Caliph&Emir: Semantic Annotation and Retrieval in Personal Digital Photo Libraries

Mathias Lux, Jutta Becker and Harald Krottmaier

Know-Center, Inffeldgasse 16c, A-8010 Graz, Austria
[mlux, jbecker]@know-center.at
Institute for Information Processing and Computer Supported new Media (IICM),
Graz, University of Technology, Inffeldgasse 16c, A-8010 Graz, Austria
hkrott@iicm.edu

Abstract. Nowadays research and development activities are accompanied by an increasing focus on future user needs in the field of multimedia retrieval. The fast growing of multimedia data repositories is an undeniable fact. Besides big companies like newspaper publishers, broadcasting stations, car manufacturers etc., dealing with digital content, an increasing number of private users invent private multimedia repositories for their self-made photos and videos. The development of suitable tools has to fulfill special requirements concerning the storage, indexing and retrieval of multimedia content. In addition easy-to-use content exchange via the Internet is a preferable feature. The transition from text to photo retrieval raises the necessity of additional meta-information about the content to allow semantic retrieval. As a result metadata has to be generated, stored and indexed to enrich raw visual information. As a result and proof of concept a pair of prototypes, called "Caliph & Emir", are presented.

Introduction

The evolution of digital information repositories produces more and more specialized requirements towards intelligent information retrieval. Numerous research and development teams are doing fundamental research concerning various unforeseen topics like managing more than 200 TV channels with a remote control without loosing orientation. Base for interdisciplinary future developments are overall agreed standards and standardized methods. Using the following scenario we examined the possibilities current technologies, like MPEG-7, bring us in context of one real world problem:

A user of a digital camera produces lots of images throughout a year and saves them to his local hard-disk. The number of taken photos soon exceeds the critical mass for being manageable without specialized tools. To overcome this problem most people create an intuitive structure for storing their personal image library. They create folders for images taken in the same context, for example "Photos from the I-Know '02 conference" or "Birthday party of uncle Adam". Nevertheless this does not enable the user to find a photo that shows a certain person, object or even expresses a

specific idea or feeling when needed. Some file formats like TIFF and JPEG permit the user to enrich the visual information with structured textual descriptions ([13], [17]), but they only offer limited retrieval capabilities.

In this paper we specify the requirements for managing personal digital photo repositories and then take a look on existing tools. The realization of Caliph & Emir, the software prototypes, is discussed. The chapter future work identifies possible next steps.

Requirements for managing personal digital photo repositories

Semantic retrieval of digital photos is not possible without existing metadata describing the image-content and the image itself. Most digital photos do already contain some metadata stored in the image files itself, whereas the format of the metadata depends on the digital camera the photos were taken with. For instance the model and manufacturer of the camera, exposure values or timestamps for creating and saving the image are saved in common digital cameras inside the image files using the Exchangeable Image File Format, EXIF [17]. These values have to be imported into a description. Besides defining who has taken the image and where it was taken it should also be possible to save who has created the metadata. Obviously, the possibility of describing the image with a short text should be integrated. However the main objective is to create, compute and save a description of the visual contents of the image, besides the user should also be allowed to describe not visible content like feelings or states like “friendship” or “comradeship”, which might be associated with the image. In addition to the time and place where the picture was taken, persons, person groups and objects have to be described to improve retrieval capabilities. These description entities are interconnected as can be easy seen in following example:

“Puffy, the dog which belongs to my uncle Adam” (**1**)

(1) defines a dog named “Puffy” and a person called “Adam”, which is the owner of the dog, and has a kinsmanlike relation to the narrator. This can be visualized in a directed graph shown in figure 1. The benefit of storing a description in such a structured way is rather obvious: the description becomes computable, for instance software agents could deal with those semantic graphs.

