# SAFIRE: Towards Standardized Semantic Rich Image Annotation

Christian Hentschel, Andreas Nürnberger, Ingo Schmitt, and Sebastian Stober

Faculty of Computer Science
Otto-von-Guericke-University Magdeburg, D-39106 Magdeburg, Germany
chentsch@student.uni-magdeburg.de
{nuernb,stober}@iws.cs.uni-magdeburg.de
Ingo.Schmitt@iti.cs.uni-magdeburg.de

**Abstract.** Most of the currently existing image retrieval systems make use of either low-level features or semantic (textual) annotations. A combined usage during annotation and retrieval is rarely attempted. In this paper, we propose a standardized annotation framework that integrates semantic and feature based information about the content of images. The presented approach is based on the MPEG-7 standard with some minor extensions. The proposed annotation system SAFIRE (Semantic Annotation Framework for Image REtrieval) enables the combined use of low-level features and annotations that can be assigned to arbitrary hierarchically organized image segments. Besides the framework itself, we discuss query formalisms required for this unified retrieval approach.

## 1 Introduction

Due to the vast amounts of images that are digitally available today, the development of advanced techniques for storing, structuring and especially for efficiently retrieving images are required. The image retrieval process entails several specific problems, e.g. the extraction of relevant and descriptive features, the problem of computing the similarity between images or images and a user query that require flexible and adaptive similarity measures, and the problem of designing interactive user interfaces that provide besides basic query support also visualization techniques for the retrieved set of images [16, 22]. Unfortunately, several aspects of this process are still insufficiently studied. This involves problems of automatically extracting descriptive features or segmenting images into descriptive regions, but also the lack of methods to appropriately analyze and process user queries. Thus image retrieval is still a very time consuming task, since usually several search steps are necessary until a desired image is found.

The vision that most research currently follows is that of a system where a user can provide a natural language query, in which the desired content is described, in order to retrieve the searched images. However, the main problem in this setting is the semantic gap between the user need on one hand and the features that we can currently extract automatically from images on the other hand: Unfortunately, it is not yet possible to extract reliably and automatically a semantic content description of an image except of one of a very restricted image collection. In order to circumvent this problem some research projects focus on the problem of extracting more descriptive features in order to allow more reliable image comparison within interactive retrieval systems. Others try to design systems that enable semantic annotation of images. Unfortunately, only a few projects try to make use of a standardized way to merge research results from both sides in order to iteratively bridge the gap between both approaches.

In this paper, we propose a standardized annotation framework that integrates semantic and feature based information about the content of images. The presented approach is based on the MPEG-7 standard with some minor extensions. Therefore, parts of the information stored can be used or maintained by a great number of tools. It enables the combined use of low-level features and annotations that can be assigned to arbitrary hierarchically organized image segments.

In the following, we first discuss briefly related work in order to motivate our approach. In Sect. 3 we provide an overview of the developed framework. In Sect. 4 we discuss the used MPEG-7 extensions and propose in Sect. 5 a refined model for semantic querying. Finally, we give in Sect. 6 a brief overview of our prototype and the annotation process.

## 2 Related Work

In order to narrow the semantic gap between linguistic user queries and image collections different strategies can be applied. The two orthogonal strategies are to use only (low level) features extracted from the image itself or to use only textual annotations and to ignore completely information that could be extracted from the image. The latter approaches use, e.g., if an image is stored in a web page or in an electronic document, only the surrounding text and captions (see, for example, Google Image Search). The former approaches usually make use of highly interactive and adaptive retrieval systems that require as starting point either a sample image or provide initially an overview of the available image collection [2, 31]. This is done either by randomly selecting images or by structuring the collection and representing a prototypical image for each discovered cluster [25]. Starting from this overview, the user has to iteratively navigate through the collection in order to retrieve the searched images.

Meanwhile, several approaches for semantic annotation of image collections have been proposed. The main idea is to support the annotation of images with free text or keywords in order to enable structuring of the collection itself and to support a more 'semantic' retrieval of images in huge collections. Some of the proposed tools even make use of ontologies in order to enable an unambiguous keyword annotation, see e.g. [3, 13]. One main problem of these tools is that text and descriptive keywords have to be assigned manually. Therefore, motivated by the success of community based portals, recently several annotation or so-called tagging platforms like Flickr and Marvel[1] have been developed that

[1] see http://www.flickr.com and http://www.research.ibm.com/marvel/

allow users to freely upload and annotate images. The idea is to make use of the self-organization process of huge communities in order to structure and to annotate image collections as well as to support this process – in case of Marvel – by means of machine learning techniques. We can also find approaches that try to make use of partially annotated image collections or a small set of sample queries and images. By training a classifier, unseen images can be automatically labelled (supervised or unsupervised) by propagating annotations of sample images (training samples) to the remaining or newly added ones. The probability for a single keyword belonging to a specific image is estimated based on the distances and density of the training samples in the applied model space [21, 17, 10].

