DESCRIBING LOW-LEVEL IMAGE FEATURES USING THE COMM ONTOLOGY

Miroslav Vacura, Vojtěch Svátek

University of Economics, Prague
Dept. of Information and Knowledge
Engineering, W. Churchill Sq. 4
130 67 Praha 3, Czech Republic

Carsten Saathoff, Thomas Franz

University of Koblenz-Landau
ISWeb Group
Institute for Computer Science
D-56016 Koblenz, Germany

Raphaël Troncy

CWI
Kruislaan 413,
1098 SJ Amsterdam
The Netherlands

ABSTRACT

We present an innovative approach for storing and processing extracted low-level image features based on current Semantic Web technologies. We propose to use the COMM multimedia ontology as a "semantic" alternative to the MPEG-7 standard, which is at the same time largely compliant with it. We describe how COMM can be used directly or through its associated Java API.

1. INTRODUCTION

Intelligent processing of image data usually begins with extraction of some kind of low-level features from individual images. Regardless of the type of these features, the next logical step is to associate them with the original image, as well as to store this metadata information in some standardized format. This enables further reuse of this information by various third party tools and generally enhances the usability of the image analysis results. The MPEG-7 [1, 2] standard, syntactically based on XML Schema, is often used as a storage format. This standard includes most (though not all) descriptors that are required for representing low-level media-specific features extracted from multimedia data. Such extraction can often be performed automatically, using advanced image analysis tools.

This paper presents an alternative approach to the formal description of low-level image features based on COMM, a Core Ontology for MultiMedia. The ontology is available on the web[1] and its design rationale is described in [3]. We will thus, in this paper, reduce the general explanation of its underlying principles to a minimum, and will focus on the example of its practical application and on its software engineering aspects.

In general, why should one consider adopting the COMM multimedia ontology instead of the commonly-used MPEG-7 standard, when developing a multimedia analysis application? There is a number of aspects that potentially make the use of MPEG-7 cumbersome and ineffective. Many problems of MPEG-7 have been illustrated in depth in [3]. Most notably, MPEG-7 does not support machine processable semantic annotations of the image subject matter. At a syntactic level, MPEG-7 is based on XML Schema rather than on semantic languages such as RDF. Consequently, at the semantic level, the labels used in the annotations are either just plain strings or defined in lightweight MPEG-7-proprietary controlled vocabularies, incompatible with Semantic Web ontologies. Finally, the use of non dereferencable URNs goes against the whole Semantic Web vision.

The COMM multimedia ontology is designed for addressing these limitations. It is modeled using a sound ontological engineering approach, relying on the well-known foundational ontology *DOLCE* [4] and on the usage of *ontology content design patterns*. MPEG-7 is, however, an established standard in the multimedia community. COMM aims also to leverage on such a pre-existing body of knowledge, and allows the users to utilize their existing experience with MPEG-7. Therefore, COMM contains all MPEG-7 descriptors, formalized using the same naming convention as in the standard.

The COMM multimedia ontology is formalized in OWL[2], the standard ontology language promoted by W3C. The resulting multimedia annotations are expressed in the RDF/XML format, and can be stored in text files or in RDF repositories in general. Our multimedia annotation application in the K-Space project uses the *Sesame 2* RDF repository[3]. The employment of OWL and state-of-the-art metadata tools allows the user to benefit from the inherent reasoning capabilities, as the OWL language is formally grounded on *description logics* [5].

The rest of the paper is devoted to the process of describing low-level image features using COMM (section 2) and to the Java interface that supports the creation of COMM-based multimedia annotation by various applications (section 3).

[1] `http://multimedia.semanticweb.org/COMM/`

[2] `http://www.w3.org/2004/OWL/`

[3] `http://www.openrdf.org`

2. DESCRIBING LOW-LEVEL IMAGE FEATURES

The COMM multimedia ontology is *modularized* to a core module and to modules specialized on each media type. In this paper we only presuppose to work with image data, so considering the *visual* module together with the core module is sufficient. Aside design-time advantages, the modularization also minimizes execution overhead when processing data in an RDF store or when using some reasoning services.