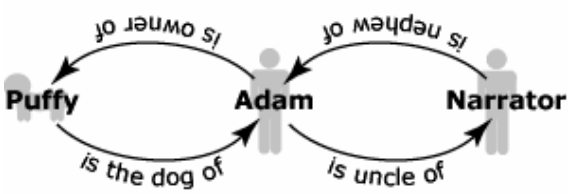

Fig. 1. Visualization of the description “Puffy, the dog, which belongs to my uncle Adam”

Another intuitive requirement is the retrieval of similar images using low-level content-based characteristics like color histograms or textures of images. Subjective and objective quality ratings are also important features for image retrieval, for instance used as a search constraint. If the image retrieval takes place in an internet based or mobile environment, several instances of an image in various resolutions and colorspaces are needed.

While image content and metadata based retrieval is not supported by existing operating systems, image browsing and categorization is already a common feature, for instance with Microsoft Windows XP. Our approach is based on the idea of integrating semantic descriptions into a framework of annotation and a following retrieval using common and established technologies for support.

Existing tools and standards

The assessment of existing tools is based on the following criteria:

1. Browsing and categorization features
2. Annotation options and their level of abstraction and structure
3. Intelligence and relevance of retrieval
4. Allowance of metadata exchange
5. The degree of integration of features

Following the assessment of two software tools existing metadata standards are examined.

ACDSee 5.0

ACDSee is a commercial tool developed by ACD Systems [14]. It gives the user the possibility to browse collections of photos interactively. Thumbnails and filenames are used for previewing images and folders. The user can define a category for each image, browse and edit the EXIF [17] information and can give a description for the image by typing a text into four different textboxes which are reserved for keywords, description, author and additional notes. The annotated photos can be retrieved by an inbuilt search engine. Content based image retrieval is not supported and a differentiation between description of the image and description of the metadata is not possible. The metadata not based on the EXIF standard is stored and indexed in an inbuilt database. As a result the information cannot be extracted and used by another program without severe reengineering. However, the structure of metadata proposed by ACD Systems does not allow creating computable and complex semantic graphs.

PhotoFinder 3.02

PhotoFinder [15] is a project of the University of Maryland Human Computer Interaction Laboratory to research personal photo libraries. The program uses a Microsoft Access database to store the metadata of the photos. PhotoFinder stores the photos in so called collections. These collections contain all images regarding a specific sequence of photos. Potential metadata for collections are title, description, start and end date, keywords and location. Inside the collections reside the images, which, not obligatory, have following metadata tags: date, location, rating and number of people showing on the photo.

However the main annotation mechanism is to define names and after putting them into a list you can drag and drop these names onto your image. Besides the name or label itself, which is dropped onto the image, its position in the image is stored. The retrieval is based on these names. You can either drag and drop these names onto a search panel or search using keywords. In addition to this mechanism, saving and searching for a textual description of an image are also supported. Existing metadata like EXIF [17] is not used for annotation and retrieval purposes. Besides, while annotation using predefined person-lists can be seen as an approach to structured semantic descriptions, the creation of graph-like and computable descriptions is not supported.

Existing metadata standards

The standard being used to define the way of handling the metadata has to be a lot more powerful than EXIF or for instance Dublin Core. DC only defines 15 core qualifiers, which can be understood as metadata tags, which can be filled by the user [1]. A combination of Dublin Core and adapted Resource Description Framework structures, RDF [2], would at least permit a structured storage of graphs similar to the above example and a quality rating, although content based image retrieval would not be supported. An import of the EXIF information to a RDF-based structure is possible. The main proposition against RDF is that there exists, at this time, no standardized structure for saving all or most of the metadata defined in the requirements above. Although it would not prove impossible to create such a structure, to gain interoperability with other systems and implementations agreeing on the same RDF based enhancements with all other developers or vendors is necessary.

Based on this facts a much better choice is MPEG-7 [11], the “Multimedia Content Description Interface”. It is an international standard, defined by the Moving Pictures Expert Group, which specifies a metadata system for describing multimedia content. MPEG-7 defines structures, called descriptors, needed for annotation of audio, video and image data and a standardized way to create new descriptors. A descriptor consists of either values or other descriptors; the format for defining those descriptors is W3C’s XML Schema [19].

