ImageNotion as a Mashup Service for a Semantic Image Web*

Andreas Walter[1] and Gábor Nagypál[2]

[1] FZI Research Center for Information Technologies , Information Process Engineering, Haid-und-Neu-Straße 10-14, 76131 Karlsruhe, Germany
Andreas.Walter@fzi.de
[2] disy Informationssysteme GmbH, Erbprinzenstr. 4-12, Eingang B, 76133 Karlsruhe, Germany
nagypal@disy.net

Abstract. Most web image archives still use plain text to annotate images. The ImageNotion system ([1] [2]) extends the state-of-the-art by providing semantic annotation of images and their parts. In this paper, we show how to implement a mashup of ImageNotion with popular web image archives such as Flickr. This allows our users to load the images from external image platforms in ImageNotion and to automatically create semantic annotations using ImageNotion. Further, they can use the advanced search features of ImageNotion on those images, including the search for related images. In addition, we also show how to extend this mashup to a semantic web service. Such a service creates semantic image annotations semi-automatically, and thus makes those images available for processing on the Semantic Web.

1 Introduction

State-of-the-art image archives use textual annotations. Those textual annotations may be image caption, description or a set of simple tags, as in Web 2.0 systems. For users of such systems, using semantic image annotations instead of textual ones would provide many advantages. For example, ontologies and also thesauri make it possible to separate homonymous meanings of a tag into different ontology elements, such as "Paris" as a person (Paris Hilton) and "Paris" as a city. Additionally, it is also possible to propose semantically related images in an image archive by using "narrower/broader term" relations of thesauri, such as done in Riya [3]; or "subclass-of" and "is-a" relations of ontologies, such as done in SemSpace [4].

Further, users can explore the contents of an image archive instead of formulating precise queries. They can navigate to related images, following links to topics, persons and events in the context of their current search request. E.g., they may start searching for all "portraits of actors from Germany", refine their search to "images from female actors in the movie Barfuss" (e.g. *Johanna Wokalek* and *Alexandra Neldel*), and finally to "images of the main actors of Barfuss" (*Johanna Wokalek* and *Til Schweiger*) just by following links among ontology elements that are attached to the images or to their parts.

* This work was co-funded by the European Commission within the project IMAGINATION.

Semantic image annotations also allow automated systems, e.g. agents, to query for images, and to "understand" the displayed image contents [5]. E.g., agents can search for images containing persons, and they can understand who are those people based using the semantic image annotations.

Since state-of-the-art systems use textual image annotations, a new system is required that is capable of creating semantic image annotations, of transforming textual annotations to semantic annotations, and of creating semantic annotations on image parts. The ImageNotion application [1] provides a visual methodology that supports collaborative, work-integrated ontology development. ImageNotion allows for semantic search and for the navigation through image archives based on the available image annotations.

In most information systems, it is clear that semantic technologies would help the system to provide a better end-user experience. As the creation of semantic annotation is very time consuming, using only manual annotation methods often make it impossible to employ such technologies in real-world systems. To overcome this problem in ImageNotion, we automatize the generation of semantic annotations to the maximum possible extent.

Users have already uploaded many images to many different image platforms, e.g. to Flickr or Facebook. ImageNotion should be able to use these images directly from their sources. In this paper, we propose a system architecture to use ImageNotion as a mashup service. The service should read images from image sources available on the Web, and should automatically create semantic image annotations for them. This mashup then may work as a middleware between the world of textual image annotations on the one hand, and between the world of Semantic Web relying on semantic annotations on the other hand.

Based on this mashup, we also show how ImageNotion may run as a Semantic Web service for images. This Semantic Web service creates image annotations that use popular ontologies [6] and thus provides semantic interoperability with other image processing applications on the Semantic Web.

The paper is structured as followed: in Section 2, we discuss related work. Section 3 gives an overview about the features of ImageNotion. These features include collaborative and work integrated ontology development, the combination of the results using automated processes to improve the quality of generated metadata, and using the semantic annotations during semantic search. In Section 4, the mashup system based on ImageNotion is introduced. This mashup is extended to a Semantic Web service in Section 5. Section 6 finally concludes and gives some outlook.