Furthermore, much relevant work deals with the problem of modelling vagueness in the retrieval process. At the beginning of the nineties, techniques based on fuzzy logic [34] were applied to traditional database technology in order to cope with vague or uncertain information, which is especially important if we try to combine (low level) image features with textual annotation. The capability to deal with vagueness is even fundamental if we like to process natural language queries [34]. An overview of recent work in the area of databases is given in [11]. Unfortunately, the problem of query processing, which includes information aggregation, similarity measures and ranking, is still frequently underestimated, see e.g., [28, 4, 24]. Therefore, we discuss this problem in more detail in Sect. 5.

One problem of many approaches mentioned above is that they still consider an image as an integral entity, i.e. all annotations refer to the image as a whole and not to individual parts in it. Even though, some methods for automatic image segmentation in retrieval systems have been proposed [6, 23], they still rely on low level image features.

One further problem of most approaches mentioned above is the use of very specific annotation formats. This makes the exchange of annotations and the development of benchmark collections and learning methods for automatic annotation very difficult. However, several standards for storing image metadata are available: Dublin Core[2] is a metadata scheme that can be used to describe documents or objects on the internet. The Exchangeable Image File Format (EXIF)[3] specifies standards related to digital still cameras. EXIF metadata are commonly included in JPEG images. However, both Dublin Core and EXIF provide only limited support for annotations and do not allow to specify and annotate regions within images. Standards that do provide this are FotoNotes™[4], JPEG-2000 metadata[5] and MPEG-7[6]. All these standards are based on XML. The former two are bound to specific image formats, JPEG and JPEG-2000 respectively. Only the latter is independent from the format of the medium being described and stored separately.

[2] The Dublin Core Metadata Initiative (DCMI), http://dublincore.org

[3] http://www.exif.org

[4] http://fotonotes.net

[5] Joint Photographic Experts Group, http://www.jpeg.org/jpeg2000/metadata.html

[6] Moving Picture Experts Group (MPEG), http://www.chiariglione.org/mpeg/

## 3 Requirements of a Standardized Annotation Approach

A framework that enables a standardized way to create, store, and maintain semantic rich annotated images has to follow certain rules and has to provide a basic set of features. It must be possible to store information about image segments together with their spatial and hierarchical relation, i.e. segments may be grouped in arbitrary hierarchies. Furthermore, it has to provide means for storing low and high level features as well as textual (semantic) annotations for each segment, group and the image itself. A further fundamental requirement is that all information – from low level features to high level semantic annotations – have to be stored in a data structure that can be easily modified, is well documented and possibly already supported by existing software tools. Therefore, we decided to make use of the MPEG-7 standard with some minor extensions we regard to be necessary. Thus, users can already use a wide variety of tools to maintain and access collections of annotated images. Since MPEG-7 was designed for describing multimedia data in general, a further advantage is, that image annotations can be easily embedded into video sequences as well.

Currently, an automatic assignment of high-level semantics to low-level features of an arbitrary image is still impossible. There exist several approaches, which lead to considerably good results – however they are restricted to a specific domain. By including domain knowledge in form of objects to be detected as well as their low-level visual features and spatial resolution, Voisine et al. [32], for example, accomplish a semantic interpretation of Formula One and Tennis video sequences. In [15] an approach is presented to prove, that again domain specific cues for segmentation can be learned from pre-clustered image sets in order to gain high classification rates. Since we did not want to be a priori restricted to a specific image domain, the annotation process we present requires user interaction to introduce the required model information. In order to support the user during the annotation task itself, an annotation system should meet the following requirements:

- provide initial automatic segmentation and user interaction mechanisms in order to minimize the required amount of user actions,
- allow to create, delete and modify semantic regions,
- allow to structure segments into more general region groups to create a region hierarchy (i.e. atomic regions as created by the user or an automated process can be grouped into semantic units which again can be grouped likewise),
- offer methods to automatically compute low and if possible higher level features for all levels of the segment hierarchy,
- provide methods for (semantic) annotations on every level of the region hierarchy,
- support methods to automatically propose (semantic) annotations based on existing annotations,
- provide capability to create different views on the same image (e.g. create different atomic regions and different region hierarchies),
- ensures a semantic unambiguous annotation, e.g. by including references to unique items of an ontology.

Providing the possibility to link ambiguous terms or phrases to unique entries in an ontology (already during the process of annotation) has several advantages: Firstly, it avoids ambiguous annotations. For example, the noun "bank" has about 10 different senses according to the WordNet ontology [20]. Among them, only one meaning is appropriate in a given context. Someone searching for an image showing a bank in the sense of a financial institution should not receive an image describing a river bank as a search result. Secondly, apart from disambiguation of annotations, the linkage to unambiguous entries (WordNet SynSets) allows for an automated extension of image annotations e.g. by synonyms. Utilizing the EuroWordNet InterLingual Index [33] even makes it possible to perform multilingual searches.

Finally, by using synonyms and homonyms as defined in an ontology for a specific image keyword, the content description of an image can be improved by providing a more universally applicable semantic annotation. This can help to augment the relevance of the retrieval results.

The image annotation system protoype presented in this paper fulfills many of the requirements mentioned above. It exploits the MPEG-7 standard for data storage and offers an intuitive user interface for semi-automatic image annotation. By providing the possibility to link textual queries and annotations to the WordNet ontology, word disambiguation becomes feasible. The underlying system architecture is presented in Fig. 1.

