Img(Rummager): An Interactive Content Based Image Retrieval System.

Savvas A. Chatzichristofis and Yiannis S. Boutalis
Department of Electrical & Computer Engineering
Democritus University of Thrace, Xanthi, Greece
schatzic@ee.duth.gr, ybout@ee.duth.gr

Mathias Lux
Institute of Information Technology
Klagenfurt University, Austria
mlux@itec.uni-klu.ac.at

***Abstract*—This paper presents an image retrieval suite called img(Rummager) which brings into effect a number of new as well as state of the art descriptors. The application can execute an image search based on a query image, either from XML-based index files, or directly from a folder containing image files, extracting the comparison features in real time. In addition the img(Rummager) application can execute a hybrid search of images from the application server, combining keyword information and visual similarity. Also img(Rummager) supports easy retrieval evaluation based on the normalized modified retrieval rank (NMRR) and average precision (AP).**

***Keywords*-Content Based Image Retrieval (CBIR); Low Level Features; Descriptors**

I. Introduction

Content Based Image Retrieval (**CBIR**) is any technology that in principle helps to organize digital image archives by their visual content. By this definition, anything ranging from an image similarity function to a robust image annotation engine falls under the purview of CBIR [1]. The most common form of CBIR is an image search based on visual example. The user inputs an image (query image), and, based on certain global features, the system brings up similar images. This sort of feature is used for describing the content of the image and that is why they must be appropriately selected on occasion. The visual content of the images is mapped into a new space called the feature space. The features that are chosen have to be discriminative and sufficient for the description of the objects.
Img(Rummager) application implements a sum of low level features (color, texture and shape) which can be used for visual similarity image retrieval in a user friendly Graphical User Interface (**GUI**).

II. Technical Details

The Img(Rummager) application is programmed in C# and requires a Windows XP+ Operating System with a 3.0 .NET Framework. This is a portable application that does not require installation. The index files it creates can be stored in any part of the user's hard disk, or even on a local network. They are normal XML files where documents consisting of fields each have a name and a value. Img(Rummager) is available online along with documentation and video tutorials on http://www.img-rummager.com.

III. img(Rummager)

The application environment can be divided into 3 sections:

1) Image Laboratory Section
2) Image Retrieval Section
3) Extras Section

A. Image Laboratory Section

In this section of the application, the low level features used by the application for retrieval procedure are visually depicted.

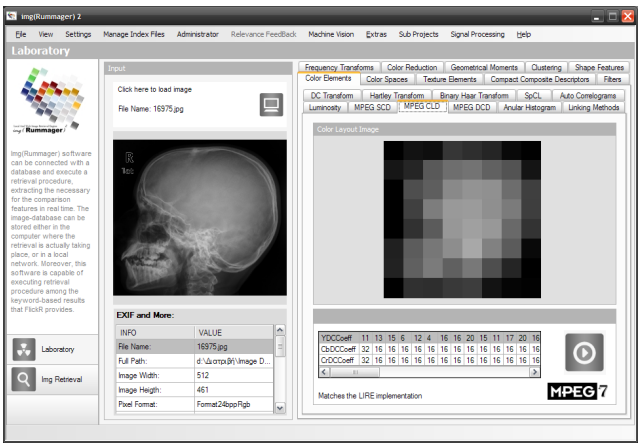

Figure 1. Image Laboratory Section

This section includes graphic representation of the low level features of the query image, and provides information extracted from them.

B. Image Retrieval Section

The application offers 3 different search options:

1) Search from XML index files
2) Search in local / network folder.
3) Search from the application server

978-0-7695-3765-8/09 $25.00 © 2009 IEEE
DOI 10.1109/SISAP.2009.16

1) Search from XML index files: The application supports the creation of XML files containing information from a group of images selected by the user. An XML index file can include information from 12 different descriptors[1].

Descriptor	XML Size (KB)	Extraction Time (sec)	Retrieval Time (sec)
CEDD [3]	210	29.30	0.56
FCTH [4]	258	52.35	0.61
Compact CEDD	131	26.71	0.54
Compact FCTH	151	50.86	0.58
JCD - ADS [5]	412	66.44	0.71 / 1.44
MPEG -7 SCD [6]	231	24.40	0.59
MPEG -7 CLD[6]	**103**	20.17	0.53
MPEG -7 EHD[6]	147	18.43	0.53
Fuzzy BTDH [7]	195	41.10	0.55
Tamura Dir.[8]	373	12.81	0.55
RGB Histograms	761	**11.70**	0.75
Correlograms [9]	27336	195.78	66.8
All Descriptors	30024	374.39	-

Table I
DESCRIPTORS CONTAINED IN THE IMG(RUMMAGER) AND THE INDEX FILE SIZE FOR THE WANG'S[10] DATABASE. TIME CALCULATIONS WERE MADE ON AN INTEL CORE 2 QUAD 2.8 GHZ, 2 GB RAM.