2 Related work

In this section, we report on related work that is relevant to our goal to extend ImageNotion to a mashup service and to provide a Semantic Web service for this mashup.

Data integration from image platforms: For our mashup, the integration of image metadata from image platforms is necessary. Image platforms include portals (such as the German portal fotomarktplatz[3] for professional photographers and image agencies),

[3] www.fotomarktplatz.de

images displayed on web pages (e.g., crawled by Google), photo sharing platforms (e.g., MySpace, Flickr and Riya) and social network sites (e.g. Facebook, studiVZ or LinkedIn). We give an overview on possible data integration techniques for these image platforms.

One possibility is to use the OpenSocial API[7]. It defines a set of commonly used, standardized methods for social network sites. Thus, it allows for interchanging and linking among others profile data, friend lists and even images from various sources supporting this standard. E.g. images from MySpace [8] can be accessed this way. Currently, for most sites, e.g. Flickr, Facebook and Riya, OpenSocial is not supported but proprietary APIs are offered and proprietary data exchange formats are provided. For such services, a wrapper is needed. The APIs allow either to retrieve all publicly available images or images of a given user where the the user agreed exchanging data with the service using the API. Via the APIs, it is possible to read the existing image annotations and also to retrieve the images themselves. Image platforms, such as Riya or Facebook, that support the annotation of image parts, also support the retrieval of image part annotations [9].

Integrating automated processes: Creating image annotations, and especially annotations for image parts, is very time consuming. This process should be automatized to as much as possible. We give an overview how automated processes are used in other image platforms. Tag4you[4] uses the Flickr API to allow face detection in Flickr images. The system automatically marks the areas of detected faces and also allows adding tags to those areas manually. The tags are then automatically written back to Flickr. Riya offers face detection and recognition algorithms. Text recognition based on OCR in images is also provided. The ImageSorter application [10] sorts images by similarity of colors and structures and allows for searching for "similar images by structure or color". The face detection and recognition algorithm of Fraunhofer IIS [11] (we use it in ImageNotion) also supports gender classification and can detect moods like happy, angry or surprised.

Semantic image annotation: To create semantic image annotations, users have to be supported by adequate tools. RDFPic [12] and PhotoStuff [4] allow for the generation of semantic image annotations using RDF and imported domain ontologies. Both applications are only available as desktop applications and offer no support for collaborative ontology development and semantic annotation.

Standardized ontologies for semantic interoperability on the Semantic Web: Semantic Web services must use standardized ontologies and multimedia standards so that they can provide semantic description of the images that can be interpreted in different systems [6,5]. The internal ontology of the system therefore should be mapped to commonly used domain ontologies [6] describing people, objects, events and their relations. FOAF [13] is an ontology for the Semantic Web to describe people, their activities and their relations to other people and objects. In our case, this ontology may be used to exchange semantic annotations of people. In a similar way, the CYC ontology [14] can be used to describe objects and CIDOC-CRM is a standard ontology for describing cultural heritage objects. Using these ontologies, it is possible to map the ontology internally used in ImageNotion to these standards, as long as adequate concepts

[4] www.tag4you.com

are available in the internal ontology. Otherwise, the mapping is only possible with core concepts of these ontologies [15].

MPEG-7 [16] can describe the structure of an image, including the contours of image parts. In addition, it is also possible to include semantic annotations of image parts or of the whole image. The annotations may be defined in an arbitrary ontology, such as FOAF or CYC. MPEG-7 is therefore suitable for exchanging semantic metadata between multimedia information systems.

3 The ImageNotion application

In this section, we give a brief overview on the ImageNotion application. For further details please refer to [1,2,17]. The ImageNotion application is publicly available at `www.imagenotion.com`.

3.1 Collaborative and work-integrated ontology development

We call our ontology elements (concepts and instances) *imagenotions*, formed from the words image and notion. An imagenotion visually represents an ontology element through corresponding images. The visual representation of ontology elements helps image annotators to get a better understanding of their meaning. Based on the ontology maturing process model [18], a collaborative and work integrated ontology development methodology is implemented in ImageNotion. New imagenotions may be added by users in the first phase. In the next step, imagenotions are consolidated in communities of users (in Fig. 1, the annotators add descriptive information for the current EU president Manuel Barroso). In this phase, a stable definition of the concept emerges as users communicate with each other, or work on the same concept definition. In the third phase, it is possible to add relations between imagenotions (see Fig. 2, where annotators added relations to “president”, “male” and “EU commission”). Imagenotions from each maturing grade may be used for semantic image annotations immediately after creation.