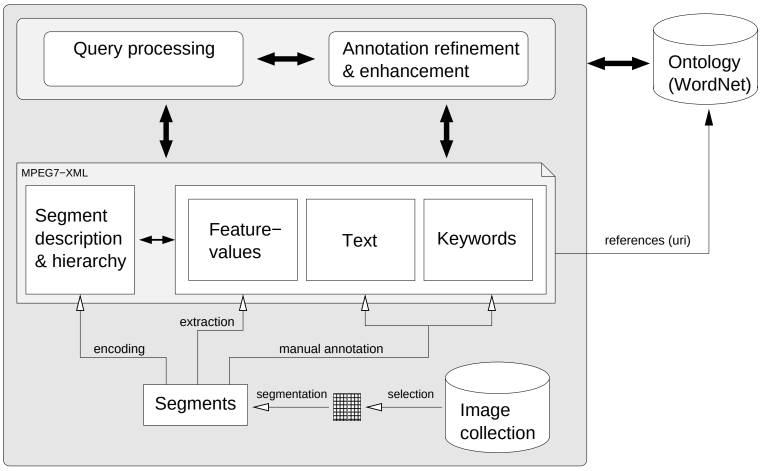


**Fig. 1.** Structure of the System Architecture

## 4 Image Description

Information about the segments of an image, their automatically extracted low level features and their manual annotations are stored in an MPEG-7 conform XML data format. MPEG-7 is based on an XML schema and provides a very flexible framework for describing audiovisual data that can be easily extended. Furthermore, the use of this common standard eases data exchange with other applications.

Regarding the storage of image segmentation information, our approach makes use of the `StillRegion` and `StillRegionSpatialDecomposition` description scheme (DS) as defined in MPEG-7.[7] The `StillRegion` DS represents an image or a part of it. Several visual descriptors for automatically extractable low level features such as shape, color or statistical texture measures of such a `StillRegion` are already defined in the MPEG-7 standard. In particular we (intend to) make use of the `DominantColor` and `ScalableColor` Description Tools and intend to exploit the Homogeneous Texture Descriptor defined by MPEG-7. Furthermore, MPEG-7 provides means to model spatial as well as semantic relations as e.g. used in [18]. The `StillRegionSpatialDecomposition` DS allows to decompose a `StillRegion` according to some criteria. Such a decomposition comprises a set of still regions (or references to still regions) which again may be decomposed thus allowing hierarchical decompositions. As the number of decompositions for each still region is unbounded, it is even possible to store any arbitrary number of hierarchical decompositions for an image by solely using `StillRegion` and `StillRegionSpatialDecomposition` DS. This method however would create an unnecessary overhead of redundant data if more than one hierarchical decomposition has to be stored because of the following reason: The preferred way to incorporate a still region that is already contained in a different hierarchical decomposition would obviously be to use a reference to the already stored data instead of creating a copy. Referencing a still region would however mean to implicitly reference all decompositions associated with this region, as well. But these associated decompositions will usually differ, thus making it necessary to store duplicates of the same still region with different decompositions but identical features that ought to be stored only once. To circumvent this data overhead, we defined the custom visual descriptor shown in Fig. 2.

An `HierarchicalSegmentationDescriptor` comprises a so called “flat decomposition” and an arbitrary number of `HierarchicalSegmentations`. A flat decomposition is a `StillRegionSpatialDecomposition` containing all still regions of all `HierarchicalSegmentations` defined subsequently. The `HierarchicalSegmentations` in turn solely contain references to the still regions of the flat decomposition or (references) to sub-segmentations. This way, it is ensured that each still region is defined only once. The flat decomposition may be associated with the still region corresponding to the whole image instead of being stored directly in the custom visual descriptor. In this case, it is referenced by `FlatSegmentationRef`. The advantage of this approach is, that information of the flat decomposition is accessible for tools that can only process pure MPEG-7 annotations and ignore the content of the custom visual descriptor. A simple example for an `HierarchicalSegmentationDescriptor` is given in Fig. 2.