MPEG-7 matches the above requirements and because MPEG-7 is a standard for metadata-based description of multimedia, it is interchangeable between various programs from different vendors. Besides it is possible to integrate existing standards, for instance Dublin Core [18], in MPEG-7.

Most important feature for our purpose is the possibility of describing the semantics of multimedia data [3]. The nodes or vertices in the above graph can be defined with the MPEG-7 semantic entity tools, which describe semantic entities such as objects, places, agents, times, concepts and states. The edges are the common semantic relations defined by the standard.

At this time no MPEG-7 based tools for managing digital photo repositories are available.

Realization of Caliph & Emir

As MPEG-7 is a complex XML based standard, it would be no good idea to confront the user with a XML editor and an instruction manual as tools for expressing the semantics of a photo. To deal with large description graphs a visualization of this graph, besides a possibility to edit this graph interactively, is necessary. As a result "Caliph", the "Common And Lightweight Interactive PHoto annotation", was designed for supporting the user in the time consuming task of annotating photos.

For retrieval of the annotated photos "Emir", the "Experimental Metadata based Image Retrieval", was built. Because the implementation was planned as a prototype for researching the possibilities of creating semantic descriptions in context of administering personal digital photo archives, the retrieval prototype uses no database but the file system to store the descriptions. This reduces the speed of retrieval a lot but the retrieval remains platform independent and lightweight for easily trying and demonstrating the software without a connection to the internet.

Both, mainly autonomous, prototypes Caliph and Emir were implemented using Suns Java SDK 1.4 [5] while as runtime environment the versions JRE 1.4 and higher are supported. For XML handling the libraries JDOM [6] and JAXEN [7] are used since they provide high level functions for dealing with XML based contents, which speeds up the development significantly. For reading the EXIF information stored in the images Drew Noakes' exifExtractor classes [4] were used.

For testing a set of 160 photos from a Kodak DC 215 Zoom camera was used. All photos were taken in context of the I-Know '02 conference. The photos were categorized in six directories (in chronological order):

1. Preparations for conference
2. 1st day of the conference
3. On the way to the social event
4. Social Event
5. 2nd day of the conference
6. Internal after-party

Annotation with Caliph

Since Caliph is a Java Swing [16] application, the designing started with creating a user interface that divides the annotation methods from the image preview and file browsing mechanisms. The annotation methods where separated from each other in extending a JPanel GUI element for each method or logical group of methods. There

are panels for creating the ColorLayout and ScalableColor descriptor, which are extracted from the image on first loading. There is the so called “creation panel” which shows the EXIF tags and values and holds the creator of the image and there are the “metadata description panel” for defining version and author of the metadata description. The “quality rating panel” is used for assigning a quality value and defining the person who rated the image quality, and the “text annotation panel” allows the input of a simple textual description of the image contents. Since a series of photos should be annotated in short time the file browsing tool is a specialized table, which allows the user to select the image in a fast and intuitive way. Obviously a preview panel is also required to allow the user to examine the image, but also a full size preview has been implemented as well as the possibility to define an external image viewer, which can be called using a keyboard command to give the user the ability to use his favourite tools.