The COMM ontology modeling approach is based on the *Descriptions and Situations* design pattern [6], which is in its most general form presented in Figure 1. It formalises the general paradigm of a situation and entities that constitute it (are the setting for it), and various means of describing this situation. This design pattern is in COMM particularly used in the two most common phases of multimedia description: decomposition and annotation. We consider that the annotation process is a situation (i.e. a reified context) that needs to be described. Such a situation then represent the state of affairs of all related data: actual multimedia data as well as metadata. Based on the general Descriptions and Situations design pattern, specialized patterns thus have been developed for these two phases: the *Decomposition* pattern and the *Annotation* pattern. In our example we only focus on the annotation phase, and assume that we use the dominant color descriptor to either describe the segment that arose by some previous preprocessing, or the whole image (in which case no segmentation was performed and the whole image is considered as the base segment).

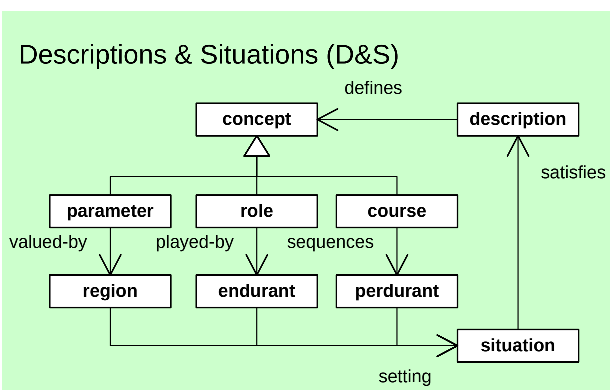

Fig. 1. Descriptions and Situations pattern

COMM supports the low-level audio and visual features represented in MPEG-7 including the color descriptors (layout, dominant color, etc.), motion descriptors (activity, trajectory, etc.), shape descriptors (contour, region, etc.) and texture descriptors (homogeneous texture, edges histogram, etc.). Other common terms used in signal processing and image analysis for describing detailed low-level features such as eccentricity, major axis length, lightest color, etc. are lacking in the MPEG-7 visual descriptors. These extra visual feature descriptors can be easily added in the COMM ontology, subclassing for example the color descriptors.

Even though COMM is quite comprehensive and can be used to describe any aspect of multimedia data in principle, in this paper we focus on presenting it as an easy, convenient and straightforward way to describe basic *low-level image features*. More specifically, in the example below, we deal with the *dominant color descriptor* as a well-known element of the MPEG-7 standard. The dominant color descriptor specifies a set of dominant colors in an arbitrarily shaped region [1] (Part 3). Such a region can be the result of a decomposition or the whole image. In MPEG-7, this descriptor is quite complex, as it may contain various information about the used color space, color quantization, etc. Even if all such information can be represented in COMM, for simplification and convenience we will only focus on two attributes: *ColorIndex*—the value specifying the index of the dominant color in the selected color space, and *Percentage*—the percentage of pixels that have the associated color value.

Figure 2 is the example image that we will use to extract low-level feature descriptors.

Fig. 2. French midfielder Zinedine Zidane converts a penalty kick for the game's first score during the semi-final World Cup football match between Portugal and France at Munich's World Cup Stadium, 05 July 2006. Credits AFP PHOTO / ARIS MESSINIS

The Annotation design pattern, refined for annotation via dominant color, is presented in Figure 3. Colored rectangles correspond to specialized classes necessary for the dominant color annotation. We consider individuals of class DominantColorAlgorithm to be individual *applications*[4] of given algorithm on the image or the segment. This enables us to store not only the results of image processing but also individual parameters (considered to be digital data) set in that particular processing. The fact that the algorithm outputs the value

[4]Strictly spoken, an application of an algorithm cannot be considered as its instance from the ontological modelling point of view; instead of 'application', we would have to formally consider a version of the algorithm that is 'fully-instantiated' (has all parameters set) as instance of the generic algorithm. Such versions of (deterministic) algorithms have, in most realistic cases, a 1-1 mapping with actual runs.

named ColorPercentage is then also considered as a kind of 'parameter' of algorithm. This parameter is then valued by a concrete scalar value. We also know that this value is part of what is named DominantColorValueDescriptor—this descriptor is, in a sense, about it. All this information is connected to original ImageData that play the InputRole in this process. The annotation pattern has been designed so that it is understandable to people with previous knowledge of MPEG-7 as well as to people without such experience. Unsurprisingly, some ontological design compromises were necessary to make the ontology easily usable, to overcome MPEG-7 drawbacks and to provide alignment to the DOLCE foundational ontology.