Fig. 1. Adding descriptive information to imagenotions

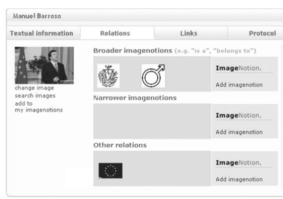

Fig. 2. Adding relations to imagenotions

The ImageNotion methodology allows to start the development of a new ontology either from scratch or by reusing existing ontologies (such as CIDOC-CRM or CYC) or parts of these. A community may then collaboratively add further ontology elements. For example in the EU project IMAGINATION, the group of image annotators decided to start with core parts of the multimedia ontology CIDOC-CRM and to use only the relations broader, narrower and unnamed relations from SKOS [19].

3.2 Combining automated processes in ImageNotion

For the automatic generation of semantic annotations, we combine the results of face detection and recognition (from Fraunhofer IIS, [11]), object and person detection (from NTUA, [20]) and text mining algorithms (e.g. text classification of JSI [21]). The generated semantic annotations use the existing imagenotions in the ontology.

Example: Fig. 3 shows the result of applying the automated processes in ImageNotion for a given image. The text mining algorithm has created the semantic annotations "Romano Prodi" and "Manuel Barroso", based on the textual title of the image: "EU president Barroso meets Prodi". The person and object detection algorithms have created bounding boxes for the shapes of the two persons. The face detection algorithm has created two bounding boxes for the detected faces. The face recognition has identified "Manuel Barroso" with a score of 80 percent and "Jan Figel" with a score of 20 percent for Barroso's face. Romano Prodi's face was identified as male person, but was not recognized by the face recognition algorithm.

The controller now combines the results (see Fig. 4). For the second face, it creates a new image annotation "Romano Prodi" for Prodi's face with a score of 100 percent and sets the score of "Male" to zero. This is possible because the annotations created by text mining state that there must be the persons "Romano Prodi" and "Manuel Barroso" on the image, and the other region was already correctly recognized as Manuel Barroso. Also, it sets the score of the detected areas for the persons to zero, since faces inside those areas were detected, and our end users prefer annotating faces to annotating person bodies.

Training of the automated processes Some of the automated processes needs training to provide acceptable results. This is the most challenging for the face recognition algorithm where each person that should be recognized by the algorithm, needs training images. In ImageNotion, we embed the specification of training images into the process of creating new imagenotions and new semantic annotations as follows:

1. A user uploads images. One or more faces on these images are unknown for the system or although the faces are known, they cannot be recognized correctly for some reasons. The face detection algorithm that operates very reliable, determines the bounding boxes also for such faces. The system notices that the face recognition algorithm failed to recognize the person because the recognition score is too low.
2. The web user interface asks, who the person is. The user may associate the bounding box with an existing imagenotion of a person or create a new imagenotion.
3. If a new imagenotion is added, gender detection is additionally executed the user is asked whether the detected gender is correct. Finally, relations to the imagenotions "gender" and "person" are added automatically.
4. The image part showing the person is added to the training images of her imagenotion.