[7] ISO/IEC 15938-5:2003, available at http://www.iso.org

```
<complexType name="HierarchicalSegmentationDescriptorType">
  <complexContent>
    <extension base="mpeg7:VisualDType">
      <sequence>
        <choice minOccurs="1" maxOccurs="1">
          <element name="FlatSegmentation" type="mpeg7:StillRegionSpatialDecompositionType"
            minOccurs="0" maxOccurs="1"/>
          <element name="FlatSegmentationRef" type="mpeg7:ReferenceType" minOccurs="0"
            maxOccurs="1"/>
        </choice>
        <element name="HierarchicalSegmentation" type="HierarchicalSegmentationType"
          minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
<complexType name="HierarchicalSegmentationType">
  <complexContent>
    <extension base="mpeg7:SpatialSegmentDecompositionType">
      <sequence>
        <element name="StillRegionRef" type="mpeg7:ReferenceType" minOccurs="0"
          maxOccurs="1"/>
        <choice minOccurs="0" maxOccurs="unbounded">
          <element name="Semantic" type="mpeg7:SemanticType"/>
          <element name="SemanticRef" type="mpeg7:ReferenceType"/>
        </choice>
        <choice minOccurs="1" maxOccurs="unbounded">
          <element name="SubStillRegionRef" type="mpeg7:ReferenceType"/>
          <element name="SubSegmentation" type="HierarchicalSegmentationType"/>
          <element name="SubSegmentationRef" type="mpeg7:ReferenceType"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<!--    example:    -->
<VisualDescriptor xsi:type="HierarchicalSegmentationDescriptorType">
  <FlatSegmentationRef idref="flatDecomposition" />
  <HierarchicalSegmentation>
    <StillRegionRef idref="imageRegion" />
    <SubStillRegionRef idref="skyRegion" />
    <SubStillRegionRef idref="roadRegion" />
    <SubSegmentation>
      <StillRegionRef idref="carRegion" />
        <SubStillRegionRef idref="windscreenRegion" />
        <SubStillRegionRef idref="carDoorRegion" />
      </SubSegmentation>
    <SubSegmentation>
  </HierarchicalSegmentation>
</VisualDescriptor>
```

**Fig. 2.** Top: Custom description scheme modeling hierarchical decompositions. Bottom: A simple example for the `HierarchicalSegmentationDescriptor` defined above. The still region representing the whole image has 3 subregions for the sky, a road and a car. The region of the car has subregions for the windscreen and a door of the car.

For the annotation of still regions, the `Linguistic` Description Scheme [12] of the version 2 schema definition of MPEG-7[8] is used. This DS is based on the GDA tag set[9] and provides means to annotate linguistic data associated with multimedia content. Its descriptive power is much more comprehensive compared to the `TextAnnotation` datatype included in the first release of the MPEG-7 standard[10]. We currently only use the `Linguistic` DS to enrich image annotations with references to external resources such as ontologies. However, far more sophisticated extensions are imaginable such as those described in [12]. Using the `Linguistic` DS for image annotations, any single term or group of terms can be linked to several external resources by specifying corresponding URIs in the `semantics` attribute of an encapsulating `LinguisticEntityType` element, e.g. `<Phrase>` or `<Sentence>`. An example annotations is shown in Fig. 3.

```
<Sentence xml:lang="en" id="annotation_1">
  <Phrase id="annotation_1.1">
    My
    <Phrase semantics="WordNet:SynSetID=2471824 EuroWordNet:ILISynSetID=8542395">
      brother
    </Phrase>
  </Phrase>
  at
  <Phrase id="annotation_1.2">
    the
    <Phrase semantics="WordNet:SynSetID=86786241 EuroWordNet:ILISynSetID=1332468">
      bank
    </Phrase>
  of the Thames
  </Phrase>
</Sentence>
```

**Fig. 3.** Image annotation "My brother at the bank of the Thames." where the terms "brother" and "bank" are linked to SynSets in WordNet and EuroWordNet InterLingual Index (ILI) [33].

Annotations in turn can be assigned to still regions by using the `SemanticRef` element defined in the `StillRegion` DS. A `SemanticRef` may point to any part of the annotations contained in the `Linguistic` DS decribed above making it possible e.g. to link segments of an image to specific phrases of a sentence that describes the images as one. Recalling the example annotation in Fig. 3, the phrases "my brother" and "bank of the Thames" could be assigned to segments of the image as shown in Fig. 4.

[8] ISO/IEC 15938-10:2005: Information technology - Multimedia content description Interface – Part 10: Schema definition, available at http://www.iso.org

[9] http://i-content.org/GDA/tagset.html

[10] Refer to ISO/IEC 15938-5:2003 available at http://www.iso.org for specifications or [19] for an overview on MPEG-7 description tools that also cover textual annotation.

```
<image>
  <SemanticRef idref="annotation_1" />          <!-- sentence annotating the whole image -->
  ...
  <SpatialDecomposition overlap="true" gap="true" criteria="flat decomposition of the image"
    id="flatDecomposition">
    <StillRegion id="region_1">
      <SemanticRef idref="annotation_1.1" />    <!-- annotation: my brother -->
      <SpatialLocator>...</SpatialLocator>
    </StillRegion>
    <StillRegion id="region_2">
      <SemanticRef idref="annotation_1.2" />    <!-- annotation: the bank of the Thames -->
      <SpatialLocator>...</SpatialLocator>
    </StillRegion>
  </SpatialDecomposition>
</image>
```

**Fig. 4.** Example for assigning annotations to still regions. The referenced annotations are shown in Fig. 3.

## 5 Searching for Images

Our annotation framework is designed to create and to provide a full range of data describing the content of images. That data include automatically extracted low-level features, user-defined structured data as well as annotations like keywords and textual descriptions with references to an ontology. They are assigned to image segments organized in segmentation hierarchies. Additionally, segments can stand in mutual spatial relationships.

There are various paradigms of searching images based on different types of data:

1. *text retrieval* on textual descriptions and keywords;
2. *navigation* through the image collection by means of a highly interactive user interface and clusters pre-computed from low-level features;
3. *content-based retrieval* based on query images, low-level features and an appropriate similarity measure; and
4. *database query* on spatial relationships, segment descriptions and user-defined data.