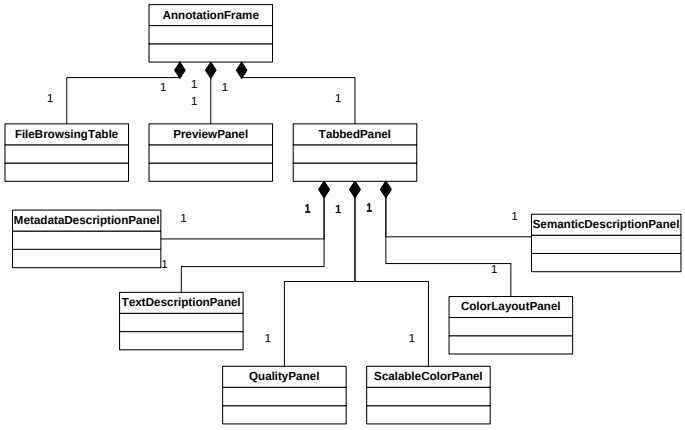

Fig. 2. Simplified UML diagram of Caliph

Central part of Caliph is the so called “semantic description panel”. It allows the user to define semantic objects like agents, places, events and times which are saved on exit for reusing them the next time starting Caliph. These semantic objects can also be imported from an existing MPEG-7 file to allow exchange of objects between users and editing and creating those objects in a user preferred tool. Semantic objects can be used for creating the description by dragging and dropping them onto the blue panel with the mouse, shown in figure 2. While testing this model we experienced, that this model is sufficient for users who often take pictures of the same persons and objects, and users that take pictures in series, which is quite often the case with hobby and amateur photographers. As once the objects exist, they can be reused if some pictures or series have the same context. This is especially true for objects representing persons, animals and places like the relatives, colleagues, friends, favorite pets or places like “at home” or “at work”.

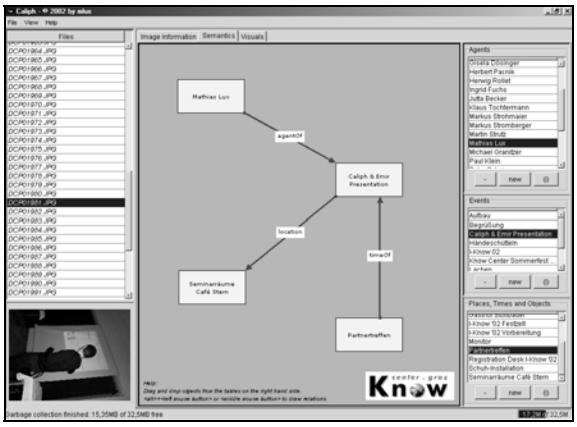

Fig. 3. Creating a semantic description using Caliph by drawing a graph as abstraction of the semantics

After dropping all the needed objects onto the blue panel the user can interconnect these objects by drawing relations between them using the middle mouse button. The graph, which is generated through these user interactions with Caliph, can be saved as part of an MPEG-7 description. In addition to the ability to create a new graph, Caliph is also a tool for importing, editing and deleting existing graphs or sub graphs.

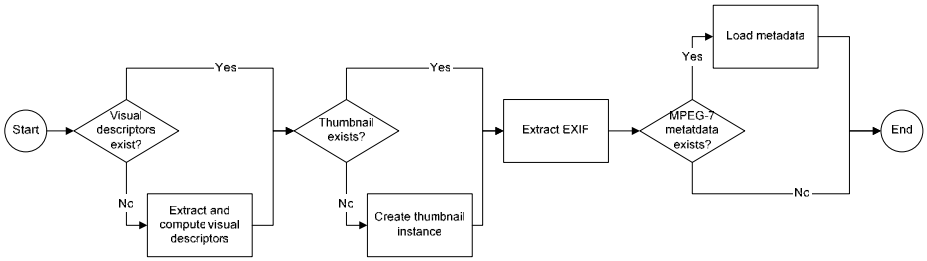

Fig. 4. Loading an image for annotation is a time consuming task if the visual descriptors have not been already created.

Further a whole series can be pre-annotated for simplifying and speeding up the task of annotating multiple images. All images within the same context are placed in one file system folder and the user opens the first one using Caliph. After defining a "base" description which is the same for all images of the series like the creator, a base textual description like "on our visit in Graz" and a base graph including the location where the photos were taken and time and motivation when they were taken. Finished with this minimal description the so called "autopilot" can be used, which opens all images in the defined folder sequentially, calculates the visual descriptors, which is a rather time consuming task depending on the size and resolution of the image, extracts the EXIF data and image specific parameters, creates a thumbnail

instance of the image for later retrieval and saves the base description. An obviously positive effect is that, when opening one of the pre-annotated photos afterwards, the thumbnail instance and the visual descriptors can be read from the existing metadata and do not have to be created, which saves time when opening a single image for editing.