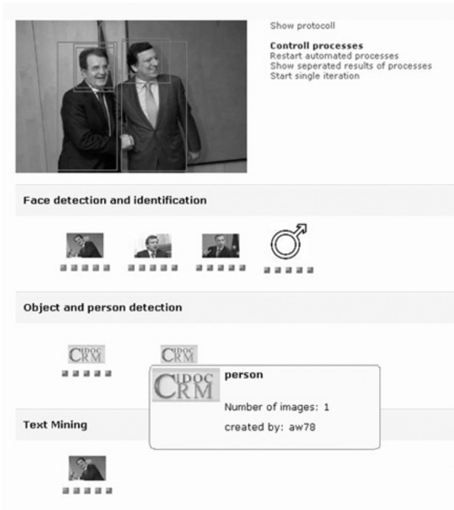

Fig. 3. Results of the automated processes

Fig. 4. Overall result

3.3 Visualization of semantic annotations and semantic search

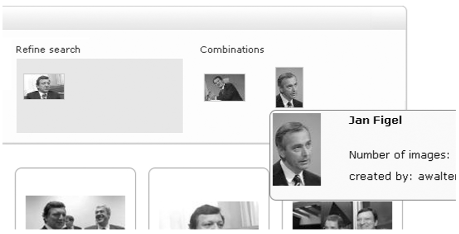

Fig. 5. Visual search refinment

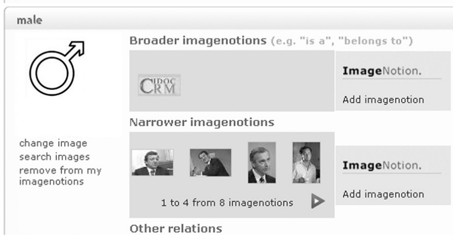

Fig. 6. Browsing the ontology

The manually or automatically created semantic image annotations can be used in ImageNotion for the navigation through the image archive. Here, we give a short overview of the already implemented techniques. For details, we refer to [2]. A user may start with a full text search and visually choose her desired imagenotions to start a semantic search. When a user clicks on an image part, the user may choose between starting a new search, refining an existing search or viewing the details of the imagenotion used for the semantic image annotation.

Fig. 5 shows the image clustering technique in our system. After a semantic search, annotations of the resulting images are analyzed, and the imagenotions that are most frequently used together with the imagenotion(s) forming the semantic search are grouped together in a cluster, e.g. all images displaying “Manuel Prodi” together with “Jan

Figel". Fig. 6 shows an example scenario of our ontology browser. By searching for "male", a user can browse all existing imagenotions for male people to start new semantic search requests.

4 Building mashups with ImageNotion

The ImageNotion application currently stores all images, ontologies and semantic annotations in its local database. Our aim is to integrate images from external image sources by building a mashup system. Then, it would be possible to use the ImageNotion application for semantic search and visualization of the semantic annotations for these images. In addition, we can provide an API to read the generated annotations. In this case, the integration of ImageNotion in existing applications, e.g., in image search engines, is possible. Those applications may then benefit from the semantic annotations generated by ImageNotion.

4.1 System architecture

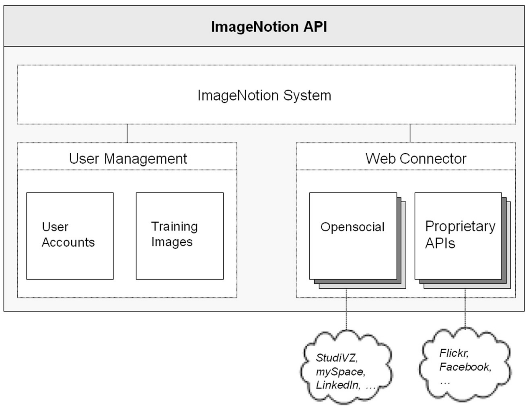

Fig. 7. System architecture for building mashups with ImageNotion

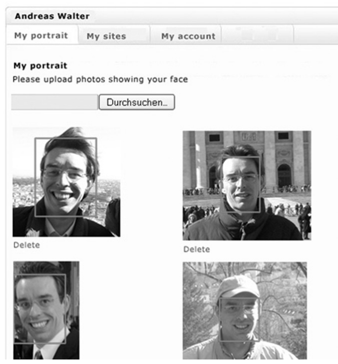

Fig. 8. User inteface of the mashup sytem for the user Andreas Walter

Fig. 7 shows the system architecture for building ImageNotion as mashup system. It consists of a user management module for storing user related information and a web connector module for connecting to the image sources.

User management module: This module stores the information of users accounts in external image platforms in the *user account* component. Consequently, the mashup application can load all images of a user and create semantic image annotations for them. Because face recognition requires training data (see Section 3.2), a user may train some faces using the procedure introduced before. The training images are stored in the *training images* component and passed to the face detection and recognition algorithms as training data.

Web connector module: The web connector module is responsible for loading all images from the desired image platforms. Each platform with a proprietary API requires a wrapper. Each wrapper connects to the corresponding image platform, e.g. Flickr, to load the images. With the *OpenSocial* component, we could load images from all social network sites which implement this standard, such as MySpace.