There has been a huge amount of research done on these individual search paradigms. Each of them comes with its own limitations and none of them can be seen as the best search paradigm. However, only few attention is paid to the *combination* of them into one unifying query system. Combining various search paradigms requires a sophisticated query language which enables the user to formulate queries that are possibly composed of different query conditions. The main problem is therefore, how to combine query conditions from different paradigms into one unifying formalism. Optimally, a query system is capable of processing natural language queries. In order to be as close as possible to that vision we decided to take advantage of formal logic as basic formalism.

### Querying Using Logic-based Approaches

First order logic is the main concept of database query languages like SQL and XQuery. Unfortunately, they do not adequately offer concepts needed for processing queries which combine retrieval and traditional database search conditions.

For example, the keyword query `keyword = ‘rock’` as a typical database query returns a set of images for which that condition holds. Contrarily, the query `image is visually similar to a given query image` is a content-based retrieval query returning a list of images sorted in descending order by their respective similarity scores. Assume, we want to conjunctively combine both queries into one query:

`keyword = ‘rock’  AND image ≈ query image`

What would be the result, a list or a set of images? The problem here is the illegal logical combination of an exact query providing us boolean values with an imprecise retrieval query returning similarity scores from the interval $[0, 1]$. There are two prominent approaches to circumvent that conflict:

*Boolean Query:* The idea realized in most logic-based query systems like in commercial database systems is to transform the retrieval query into a boolean one. This is achieved by applying a threshold value. That is, every similarity score greater than the threshold is considered true otherwise false. This approach has several drawbacks. First, finding a suitable threshold value is not an easy task. Second, as result, we lose the information of what degree the similarity condition holds. Thus, we cannot discriminate among images from the result set w.r.t. their similarity to the query image. Especially in queries composed of several conditions we need that lost semantics.

*Retrieval Query:* The idea here is to transform the database query into a kind of retrieval query. That is, logic values from the database query evaluation are mapped to the score values 1 for `true` and 0 otherwise. These scores can then be arithmetically combined with the scores from a retrieval query, e.g. by a simple weighted sum. However, it is not clear at all which combination formula should be applied for a specific query. There is a plethora of possible aggregation formulas for that scenario. Furthermore, there is no logic framework (conjunction, disjunction, negation) supporting the formulation of complex queries. That is, we cannot utilize the rich theory of database querying.

Summarizing, the first approach lacks support for similarity scores whereas the second one fails with respect to an available logic for query formulation and processing.

A straightforward solution to the problem is to take advantage of fuzzy logic [9] as proposed, for example, in [34]. In fuzzy logic, similarity scores as well as boolean truth values are interpreted as fuzzy set membership values which can be combined via logical junctors to construct complex queries. Scoring functions t-norm and t-conorm behave like the logical conjunction and disjunction, respectively. Examples of query languages based on fuzzy logic are the `same` algebra [8], WS-QBE, $\mathcal{SDC}$, and $\mathcal{SA}$ as proposed in [27, 28]. Fagin’s weighting schema [9] is used in those languages in order to equip search conditions with different weights of importance. For example, matching a keyword condition should be of more weight than a visual similarity condition. Bellmann and Giertz [1] proved that fuzzy logic with t-norm `min` for conjunction and t-conorm `max` for disjunction obeys the rules of the boolean algebra. Thus, most query processing techniques known from the database theory are still valid.

Nevertheless, there are some common problems of fuzzy-based querying. First, applying the standard fuzzy norms `min/max` in our context suffers from a specific property: The minimum as well as the maximum of two certain scores returns always just one of them and ignores completely the other one. For example, assume two conjunctively combined retrieval conditions. The condition which returns smaller scores dominates completely the result semantics. Contrarily, a *non-dominating* t-norm which respects both scores simultaneously would better meet our understanding of query combination. Actually, fuzzy logic comes with different non-dominating t-norms, e.g. the algebraic product. Unfortunately, none of them holds idempotence. Thus, in combination with a t-conorm, e.g. the algebraic sum, distributivity cannot be guaranteed. Furthermore, we are faced with problems of failing associativity and distributivity [29] when Fagin's weighting schema is used, even on the `min/max` pair. Table 1 summarizes the properties of the approaches discussed so far.

| approach | scores | distributivity | non-dominating |
|---|---|---|---|
| boolean query | no | yes | — |
| retrieval query | yes | — | — |
| fuzzy logic (`min/max`) | yes | yes | no |
| fuzzy logic (not `min/max`) | yes | no | yes |
| weighted fuzzy logic | yes | no | — |

**Table 1.** Properties of different approaches to combine retrieval and database queries

The problem of dominance turns out to be even more serious when we examine the way how fuzzy logic is utilized for query evaluation. As shown in Fig. 5 (left) fuzzy logic relies on *importing* scores (here two score functions) and truth values (here from one database condition) and interpreting them as membership values. Thus, the generation of membership values is *not under control* of fuzzy logic. Therefore, there is a high risk that scores are not comparable due to possibly different score functions producing an error-prone dominance. Figure 5 (right) depicts exemplarily two fictive non-comparable score functions. If the scores from both function are combined by the `min`-function, the scores from function A would predominate the ones from function B. Furthermore, assume `a` and `b` are different perceived similarity values of one image based on texture and color histogram, respectively. Using non-comparable score functions can even change the order of different scores (`b < a` but `a' < b'`) making the distinction between conjunction and disjunction meaningless.