Inside an MPEG-7 document the MediaProfile descriptor is used to reference instances of the media, described by the metadata. As well as the original imagef, which is referenced in the master profile, a thumbnail instance, created by Caliph if not already present, is referenced in another MediaProfile descriptor to allow the retrieval tool to use a prepared thumbnail instead of scaling the image during the visualization process of the retrieval results.

Retrieval with Emir

Emir gives the user the ability to retrieve annotated photos. Due to the fact, that this is experimental software the retrieval mechanism is file system based. All MPEG-7 documents found by Emir in a specified directory and in further subdirectories are searched.

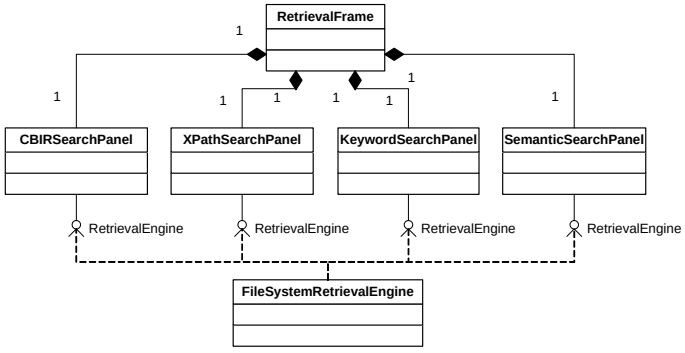

Fig. 5. Simplified UML diagram of the Emir retrieval architecture

Emir offers four different ways to search for a matching photo:

1. Searching through an XPath [10] statement
2. Defining search options through textboxes with various options
3. Content based image retrieval using the visual descriptors ColorLayout and ScalableColor defined in the MPEG-7 standard.
4. Searching for a similar semantic description graph

XPath search The first option is mainly used for developers and debugging of XPath statements, because all other retrieval mechanisms use XPath as query language. To search for matching documents using XPath requires detailed knowledge of the structure of the documents being searched, although basic statements like

```
//*[contains(.,'textToSearchFor')]
```

could be used querying documents without knowing the structure, but these statements only offer minimal retrieval features.

Text based search supported by a graphical user interface Well-known tools for defining search queries are GUI components for textual input and option lists to choose from. A user interface offering these components allows the user to specify what he searches for without knowing the structure of the documents being searched. Although the backend of the search also uses XPath to query the MPEG-7 data, the user never encounters the statement itself. Basic functions are searching for agents like a semantic agent or a metadata or image creator, searching for keywords in various textual descriptors and defining minimum or maximum quality as search constraint.

Content-based image retrieval Since MPEG-7 defines many low-level descriptors, a search mechanism using two of them for content-based image retrieval has been implemented.

The descriptor ScalableColor basically uses a color histogram which is prepared for difference measurement by transforming it using a Haar transformation [12]. The resulting coefficients can be used to calculate a color histogram based difference between two images. This descriptor proves useful if the images where rotated but retained their original colors.

The descriptor ColorLayout divides the image into 8x8 sub images. The dominant color of each tile is calculated and kind of a chessboard with a dominant color for each tile is created [12]. Based on these 64 dominant colors the difference between two images can be calculated. This method is stable against resizing the image, though a dominant color in a certain region retains dominant after resampling, while rotation changes the positioning of the tiles.

Semantic search The component of most interest is the panel offering a search mechanism for searching semantic descriptions.

