{I} = (\Delta^\mathcal{I}, (\cdot)^\mathcal{I})$ where $\Delta^\mathcal{I}$ is a non-empty set (called the *domain of discourse*) including the image universe $\mathcal{U}$, and $(\cdot)^\mathcal{I}$, the *interpretation function,* maps each concept into a function from $\Delta^\mathcal{I}$ to $[0, 1]$, and each role into a function from $\Delta^\mathcal{I} \times \Delta^\mathcal{I}$ to $[0, 1]$, so that for all $d \in \Delta^\mathcal{I}$ the following conditions are satisfied:

$$(C_1 \sqcap C_2)^{\mathcal{I}}(d) = min\{{C_1}^{\mathcal{I}}(d), {C_2}^{\mathcal{I}}(d)\}$$
$$(\neg C)^{\mathcal{I}}(d) = 1 - C^{\mathcal{I}}(d)$$
$$(\forall R.C)^{\mathcal{I}}(d) = min_{d' \in \Delta^{\mathcal{I}}}\{max\{1 - R^{\mathcal{I}}(d, d'), C^{\mathcal{I}}(d')\}\}$$

A fuzzy interpretation $\mathcal{I}$ *is a model of* an assertion $\langle C(a), n\rangle$ ($\langle R(a_1, a_2), n\rangle$, $\langle T \sqsubseteq T', n\rangle$, respectively) iff $C^{\mathcal{I}}(a^{\mathcal{I}}) \geq n$ (resp. $R^{\mathcal{I}}({a_1}^{\mathcal{I}}, {a_2}^{\mathcal{I}}) \geq n$; for all $d \in \Delta^{\mathcal{I}}$, ${T'}^{\mathcal{I}}(d) \geq n \cdot T^{\mathcal{I}}(d)$), and *is a model of* a set of fuzzy assertions iff $\mathcal{I}$ is a model of all the assertions in the set. A set of fuzzy assertions Σ *entails* a fuzzy assertion $\langle \alpha, n\rangle$ (written $\Sigma \models^f \langle \alpha, n\rangle$) iff all models of Σ are models of $\langle \alpha, n\rangle$. Given Σ and a crisp assertion β, we define the *maximal degree of truth* of β w.r.t. Σ (written $Maxdeg(\Sigma, \beta)$) to be $n \geq 0$ iff $\Sigma \models^f \langle \beta, n\rangle$ and there is no $m > n$ such that $\Sigma \models^f \langle \beta, m\rangle$.

Having settled for the tool, we now specify its use for image content representation. Let i be a layout uniquely identified, in a way to be made precise later, by the individual `i`. A *content description* δ for i is a set of fuzzy assertions, consisting of the union of four component subsets:

1. the *image identification,* a set containing only a single fuzzy assertion of the form $\langle$`Ego(i)`$, 1\rangle$, whose role is to associate, along with the layout naming function n_l (see Section 6), a content description with the layout it refers to. In particular, in what follows σ(`i`) will denote the set of the (possibly many) content descriptions whose identification is `Ego(i);`
2. the *object anchoring,* a set of fuzzy assertions of the form $\langle$`Rep(r,o)`$, n\rangle$, where `r` is an individual that uniquely identifies a grounded region of i and `o` is an individual that identifies the object depicted by the region;
3. the *scene anchoring,* a set of fuzzy assertions of the form $\langle$`About(i,o)`$, n\rangle$, where `i` and `o` are as above. By using these assertions, an indexer can state what the whole scene shown in the image is about, and this would typically be a situation of which the image shows some salient aspect;
4. the *scene description,* a set of fuzzy simple assertions (where neither the predicates `Ego`, `Rep` and `About`, nor identifiers pertaining to layout such as the `i`'s and `r`'s above, occur), describing important facts shown in the image about the individuals identified by assertions of the previous two kinds.

While the task of components 1 to 3 is that of binding the form and content dimension of the same image, component 4 pertains to the content dimension only. Note that there may be more than one content description for the same image i; this is meant to reflect the fact that there may be multiple viewpoints under which an image may be considered.