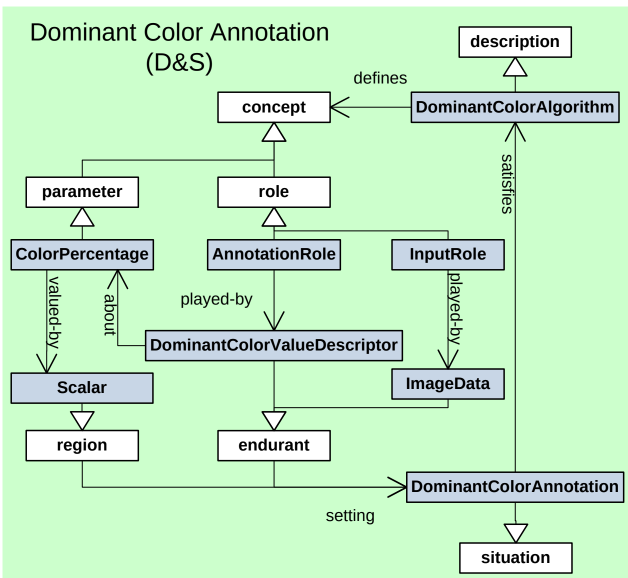

Fig. 3. Dominant Color Annotation Pattern

Figure 4 depicts the structure of individuals related to a sample image: There are 8 individuals and 11 relations. Note that the image data are presented in a rather simplified form here. COMM would also allow to describe them in a much more comprehensive way, using the Ontology of Information Objects (OIO) pattern [6], which enables the user to take care of multiple physical realizations of a single image data (e.g. multiple resolutions or multiple locations of the same image). In our case, for simplicity, we merely assign the dominant color annotation to the given image, represented by the ontology individual id0. Later, annotations of the same image obviously produce fewer individuals and relations, as they reuse parts of the existing annotation. The amount of data (RDF triples) to be inserted into the RDF data repository would be derived from the number of edges in the graph.

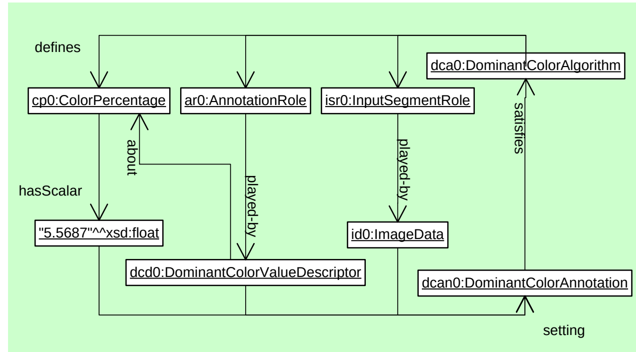

Fig. 4. Dominant Color Annotation

3. COMM JAVA API INTERFACE

A Java API (named COMM API) was developed as part of the support infrastructure for COMM. It provides an interface that allows to easily annotate multimedia data from the Java environment. COMM API is implemented with respect to two well used RDF stores – Sesame 2 and Jena[5]. It also provides help to a user who has no knowledge of ontologies, OWL or even RDF, as it completely shades the user from the data storage level. A potential user thus doesn't have to know anything about ontology engineering, and still can use COMM for describing multimedia data. Java COMM API follows many typical user intuitions based on MPEG-7, so users with previous experience with this standard find this framework familiar.

We demonstrate the ease of use of the COMM API on the previously-presented example: annotating a given image with an adequately extracted instance of the dominant color descriptor. We consider, however, the context of a real-world annotation making the situation more complex. Therefore, we also include, as part of the annotation, the reference to the image URI and some other constituents of the original MPEG-7 dominant color descriptor.