Despite of the problems discussed above, we use for our framework the query language WS-QBE since (1) it provides a user-friendly QBE-interface for query formulation, (2) it is especially designed to support multimedia queries, and (3) its implementation and source code is available and can therefore be easily adapted to specific needs. However, our long-term goal is to find a formalism unifying the generation of similarity scores, classical database evaluations as well as their combination via a logic. One promising approach in that direction is the usage of quantum mechanics and quantum logics. Since their underlying model is a vector space and many retrieval problems can be formulated in vector space there is a natural mapping into quantum mechanics [26].

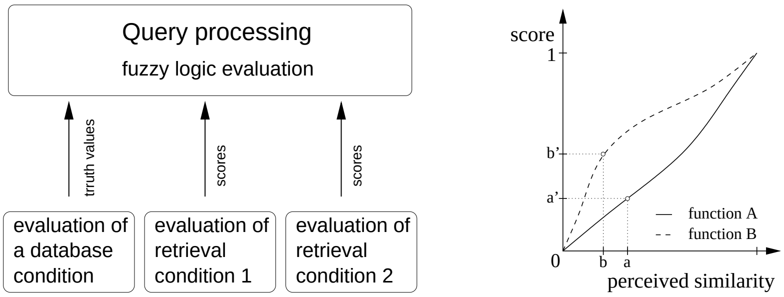
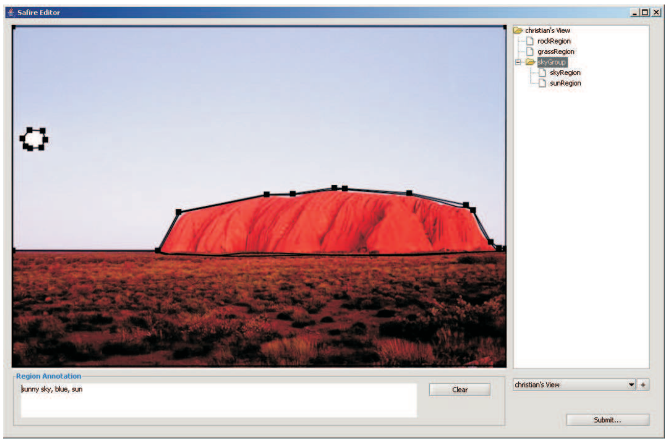


**Fig. 5.** Fuzzy evaluation by importing truth and score values (left) and score values from two different score functions (right)

## 6 An Image Annotation Prototype

Based on the requirements defined in Sect. 3 we developed a first prototype to support region based image annotation and retrieval: SAFIRE (Semantic Annotation Framework for Image REtrieval). SAFIRE implements an intuitive user interface to attach automatically extracted low-level features as well as semantic meta data to an image (see Figure 6).

As one of our main goals was to implement an approach that enables the user to annotate images on a segment level, we first of all needed to establish a way to identify regions in an image. Unfortunately, due to missing segmentation algorithms that provide appropriate segmentation of arbitrary images into



sunny sky, blue, sun

**Fig. 6.** Screenshot of annotation interface

meaningful regions, annotating images on a region level currently cannot exclusively rely on automated image segmentation. However, sophisticated algorithms exist, that can help to provide an adequate initial segmantation which is less tedious to be adapted than it would be when a segmentation of an image was to be created from scratch. Currently we apply here a simple k-means clustering on the pixel color data [14] which provides rather poor results. One of the very next steps in improving SAFIRE's performance will be to replace this algorithm with a more sophisticated one, such as the one used in the Blobworld system [7]. The results of this initial segmentation step are presented to the user. By using a canvas-like interface, the user can refine existing regions as well as create new and delete invalid ones. For each newly created region, the system automatically computes low level features - namely the shape, the color distribution as well as texture measures. These features are automatically stored in MPEG-7 conform documents – as described in Sect. 4 – and can be used in a query process to determine similarity between different regions of different images.

In a subsequent step, newly added or existing regions can be enriched by adding annotations describing the semantics of a region. The user can select regions through the annotation tool and attach keyword lists to each of them. As the image as a whole can be seen as a region as well, a global image annotation can be attached likewise. Typically, an image can be split into atomic regions representing the smallest semantic entities. These entities can be grouped into superordinate regions depicting a more abstract view on an image. In general, any image can be split into groups of semantic regions which likewise can be split again until the atomic level is reached. Our annotation framework supports this idea by enabling the user to group newly created regions into sematic units of higher abstraction. Hence, a semantic hierarchy can be created for every image, represented by a tree-like view (see Fig. 6). Each annotation is strictly related to a specific level in the hierarchy and hence represents a specific level of abstraction. Consequently, superordinate regions do not know about their children, as they represent a different semantic concept.