Any of components 2 to 4 can be missing in a content description. As an example, let us consider a photograph showing a singer, Mary, performing as Zerlina in Mozart's "Don Giovanni". Part of a plausible content description for this image, named `i`, could be (for simplicity, in this example we only use crisp assertions):

{`Ego(i),About(i,o),Rep(r,mary),DonGiovanni(o),Plays(mary,zerlina)`}

4 Querying layouts

A query addressed to an image base can refer either to the form dimension, in which case we call it a *visual* query, or to the content dimension, in which case we call it a *conceptual* query. These two categories are exhaustive but not disjoint. Visual queries can be partitioned in: *concrete visual queries:* these consist of images themselves that are submitted to the system as a way to indicate a request to retrieve "similar" images; and *abstract visual queries:* these are abstractions of layouts that address specific aspects of image similarity via artificially constructed image elements and can be further categorised into:

1. *colour queries:* colour distributions that are used to retrieve images with a similar colour distribution;
2. *shape queries:* specifications of one or more shapes (closed simple curves in the 2D space) and possibly of their spatial relationships, used to retrieve images in which the specified shapes occur as contours of significant objects, in the specified relationships;

and other categories, such as spatial and texture queries [6], which will not be dealt with in this paper.

In order to query layouts, the following SPSs are introduced:

- *symbols for global matching:* in general, there will be a set of such symbols, each capturing a specific similarity criterion. Since from the conceptual viewpoint these symbols form a uniform class, we will just include one of them in our language, to be understood as the representative of the whole class. Any other symbol of the same sort can be added without altering the structure and philosophy of the language. So, for global matching we use the SPS
  - `SI(i,j)` (standing for `Similar Image`): assesses the similarity between two layouts `i` and `j`;
- *symbols for local matching:* these come in two sorts. First we have *selectors,* which are SPSs needed to select the entity to match from a layout:
  - `HAR(i,r)` (`Has Atomic Region`): a selector relating the image `i` to any of its grounded atomic regions `r`;
  - `HR(i,r)` (`Has Region`): relates `i` to any of its grounded regions `r`;
  - `HC(r,c)` (`Has Colour`): relates the grounded region `r` to its colour `c`;
  - `HS(r,s)` (`Has Shape`): relates the grounded region `r` to its shape `s`.

  Second, we have symbols for local matching, assessing similarity between local entities. Similarly for what it has been done for global matching, we include one symbol for each category of entities to be matched; so we have:
  - `SC(c,c')` (`Similar Colour`): returns the similarity between colour distributions `c` and `c'`;
  - `SS(s,t)` (`Similar Shape`): gives the similarity between shapes `s` and `t`.

The semantics of the symbols introduced so far is fixed, and is given by the functions that capture the intended meaning of each symbol, as illustrated above. For example, if $\mathcal{I}$ is any fuzzy interpretation:

$\texttt{SI}^{\mathcal{I}} : \mathcal{U} \times \mathcal{U} \rightarrow [0,1]$, assigning to each pair of layouts their degree of similarity.

A fuzzy interpretation $\mathcal{I}$ is said to be an *image interpretation* if and only if it assigns the correct semantics to the SPSs. From now on, we will use the term "interpretation" as short for "image interpretation".

5 The query language

Below, we present the *query language* of our model (*cpt* abbreviates *concept*).

$\langle$*image-q*$\rangle$ ::= $\langle$*image-cpt*$\rangle$ | $\langle$*image-q*$\rangle \sqcap \langle$*image-q*$\rangle$ | $\langle$*image-q*$\rangle \sqcup \langle$*image-q*$\rangle$
$\langle$*image-cpt*$\rangle$ ::= $\exists$SI.{$\langle$*layout-name*$\rangle$} | $\exists$About.$\langle$*content-cpt*$\rangle$ |
$\exists$HAR.$\langle$*region-cpt*$\rangle$ | $\exists$HR.$\langle$*bound-region-cpt*$\rangle$
$\langle$*region-cpt*$\rangle$ ::= $\exists$HC.$\langle$*colour-cpt*$\rangle$ | $\exists$HS.$\langle$*shape-cpt*$\rangle$ | $\exists$Rep.$\langle$*content-cpt*$\rangle$ |
$\langle$*region-cpt*$\rangle \sqcap \langle$*region-cpt*$\rangle$ | $\langle$*region-cpt*$\rangle \sqcup \langle$*region-cpt*$\rangle$
$\langle$*bound-region-cpt*$\rangle$ ::= $\exists$Rep.$\langle$*content-cpt*$\rangle$ | $\langle$*bound-region-cpt*$\rangle \sqcap \langle$*region-cpt*$\rangle$ |
$\langle$*bound-region-cpt*$\rangle \sqcup \langle$*region-cpt*$\rangle$
$\langle$*colour-cpt*$\rangle$ ::= {$\langle$*colour-name*$\rangle$} | $\exists$SC.{$\langle$*colour-name*$\rangle$}
$\langle$*shape-name*$\rangle$ ::= {$\langle$*shape-name*$\rangle$} | $\exists$SS.{$\langle$*shape-name*$\rangle$}

Note that a *layout-name*, a *colour-name* and a *shape-name* are not concepts, but individuals. Queries are thus *not* concepts of $\mathcal{ALC}$, but of the DL $\mathcal{ALCO}$, which extends $\mathcal{ALC}$ with the "*singleton*" {} operator, which given an individual `i` returns a concept {`i`}. The singleton operator is necessary in queries because it allows the reference to specific individuals. However, this added expressive power has no impact on the complexity of the image retrieval problem.