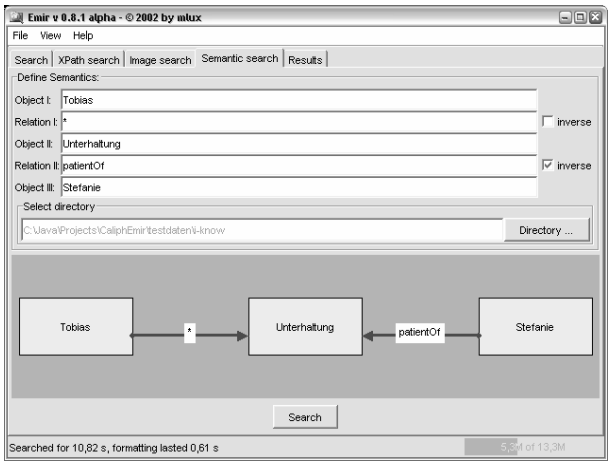

Fig. 6. Starting a semantic search using a graph as input. Three objects and two relations are defined, whereby one relation is defined by a wildcard instead of a keyword.

This component allows the user to define a graph with minimum one to maximum three nodes and two possible relations. An asterisk is used as wildcard. A search graph which only contains one node with a word defining this node will return each MPEG-7 document wherein a semantic object containing the specified word is found. If two or three nodes and one or two relations are used to define the search graph, the repository of MPEG-7 documents is filtered by the terms defined as objects or relations. If, for example, the graph in figure 6 above is used for search, all documents which contain semantic objects, which contain the terms “Tobias”, “Unterhaltung” and “Stefanie”, and a semantic relation containing the term “patientOf” are taken from the repository and checked if there is also a structural match with the given graph.

Fig. 7. Possible search graphs, which are supported in this prototype, are sub graphs of the maximum graph shown in this figure, whereas the maximum of relation between two nodes is one

The retrieval mechanism follows modular system architecture, an XPath statement is given to a class implementing the interface RetrievalEngine and the results are received as List of HTML documents, which can be visualized using standard Java Swing components. The only retrieval engine implemented yet is the “FileSystemRetrievalEngine”, which collects all MPEG-7 documents from a specified

directory and its subdirectories and executes the given XPath statement. If a matching document is found it is transformed into HTML, using XSLT [20]. This HTML result visualization is added to a list of results, which is ordered by relevance. Relevance is calculated using the number of nodes matching the XPath statement used as input. Another retrieval engine implementation would connect for instance to an XML database, which would result in a significant speedup executing the XPath statements.

In case of a content based image search each MPEG-7 document has to be loaded and the required descriptor is located using XPath. This descriptor has to be compared to the sample descriptor used as search parameter to calculate relevance. These results are put into a list ordered ascending by relevance, though a relevance of zero would show an exact match. Using a database the comparison of the descriptors has to be implemented on database side like a stored procedure, a server object or a similar mechanism, because of speed issues.

Results

Main result of the project was the identification of problems and their solutions with the examined technologies and methods while researching their practicability in the given context.

Photo annotation

Annotating digital photos is a very time consuming task. A very common problem is the extraction of existing and computable metadata like EXIF information and the visual descriptors like ColorLayout and ScalableColor. The time used for extracting a visual descriptor is proportional to the resolution of the image. Using a common computer like a Pentium III with 1.0 GHz and 256 MB of RAM the extraction of the two tested visual descriptors lasts for about 3 seconds. This cost can easily reduced by using a faster computer or extracting the metadata on a server or parallel to the interactions of the user.

Another problem is the interactive creation of the graph by the user. The creation of the main objects takes a lot of time. In another project [21], where all data comes from a specific context, a pre-built ontology based catalogue of semantic objects is used, which includes at least 95 percent of the needed objects. Using a catalogue like this in the context of a personal digital photo library does make sense, but it has to be updated and extended successively. If he user only takes photos in small numbers and on rare occasions, like two on a birthday party, three on his holidays, and one of his car newly washed, administrating and enhancing a catalogue of semantic objects demands more effort than typing in a textual description for each photo. Besides, in this case ability for retrieving annotated images will not be needed, because the number of photos will not exceed the critical mass for overlooking all of the images.