What is identified as a semantic entity of an image is strongly related to the focus of the beholder. Two different people may examine an image from two completely different perspectives and hence come to two completely different semantic hierarchies. For example a car manufacturer would 'disassemble' a picture of a car on a street into all its visible components such as doors, windows and wheels in order to annotate them with the appropriate component identifier. On the other hand, a less technically motivated beholder of the same image would probably show no interest in the cars components but rather segment the car as a whole annotating it with more abstract keywords such as 'cabriolet, red'. Both these segmentations are valid and should be supported by the system as equally correct. SAFIRE addresses this aspect by enabling different users to create different segment group trees for a single image. The underlying atomic entities are shared among the views. How they get grouped into regions of higher abstraction, however, is an individual decision. Like atomic regions, region groups can be annotated just as well. An overall view of the framework presenting the mentioned annotation steps is given in Figure 7.

As a next step, we will integrate the image retrieval component described above. Further on, more sophisticated automatic image segmentation algorithms, see e.g. [30, 5], will be analyzed and added to the framework.

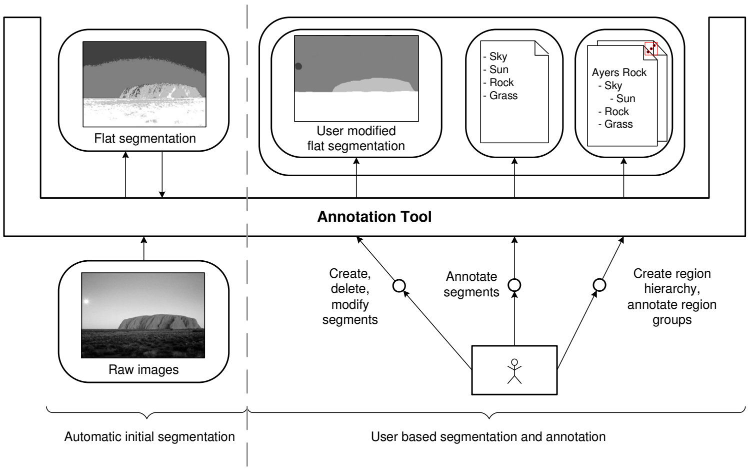

Flat segmentation
User modified flat segmentation
- Sky
- Sun
- Rock
- Grass
Ayers Rock
- Sky
- Sun
- Rock
- Grass
Annotation Tool
Raw images
Create, delete, modify segments
Annotate segments
Create region hierarchy, annotate region groups
Automatic initial segmentation
User based segmentation and annotation

**Fig. 7.** Segmenting and annotating images in SAFIRE

## 7 Conclusions and Future Work

In this paper we have proposed an annotation structure based on the MPEG-7 standard that allows us to store information about image segments together with their low and higher level features. In order to retrieve images using these annotations, we presented a logic-based query method that supports combined queries on numerical features and text based annotations. Furthermore, we presented the SAFIRE system that supports a user during the annotation process. It enables the user to cluster images into semantic entities on different levels of abstraction. As different beholders of an image might have a different focus on its contents, different semantic views on one image are supported. Each level of abstraction offers the ability to attach keyword lists to explicitly store the depicted content of a region. In addition to semantic annotations, for each atomic region a number of automatically computed low-level features is stored.

Our long-term goal is to develop a data collection containing features as well as annotations freely available on the web. We hope that annotated MPEG-7 files can serve to initiate the creation of a data archive for the development of new search and learning mechanism as well as a reference data set for evaluations like the TRECVID dataset is for video annotations. Our current work is focused on the evaluation of methods using annotated image segments to infer annotations for new and unseen images. This could be done by matching semantically described segments from our database with new images.