The first part of the code merely states that the image exists, and references (by an URI) the file in which it is physically stored. This is accomplished by using the `MediaProfile` class, which corresponds to the concept MediaFormatType in MPEG-7. The media profile is used to describe the physical realization of multimedia data.

```
MediaProfile mp = new MediaProfile();
MediaInstanceDescriptor mi = new MediaInstanceDescriptor();

UniqueIDDescriptor uid = new UniqueIDDescriptor();
uid.setUniqueID("unique-ID-of-image");
mi.setInstanceIdentifier(uid);

MediaLocatorDescriptor mld = new MediaLocatorDescriptor();
mld.setMediaURI(
  "http://newsml.cwi.nl/AFP/Photos-WC2006/DV87001_1.jpg");
mi.setMediaLocator(mld);
mp.addMediaInstance(mi);
Image img = new Image();
```

[5]http://jena.sourceforge.net

A new *still region* is then created. This still region represents the root of the image, i.e. the image as a whole. All further descriptors are added to this root region.

```
StillRegion sr0 = new StillRegion();
img.setImage(sr0);
sr0.addMediaProfile(mp);
```

We then create an object for representing the value of the descriptor (DominantColorValueDescriptor) and set its parameters.

```
DominantColorValueDescriptor d1
        = new DominantColorValueDescriptor();
d1.setPercentage(26);
Vector<Integer> vec1 = new Vector<Integer>();
vec1.add(7);
vec1.add(10);
vec1.add(3);
d1.setIndex(vec1);
```

The dominant color descriptor is created and set with the previous value defined.

```
DominantColorDescriptor dcd = new DominantColorDescriptor();
dcd.setSpatialCoherency(0);
dcd.setValue(d1, 0);
```

Finally, we assign values of other properties[6] to the dominant color descriptor and add it to the root region representing the whole image.

```
dcd.setValue(d2, 1);
dcd.setColorSpace(csd);
dcd.setColorQuantization(cqd);
sr0.addVisualDescriptor(dcd);
```

Note that the RDF/XML syntax of COMM is completely hidden from the user. The knowledge of some basic MPEG-7 conventions and the class structure of the COMM Java API is sufficient to create multimedia data annotations. The multimedia annotations and other associated metadata stored can then be queried using the SPARQL[7] query language. Simple reasoning such as subsumption entailment can be performed. For example, a SPARQL query for all color descriptors of Figure 2 will retrieve the dominant color descriptors together with the color layout and the color structure descriptors.

4. CONCLUSION

We have presented an approach for representing and processing the low-level feature metadata extracted by image analysis algorithms. The approach is based on using the COMM multimedia ontology, an alternative to the plain MPEG-7 standard while being a compatible semantic extension of it. We have described how to use the COMM ontology directly or using the associated Java API that makes its accessible even to persons with very limited knowledge of underlying Semantic Web technologies.

As the COMM-based approach is based on current Semantic Web technologies while respecting standards established in the field of multimedia analysis, it presents a bridge between low-level image analysis technologies and Semantic Web technologies, and paves the way to their future better integration leading to mutual benefit.

[6] We do not include the definition of these values for brevity.

[7] `http://www.w3.org/TR/rdf-sparql-query/`

Acknowledgments

Miroslav Vacura, Vojtěch Svátek, Carsten Saathoff and Raphaël Troncy are partially supported by the EU under the IST-6FP-027026 (Network of Excellence) K-Space[8]. Thomas Franz is partially supported by the EU under the IST-FP6-026978 X-Media project[9].

5. REFERENCES

[1] MPEG-7, "Multimedia Content Description Interface," ISO/IEC 15938, 2001.

[2] Frank Nack and Adam T. Lindsay, "Everything you wanted to know about MPEG-7 (Parts I & II)," *IEEE Multimedia*, vol. 6, no. 3-4, 1999.

[3] Richard Arndt, Raphaël Troncy, Steffen Staab, Lynda Hardman, and Miroslav Vacura, "COMM: Designing a Well-Founded Multimedia Ontology for the Web," in *6th International Semantic Web Conference (ISWC)*, 2007, pp. 30–43.

[4] Claudio Masolo, Stefano Borgo, Aldo Gangemi, Nicola Guarino, Alessandro Oltramari, and Luc Schneider, "The WonderWeb Library of Foundational Ontologies (WFOL)," Tech. Rep., WonderWeb Deliverable 17, 2002.

[5] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, *The Description Logic Handbook: Theory, Implementation, and Applications*, Cambridge University Press, 2003.

[6] Claudio Masolo, Laure Vieu, Emanuele Bottazzi, Carola Catenacci, Roberta Ferrario, Aldo Gangemi, and Nicola Guarino, "Social Roles and their Descriptions," in *9th International Conference on Principles of Knowledge Representation and Reasoning (KR)*, 2004, pp. 266–277.

[8] `http://www.k-space.eu/`

[9] `http://www.x-media-project.org`