Subscription for new images: To load new images, the mashup system can be configured in two different ways. In the *user defined* setup, the system loads only images from the image platforms for which a user provided account information. In the *public images* setup, the system regularly polls new images in the connected image platforms (e.g., by searching for "all uploaded images in the last 10 minutes").

Creating semantic image annotations: The *web connector module* loads images from the image platforms in preview size (e.g. 800 pixel picture size), creates a unique identifier (currently the source name, e.g. Flickr, combined with the image name in the source) for the image, and passes the image object together with all available textual annotations or annotations for image parts to the ImageNotion system. The ImageNotion application generates semantic annotations for the image. The image annotations are stored in the ImageNotion system and are linked to the unique identifier of the image.

4.2 Prototype implementation

Our current prototype implementation allows one to read images from Flickr and Riya. We use the *user defined* setup for the subscription of new images. A user may define her accounts to these two sites and the web connector retrieves the data of her images (see Fig. 8). The ImageNotion system then creates semantic image annotations for these images.

4.3 Integration of the mashup system in other applications

By providing an API, we make it possible to extend existing applications with the features of the ImageNotion application. E.g., one may improve an existing web portal of images by embedding the semantic search feature of ImageNotion into the portal. ImageNotion may even be embedded in the case when the portal accesses multiple independent image sources.

The ImageNotion API is currently under construction. We will offer methods for generating semantic annotations, training new faces, adding new images and performing search requests.

5 ImageNotion as a Semantic Web service

In this section, we publish the ImageNotion mashup system as a Semantic Web service. By doing that we can fulfill the following goals. First, the Semantic Web service can automatically generate semantic image annotations for any image on the Web. The service gets an URL of an image in the service request, creates the semantic annotations, and finally returns these annotations in the service response. Second, the already introduced mashup service that accesses and integrates multiple image archives can be globally published as a semantic search service. In this section, we introduce a suitable system architecture for the ImageNotion Semantic Web service. We also discuss a possible solution to achieve semantical interoperability between ImageNotion and other Semantic Web systems.

5.1 System architecture

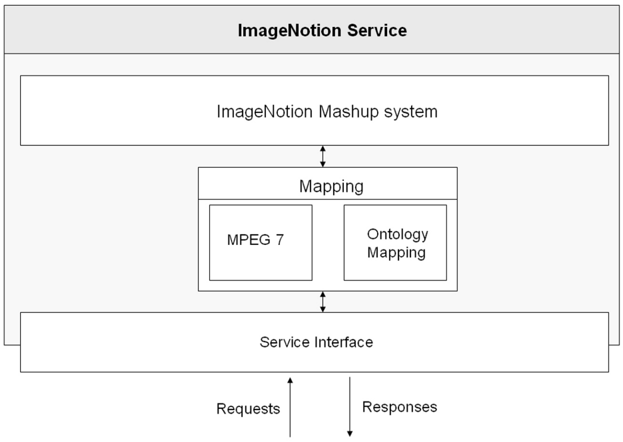

Fig. 9. System architecture for ImageNotion as Semantic Web service

Fig. 9 shows a possible system architecture for the ImageNotion Semantic Web service. The system architecture is based on the mashup architecture introduced in the last section. The architecture is extended with two new components, one for ontology mapping purposes and the other to provide the web service interface.

Service interface: The service interface offers methods for starting search requests and for initiating the annotation of new images. For example the service method *doImageAnnotation* receives as request the URL of an image and forwards this information to the *ImageNotion mashup* component. It creates the image annotations and returns them to the service interfaces. The service interface then returns the generated annotations as its response.

Ontology mapping: In the Semantic Web, standardized ontologies help exchange information between different applications. Therefore, our service should map the internally used ontologies to popular standard ontologies. FOAF is a popular format to describe information about people and thus may be used in our system. Among others, FOAF can also represent gender information. Objects and events could be mapped to core concepts of CIDOC-CRM (as it is done e.g. in the IMAGINATION[5] project).

[5] http://www.imagination-project.org

Alternatively, a mapping to matching elements or to the core concepts of the CYC knowledge base is also possible.