The graphical user interface of the annotation is, since it is a prototype, more or less an abstraction of the MPEG-7 descriptors, which is not intuitive for a user, who

does not really know about MPEG-7, so there has to be done a lot of work "hiding" the MPEG-7 from the user.

Existing metadata should not be lost while annotating the photos, but included in the MPEG-7 document. There are various ways in storing additional information inside an image, the two most common are EXIF, which is used by most digital camera manufacturers to save technical data about the photo, and a standard created by the IPTC [13], which is used for instance by the popular application Adobe Photoshop. The first one is very common and Java libraries for reading this information exist [4], while for the second one no Java implementation exists. A very interesting effect is that EXIF obviously allows the creator of the metadata to store the same information in different ways, which complicates a camera independent implementation. We experienced that Sony decided to store three tags for defining the time when the Photo was taken by using the tags "DateTime", "DateTimeDigitized" and "DateTimeOriginal", while Kodak only used the third one.

Semantic retrieval

A common problem with retrieval of XML documents is the speed, although Oracle and other big players in creating databases are already working on a possible solution. An also well-known fact is the insufficiency of XPath as query language. The upcoming standard XQuery [9] represents a possible solution. Nevertheless, most manufacturers and database vendors do not support it yet.

The computation of graph visualizations in a very semantic way with minimum crossings is also complicated in this context. For instance loading a semantic description and reading it does not prove as complicated, but arranging and visualizing the same graph can be quite tricky without generating a complete visual mess, which can only bring confusion to the user. Mostly a semantic description has a central element, take for example an image of a conversation between two persons. The central element is the conversation, therefore it should be placed in the center of the visualization, the objects representing the persons taking part in the conversion should be placed around this element along with a place, a time and a context for the conversation. Basically a very similar effect can be achieved if a visualization with a minimum number of edge crossings is calculated, because the central element takes part in most edges and therefore it is placed in the middle.

Another problem is that different users produce different descriptions for the same media instance. Also MPEG-7 defines an inverse for each semantic relation, as a result a user can choose either he takes a relation or its inverse to create the description. Therefore the retrieval mechanism must take care of this differences, like combined searching for relations and inverse relations and computing a similarity between semantic objects to identify objects which define the same person, thing or state but differ in the way they are described.

Finally the retrieval using a semantic graph as search parameter is not only a simple graph to graph matching based on a simple description standard but has to mind some parameters set from the MPEG-7 standard. In addition to the above mentioned inverse relations, MPEG-7 allows to integrate objects by reference. This means that the objects are used in the graph, but they are no fully described inside the

descriptor itself, but are only referencing the object, which is defined in another part of the document or even another document.

Although MPEG-7 defines similarity measurement for low-level content based descriptors it fails to define those measurement methods for calculating the similarity of two semantic graphs, so a generalized method has to be found and proposed.

Future work

While implementing Caliph and Emir talking to other people and possible users generated an immense transfer of ideas and knowledge. Besides enhancing the user interface and implementing a faster and fully featured version of a personal digital photo library some major topics were isolated.

An automation of annotation is required. Based on already defined MPEG-7 based visual descriptors some of the work, needed for annotation, should be done by the program, like detecting visually similar images and proposing they have similar semantics or object tracking through a sequence of images.

An Internet or intranet based retrieval engine was requested. Most companies, like the Know-Center or the IICM, have small to medium sized non-administrated image repositories like clip arts collection, royalty free images for ads and flyers or photos from objects, persons and events. Employees should be enabled to use these image libraries without the requirement of special knowledge or browsing all images. In this context user and rights management should be enabled to disable unauthorized access.

Many of the visual descriptors and combinations of descriptors have not yet proved functional in various contexts because they were not used yet. A lot of sample development and prototyping must happen to ensure context specific usefulness of parts of the new standard MPEG-7.