Let us reconsider the example introduced in Section 3. The images about Don Giovanni are retrieved by the query $\exists$`About.DonGiovanni`. Those showing the singer Mary are described by $\exists$`HR.`$\exists$`Rep.{mary}`. Turning to visual queries, the request to retrieve the images similar to a given one, named `this`, is expressed by $\exists$`SI.{this}`, and can be easily combined with any conceptual query, e.g. yielding $\exists$`SI.{this}` $\sqcup$ $\exists$`About.DonGiovanni`, which would retrieve the images that are either similar to the given one or are about Don Giovanni. As far as local visual queries are concerned, the images in which there is a blue region whose contour has a shape similar to a given curve `s` are denoted by the query $\exists$`HAR.(`$\exists$`HC.{blue}` $\sqcap$ `(`$\exists$`HS.`$\exists$`SS.{s}))`. Finally, the user interested in retrieving the images in which Mary plays Zerlina and wears a bluish dress, can use the query $\exists$`HR.`$\exists$`Rep.({mary}`$\sqcap\exists$`Plays.{zerlina})`$\sqcap$`(`$\exists$`HC.`$\exists$`SC.{blue})`.

6 Image bases and image retrieval

We define an *image base* as a 5-tuple $IB = \langle L, n_l, n_r, \Sigma_C, \Sigma_D \rangle$ where: (a) L is a finite set of layouts; (b) n_l is an injective *layout naming* function, mapping each

layout i in L into an individual $\mathtt{i}$, which therefore acts as a unique name for it. Note that, indirectly, n_l also associates i with the set of descriptions $\sigma(n_l(i)) = \{\delta_1, \ldots, \delta_n\}$, whose elements are the content descriptions of the image whose layout is i; (c) n_r is an injective *region naming* function, mapping each grounded region $\langle i, S\rangle$ of each layout i in L into an individual $\mathtt{r}$, which therefore acts as a unique name for it; (d) Σ_C is a finite set of content descriptions, such that each layout in L has at least one associated description (i.e. $\forall i \in L,\ |\sigma(n_l(i))| \geq 1$). "Uninterpreted" images will have a single content description containing just the image identification; (e) Σ_D is a set of fuzzy assertions representing domain knowledge.

Our image retrieval model is based on the idea that, in response to a query Q addressed to an image base $IB = \langle L, n_l, n_r, \Sigma_C, \Sigma_D\rangle$, the layout named $\mathtt{i}$ is attributed a degree of relevance n iff:

$$n = max_{\{\delta_j \in \sigma(\mathtt{i})\}}\{n_j = Maxdeg(\delta_j \cup \Sigma_D, Q(\mathtt{i}))\}$$

Let us consider an image base containing two layouts $\mathtt{i}$ and $\mathtt{j}$, such that:

$$\{\langle \mathtt{Ego(i)}, 1\rangle, \langle \mathtt{About(i,o)}, 0.8\rangle, \langle \mathtt{DonGiovanni(o)}, 1\rangle\}$$
$$\{\langle \mathtt{Ego(j)}, 1\rangle, \langle \mathtt{About(j,o)}, 0.7\rangle, \langle \mathtt{WestSideStory(o)}, 1\rangle\}$$

are in Σ_I. Moreover, Σ_C contains the following axioms:

$$\langle \mathtt{DonGiovanni} \sqsubseteq \mathtt{EuropeanOpera}, 1\rangle\ \langle \mathtt{WestSideStory} \sqsubseteq \mathtt{AmericanOpera}, 1\rangle$$
$$\langle \mathtt{EuropeanOpera} \sqsubseteq \mathtt{Opera} \sqcap (\exists \mathtt{ConductedBy.European}), 0.9\rangle$$
$$\langle \mathtt{AmericanOpera} \sqsubseteq \mathtt{Opera} \sqcap (\exists \mathtt{ConductedBy.European}), 0.8\rangle$$

Suppose we are interested in those images that are about an opera conducted by a European director. To this end, we can use the query $\exists \mathtt{About.}(\mathtt{Opera} \sqcap \exists \mathtt{ConductedBy.European})$. It can be verified that the degree of relevance attributed to $\mathtt{i}$ is 0.8, whereas that of $\mathtt{j}$ is 0.7.