Semantic interoperability with other applications: MPEG-7 provides a standardized way to describe multimedia objects and their parts. The *MPEG-7* component receives the information about image regions from the ImageNotion system and it creates the MPEG-7 description of these image regions. The component receives the semantic image annotation mapped to FOAF and to CIDOC-CRM or CYC from the *ontology mapping* component.

5.2 Example use of the service for semantic image annotation

We now give an example how the ImageNotion Semantic Web may be used to create semantic annotations for an image by using the service method *doImageAnnotation*. The method is invoked by a simple GET request, the URL of the image is passed as a standard URI parameter. In our example, the image that is referenced by the URI is shown in Fig. 10.

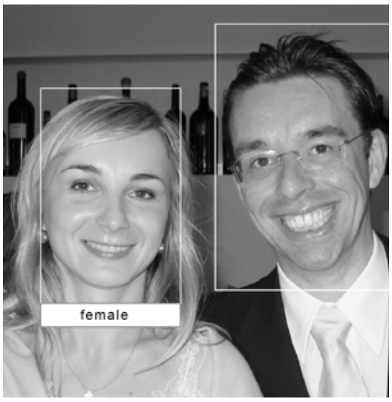

Fig. 10. Example image for the service


```
<Mpeg7>
 <MultimediaContent xsi:type="ImageType">
  <Image>
   <MediaLocator><MediaUri>
    URL of image
   </MediaUri></MediaLocator>
   <SpatialDecomposition>
   <StillImage>
   <StillRegion>
    <Region>(10,10) (100,100)</Region>
     <Semantic>
      <foaf:Person xmlns:foaf=
       "http://xmlns.com/foaf/0.1/"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
        syntax-ns#" rdf:about=
        "http://www.imagenotion.com/foaf/andreaswalter" />
      </Semantic>
     </StillRegion>
     <StillRegion>
      <Region>(140,10) (240,9)</Region>
      <Semantic>
       <foaf:Person xmlns:foaf="http://xmlns.com/foaf/0.1/">
        <foaf:gender>female</foaf:gender>
       </foaf:Person>
       </Semantic>
      </StillRegion>
    </StillImage>
    </SpatialDecomposition>
   </Image>
  </MultimediaContent>
</Mpeg7>
```


Fig. 11. Service response in MPEG-7 format

Fig. 11 shows the service response to the request in MPEG-7 format. The result in this example contains two semantic image annotations. For the first annotation box, the person “Andreas Walter” was recognized correctly and is mapped to FOAF. For the second annotation box, the face detection has detected a female person but she could not be recognized.

6 Conclusions and future work

State of the art image repositories still use textual image annotations. However, image services on the Semantic Web must be able to provide semantic image annotations instead of textual ones so that their content is processable by automatic agents. ImageNotion helps bridge the gap between text based systems and the Semantic Web by automatically transforming textual image annotation to semantic ones. ImageNotion can automatically generate semantic image annotations using a combination of face detection and recognition, person and object detection, and text mining algorithms. Moreover, the collaborative manual development of semantic annotations is also supported.

In this paper, we have first proposed a mashup system based on ImageNotion. This mashup extends ImageNotion with the ability to use images from external sources, e.g., from popular image platforms like Riya, Flickr or Facebook. In such a system, users can benefit from getting all their images annotated with semantic annotations in ImageNotion, without the need to upload them again.

In the second step, we proposed a system architecture to publish ImageNotion as a Semantic Web service. Such a service may support various scenarios. First, it is possible to send an image URL to the service and get the semantic image annotations for the image. In addition, it is also possible to install a mashup service over different image platforms and to offer semantic search functionalities via this service. To be interoperable with other services and agents on the Semantic Web, it is important to use established Semantic Web standards. Therefore, we proposed to use FOAF for the description of people and CIDOC-CRM or CYC for describing objects, places and events. In addition, we propose to use the MPEG-7 standard for structural description of images. An important feature of MPEG-7 that it supports the description of image regions and their semantic annotations.

Currently, the mashup system is only implemented as a small prototype and the Semantic Web service layer is not yet implemented. In our current and future work, we will connect our system to a higher number of image platforms, we will implement the API and will add the Semantic Web service layer. In addition, we are already working on a mashup which uses the ImageNotion API. In the PRIMO system [22], this API is used to build a privacy-aware Web 2.0 image sharing application.