Conclusion

MPEG-7 matches many of the current requirements for a metadata standard for usage in a personal digital photo library and it defines a lot more useful descriptors, which could be integrated as features in such libraries. In addition it is not only a standard for describing the content of images, but it also defines ways to annotate video and audio documents and it is prepared for general usage with multimedia data.

Also a desktop enabled method of storing and indexing the multimedia data and its metadata has to be found. No common user will install an XML database for handling his photos because it requires a big deal on resources like storage and calculation power, however upcoming software and developments will significantly enhance the possibilities for dealing with data.

Acknowledgement

The Know-Center is a Competence Center funded within the Austrian K plus Competence Centers Program (www.kplus.at) under the auspices of the Austrian Ministry of Transport, Innovation and Technology.

References

[1] Using Dublin Core, Hillmann Diane, 12.04.2001, last visited 24.11.2002, http://dublincore.org/documents/usageguide/
[2] What is RDF?, Bray Tim, 24.01.2001, http://www.xml.com/pub/a/2001/01/24/rdf.html
[3] Semantics of Multimedia in MPEG-7, Benitez Ana B. et al, 2002, International Conference on Image Processing 2002, Proceedings Volume: 1, Page(s): 137 -140
[4] exifExtractor classes, Noakes Drew, last visited 10.10. 2002, http://drewnoakes.com/
[5] Java Software Development Kit, Sun Microsystems, last visited 10.10. 2002, http://java.sun.com
[6] JDOM Beta 8, JDOM.org, last visited 10.10. 2002, http://www.jdom.org
[7] JAXEN FC 1, JAXEN.org, last visited 10.10. 2002, http://www.jaxen.org
[8] What is XQuery?, Bothner Per, 16.10.2002, last visited 24.11.2002, http://www.xml.com/pub/a/2002/10/16/xquery.html
[9] XQuery 1.0 and XPath 2.0 Data Model, W3C Working Draft 15.11.2002, last visited 24.11.2002, http://www.w3.org/TR/query-datamodel/
[10] XML Path Language (XPath), W3C Recommendation 16.11.1999, last visited 24.11.2002, http://www.w3.org/TR/xpath
[11] Multimedia Content Description Interface — Part 5: Multimedia Description Schemes, MPEG, 23.10.2001
[12] Multimedia content description interface – Part 3 Visual, MPEG, Juli 2001
[13] Extracting IPTC header information from JPEG images, Tratz Christian, 28.08.2001, last visited 24.11.2002
[14] ACDSystems.com, ACD Systems, last visited 24.11.2002, http://www.acdsystems.com
[15] Photofinder project, University of Maryland Human Computer Interaction Laboratory, last visited 24.11.2002, http://www.cs.umd.edu/hcil/photolib/
[16] Creating a GUI with JFC Swing, Sun Microsystems, Inc, last visited 24.11.2002, http://java.sun.com/docs/books/tutorial/uiswing/
[17] EXIF.org, Hawkins John, last visited 24.11.2002, http://www.exif.org/
[18] Proposal for the Integration of DublinCore and MPEG-7, Hunter, Jane, October 2000, last visited 26.22.2002, http://www.egov.vic.gov.au/Documents/DCforMPeg7.doc
[19] XML Schema, W3C Recommendation , 2.5.2001, last visited 28.11.2002, http://www.w3.org/XML/Schema
[20] XSL Transformations (XSLT Version 1.0), W3C Recommendation, 16.11.1999, last visited 28.11.2002, http://www.w3.org/Style/XSL/
[21] XML and MPEG-7 for Interactive Annotation and Retrieval using Semantic Meta-data, , Becker Jutta et al, J.UCS Vol 8 Issue 10, 28.10.2002
[22] MPEG and its Relevance for Content-based Multimedia Retrieval, Haas Werner, Mayer Harald, J.UCS Vol 7 Issue 6, 10.7.2001, http://www.jucs.org/jucs_7_6/mpeg_and_its_relevance