## References

1. R. Bellman and M. Giertz. On the Analytic Formalism of the Theory of Fuzzy Sets. *Information Science*, 5:149–156, 1973.
2. A. D. Bimbo. *Visual Information Retrieval*. Morgan Kaufmann, 1999.
3. S. Bloehdorn, K. Petridis, C. Saathoff, N. Simou, V. Tzouvaras, Y. Avrithis, S. Handschuh, Y. Kompatsiaris, S. Staab, and M. G. Strintzis. Semantic annotation of images and videos for multimedia analysis. In *Proc. of Second European Semantic Web Conf. (ESWC 2005)*, 2005.
4. M. Boughanem, Y. Loiseau, and H. Prade. Rank-ordering documents according to their relevance in information retrieval using refinements of ordered-weighted aggregations. In *Adaptive Multimedia Retrieval: User, Context, and Feedback, Postproc. of 3rd Int. Workshop*, pages 44–54. Springer-Verlag, 2006.
5. N. V. Boulgouris, I. Kompatsiaris, V. Mezaris, D. Simitopoulos, and M. G. Strintzis. Segmentation and content-based watermarking for color image and image region indexing and retrieval. In *EURASIP Journal on Applied Signal Processing*, pages 418–431. Hindawi Publishing Corporation, 2002.
6. C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, 2002.
7. C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems*. Springer, 1999.
8. P. Ciaccia, D. Montesi, W. Penzo, and A. Trombetta. Imprecision and user preferences in multimedia queries: A generic algebraic approach. In *Proc. of FoIKS: Foundations of Information and Knowledge Systems*, pages 50–71. Springer, 2000.
9. R. Fagin. Fuzzy Queries in Multimedia Database Systems. In *Proc. of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 1-3, 1998, Seattle, Washington*, pages 1–10. ACM Press, 1998.
10. H. Feng and T.-S. Chua. A bootstrapping approach to annotating large image collection. In *MIR '03: Proc. of the 5th ACM SIGMM Int. Workshop on Multimedia Information Retrieval*, pages 55–62, New York, NY, USA, 2003. ACM Press.
11. J. Galindo, A. Urrutia, and M. Piattini. *Fuzzy Databases: Modeling, Design and Implementation*. Idea Group Publishing, 2005.
12. K. Hasida. The linguistic DS: Linguisitic description in MPEG-7. *The Computing Research Repository (CoRR)*, cs.CL/0307044, 2003.
13. L. Hollink, G. Schreiber, J. Wielemaker, and B. Wielinga. Semantic annotation of image collections. In *In Proc. of Workshop on Knowledge Markup and Semantic Annotation (KCAP'03)*, 2003.
14. A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, Inc., New Jersey, 1988.
15. S. Konishi and A. Yuille. Statistical cues for domain specific image segmentation withperformance analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 125–132, 2000.
16. S. Kosinov and S. Marchand-Maillet. Overview of approaches to semantic augmentation of multimedia databases for efficient access and content retrieval. In *Adaptive Multimedia Retrieval, Postproc. of 1st Int. Workshop*, pages 19–35, 2004.
17. J. Lu, S. ping Ma, and M. Zhang. Automatic image annotation based-on model space. In *Proc. of IEEE Int. Conf. on Natural Language Processing and Knowledge Engineering*, pages 455 – 460, 2005.

18. M. Lux, J. Becker, and H. Krottmaier. Caliph & Emir: Semantic annotation and retrieval in personal digital photo libraries. In *Proc. of CAiSE 03 Forum at 15th Conf. on Advanced Information Systems Engineering*, pages 85–89, 2003.
19. J. M. Martínez. MPEG-7: Overview of MPEG-7 description tools, part 2. *IEEE MultiMedia*, 9(3):83–93, 2002.
20. G. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Five papers on WordNet. *Int. Journal of Lexicography*, 3(4), 1990.
21. A. P. Natsev, M. R. Naphade, and J. Tesic. Learning the Semantics of Multimedia Queries and Concepts from a Small Number of Examples. In A. Press, editor, *Proc. of the 13th ACM Int. Conf. on Multimedia*, pages 598–607, 2005.
22. A. Nürnberger and M. Detyniecki. Adaptive multimedia retrieval: From data to user interaction. In *Do smart adaptive systems exist? - Best practice for selection and combination of intelligent methods*. Springer-Verlag, 2005.
23. J.-F. Omhover and M. Detyniecki. Strict: An image retrieval platform for queries based on regional content. In *Proc. of Int. Conf. on Image and Video Retrieval (CIVR 2004)*, 2004.
24. J.-F. Omhover, M. Rifqi, and M. Detyniecki. Ranking invariance based on similarity measures in document retrieval. In *Adaptive Multimedia Retrieval: User, Context, and Feedback, Postproc. of 3rd Int. Workshop*, pages 55–64. Springer, 2006.
25. S. Rüger. Putting the user in the loop: Visual resource discovery. In *Adaptive Multimedia Retrieval: User, Context, and Feedback, Postproc. of 3rd Int. Workshop*, pages 1–18. Springer-Verlag, 2006.
26. I. Schmitt. Basic Concepts for Unifying Queries of Database and Retrieval Systems. Technical Report 7, Fakultät für Informatik, Univ. Magdeburg, 2005.
27. I. Schmitt and N. Schulz. Similarity Relational Calculus and its Reduction to a Similarity Algebra. In *Proc. of 3rd Intern. Symposium on Foundations of Information and Knowledge Systems (FoIKS'04)*, pages 252–272. Springer-Verlag, 2004.
28. I. Schmitt, N. Schulz, and T. Herstel. WS-QBE: A QBE-like Query Language for Complex Multimedia Queries. In *Proc. of the 11th Int. Multimedia Modelling Conf. (MMM'05)*, pages 222–229. IEEE CS Press, 2005.
29. N. Schulz and I. Schmitt. A Survey of Weighted Scoring Rules in Multimedia Database Systems. Preprint 7, Fakultät für Informatik, Univ. Magdeburg, 2002.
30. C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 246–252, 1999.
31. R. C. Veltkamp and M. Tanase. Content-based image retrieval systems: A survey. Technical Report UU-CS-2000-34, CS Dept., Utrecht University, 2000.
32. N. Voisine, S. Dasiopoulou, F. Precioso, V. Mezaris, I. Kompatsiaris, and M. Strintzis. A genetic algorithm-based approach to knowledge-assisted video analysis. In *IEEE International Conference on Image Processing*, 2005.
33. P. Vossen. EuroWordNet general document version 3, final, July 19 1999.
34. L. A. Zadeh. Fuzzy Logic. *IEEE Computer*, 21(4):83–93, Apr. 1988.