References

1. Walter, A., Nagypal, G.: Imagenotion – methodology, tool support and evaluation. In: GADA/DOA/CoopIS/ODBASE 2007 Confederated International Conferences DOA, CoopIS and ODBASE, Proceedings, LNCS. Springer. (2007)
2. Walter, A., Nagypal, G., Nagi, K.: Evaluating semantic techniques for the exploration of image archives on the example of the imagenotion system. In: Alexandria Engineering Journal (AEJ), Vol. 47, No. 4. ISSN: 1110-0168 Springer. (2008)
3. Riya: Riya - Visual search. http://www.riya.com/ (2007) (accessed 2007-12-09).
4. van Ossenbruggen, J., Troncy, R., Stamou, G., Pan, J.Z.: Image Annotation on the Semantic Web. W3C working draft, W3C (2006)
5. Bashir, A., Khan, L.: A framework for image annotation using semantic web. In: MSW2004, located at the 10th International ACM SIGKDD Conference on Knowledge Discovery and Data Mining KDD 2004, 22nd August 2004 - Seattle, WA, USA. (2004)

6. Stamou, G., van Ossenbruggen, J., Pan, J., Schreiber, G.: Multimedia annotations on the semantic web. In: IEEE Multimedia, 13(1):86–90, January-March 2006. (2006)
7. OpenSocial: Opensocial. http://code.google.com/apis/opensocial/ (2008) (accessed 2008-09-09).
8. MySpace: Myspace specific extensions on opensocial. http://tinyurl.com/ytpfgl (2008) (accessed 2008-09-09).
9. riyaApi: Riya API . http://www.riya.com/riyaAPI (2008) (accessed 2008-09-09).
10. ImageSorter: ImageSorter V3. http://mmk1.f4.fhtw-berlin.de/Projekte/ImageSorter (2008) (accessed 2008-09-09).
11. Küblbeck, C., Ernst, A.: Face detection and tracking in video sequences using the modified census transformation. In: Image and Vision Computing, Editor: K.D. Baker, Issue No. 6, Elsevier. (2006)
12. Lavon, Y., Boss, B.: Describing and retrieving photos using RDF and HTTP. W3C working draft, W3C (2006)
13. FOAF: The Friend of a Friend (FOAF) project. http://www.foaf-project.org/ (2008) (accessed 2008-09-09).
14. CYC: OpenCyc for the Semantic Web. http://sw.opencyc.org/ (2008) (accessed 2008-09-09).
15. Walter, A., Nagypál, G.: Semantical interoperability with IMAGINATION content using standardized ontologies. In: Workshop on Ontology-Driven Interoperability for Cultural Heritage Objects, Joint DELOS-MultiMatch event, Tirrenia, Pisa (Italy) (2007)
16. Martinez, J. M.: MPEG-7 Overview Version 10 . http://tinyurl.com/gcpfh (2005) (accessed 2008-09-09).
17. Walter, A., Nagypal, G.: The combination of techniques for automatic semantic image annotation generation in the imagination application. In: 5th European Semantic Web Conference, ESWC 2008, Tenerife, Canary Islands, Spain, June 1-5, 2008 Proceedings. (2008)
18. Braun, S., Schmidt, A., Walter, A., Nagypal, G., Zacharias, V.: Ontology maturing: a collaborative web 2.0 approach to ontology engineering. In: Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge at the 16th International World Wide Web Conference (WWW 07), Banff, Canada. (2007)
19. Brickley, D., Miles, A.: SKOS Core Vocabulary Specification. W3C working draft, W3C (2005)
20. Kapsalas, P., Rapantzikos, K., Sofou, A., Avrithis, Y.: Regions of interest for accurate object detection. In: TProc. of Sixth International Workshop on Content-Based Multimedia Indexing (CBMI 2008), London, UK. (2008)
21. Blaz Fortuna, Marko Grobelnik, D.M.: Semi-automatic data-driven ontology construction system. In: Proc. of the 9th International multi-conference Information Society IS-2006, Ljubljana. (2006)
22. Burghardt, T., Walter, A., Buchmann, E., Böhm, K.: Primo - towards privacy aware image sharing. In: to appear in proceedings of CISWSN. (2008)