We close with some implementation considerations. In order to effectively perform image retrieval as prescribed by the model defined so far, we envisage an IRS consisting of the following components: (1) a *global matching engine* for each global similarity predicate, responsible of implementing a specific kind of image global matching; to this end, each such engine will make use of the feature vectors for the layouts in the image base, stored in an apposite database, the *global matching database;* (2) a *local matching engine* for each local similarity predicate, using the feature vectors stored in *local matching databases,* of which there exists one for each considered image feature (colour, shape, etc.); (3) a *DL theorem prover,* which will handle the semantic information processing, collecting the assertions contained in the Σ_C and Σ_D components of the image base and appropriately using them in reasoning about image content; (4) a *query processor,* responsible of decomposing each query into abstract, concrete, and conceptual sub-queries, demanding the evaluation of each sub-query to the appropriate component, and then properly combining the results in order to obtain the final ranked list of images. For its operation, the query processor uses a database, called the *image structure database,* which stores the semantics

of selectors as well as naming functions. The details of these components are outside the scope of this paper. We only remark, at this point, that they are well within reach of the current technology. In particular, we have developed a theorem prover for a significant extension of the DL we use here [10].

7 Conclusions

We have presented an image data model providing a retrieval capability encompassing current similarity-based techniques and, in addition, making full and proper use of image semantics. Because the representations handled by the model have a clean semantics, further extensions to the model are possible. For instance, image retrieval by spatial similarity can be added to our model with moderate effort: at the form level, effective spatial similarity algorithms (e.g. [6]) can be embedded in the model via procedural attachment, while significant spatial relationships can be included in content descriptions by drawing from the many formalisms developed within the qualitative spatial reasoning research community [3]. Analogously, the model can be enhanced with the treatment of texture-based similarity retrieval.

References

1. F. Baader and P. Hanschke. A schema for integrating concrete domains into concept languages. In *Proceedings of IJCAI-91, Intern. Joint Conference on Artificial Intelligence*, pages 452–457, Sydney, 1991.
2. J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C.-F. Shu. The Virage image search engine: An open framework for image management. In *Storage and Retrieval for Still Image and Video Databases IV*, volume 2670 of *SPIE Proceedings*, pages 76–87, San Jose, CA, February 1996.
3. A. G. Cohn. Calculi for qualitative spatial reasoning. In *Proceedings of AISMC-3*, Lecture Notes in Computer Science. Springer Verlag, 1996.
4. D. Dubois and H. Prade. *Fuzzy Sets and Systems*. Academic Press, 1980.
5. C. Faloutsos, R. Barber, M. Flickner, J. Hafner, and W. Niblack. Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3:231–262, 1994.
6. V.N. Gudivada and V.V. Raghavan. Design and evaluation of algorithms for image retrieval by spatial similarity. *ACM Trans. on Inf. Sys.*, 13(2):115–144, April 1995.
7. V. N. Gudivada and V. V. Raghavan, editors. *IEEE Computer. Special Issue on Content-Based Image Retrieval*. IEEE, Sept. 1995.
8. E. J. Guglielmo and N. C. Rowe. Natural-language retrieval of images based on descriptive captions. *ACM Trans. on Inf. Sys.*, 14(3):237–267, July 1996.
9. H. V. Jagadish, A. O. Mendelson, and T. Milo. Similarity-based queries. In *Proc. of the 14th Symp. on Principles of Database Systems*, San Jose, May 1995.
10. C. Meghini and U. Straccia. A relevance terminological logic for information retrieval. In *Proc. of SIGIR-96, the 19th ACM Int. Conf. on Research and Development in Information Retrieval*, Zurich, August 1996.
11. A. Rosenfeld and A. C. Kak. *Digital picture processing*. Academic Press, 1982.
12. M. Schmidt-Schauß and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48:1–26, 1991.

13. A. F. Smeaton and I. Quigley. Experiments on using semantic distances between words in image caption retrieval. In *Proc. of SIGIR96, the 19th ACM Int. Conf. on Research and Development in Information Retrieval*, Zurich, CH, August 1996.
14. C.J. van Rijsbergen. *Information Retrieval*. Butterworths, London, GB, 1979.

This article was processed using the LaTeX macro package with LLNCS style