The user selects which descriptors to include for each file. The searching procedure is as follows: The user enters a query image and an index file. In the case that the index file contains information about more than one descriptor, the user must also select a descriptor. The application executes the search using the similarity matching technique recommended for each descriptor, as a similarity matching technique. When the procedure is complete, the application arranges the images contained in the index file according to their proximity to the query image, and presents the user with the results.

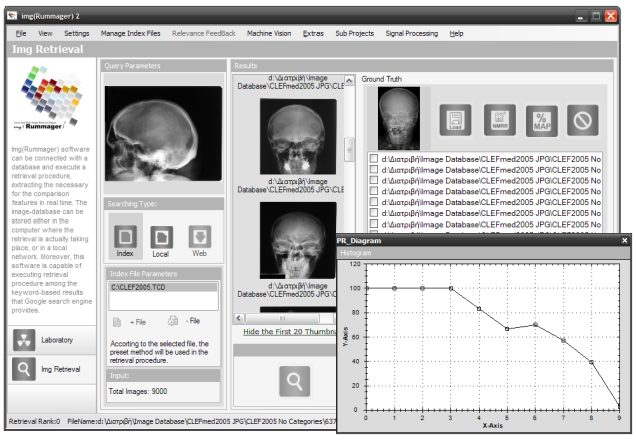

Figure 2. Screenshot from the XML file search and the Precision / Recall graph.

[1]The source code for the MPEG-7 Descriptors is a modification of the implementation that can be found in the LIRe[2] retrieval library.

The user then has the option of selecting those images matching the ground truth of the query image, and the application calculates the normalized modified retrieval rank(NMRR) [8] and/or the average precision (AP) for that query. This feature of img(Rummager) allows for easy retrieval evaluation to find the most appropriate descriptor for an arbitrary domain, where a ground truth has been defined.

An auto relevance feedback technique to tune the retrieval results is also provided to the user. The user selects one or more images, as being relevant to the initial retrieval expectations from the first round of retrieved images. After that the technique re-adjusts the initial retrieval results.

The query image does not need to belong to the index file. The user may use an image and the application extracts the descriptors for that image during the search process.

2) Search from Local / Network Folder: In this case, the user designates the query image and allocates one or more folders in which to execute the search. Next, one of the descriptors from Table 1 may be selected, or a custom retrieval method can be designated, through the selection of which low level features are to be used. These features include elements from the color histograms RGB, YIQ, HSL, the Direct Cosine Transform from an array of histograms, the Binary Harr Wavelet Transform from an array of histograms, Fuzzy Linking techniques, Moments, and others.

Every feature selected by the user comprises an independent unit. A permissible deviation limit at every unit of the system is designated. If the image is within limits, it will be led either to the outlet or to the next level. In every other case the image will be rejected.

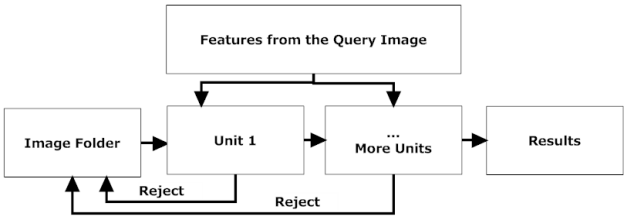

Figure 3. Flowchart of the customization process of image searching from image folder.

Upon completion of the process, all the images that successfully passed all units are presented to the user.

3) Search from the application server: Searching for images via the application server is differentiated from the two previous search methods, as apart from visual similarity, it also includes **keywords**. The server side part of the Img(Rummager) undertakes the execution of image retrieval based on **Flickr**[2] keywords (tags) and creates XML index files containing the descriptors of these images.

[2]http://www.flickr.com

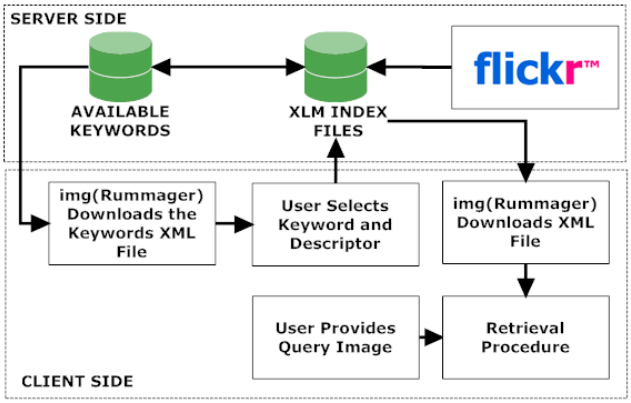

Figure 4. Searching from the application server.

When the user of the img(Rummager) client version selects the application server search, the img(Rummager) automatically connects to its server version and downloads an XML file with the available keywords. Thus far the application supports 100 keywords. The options appear in a drop down menu.
The user selects the desired keyword, preferred descriptor for the search, and imports a query image. The application downloads a second XML file containing the descriptors for 500 images tagged with the keyword specified by the user. The process that follows is similar to that described in section 3.2.1.

C. Extras Section

The img(Rummager) application integrates two additional applications: the img(Finder) which executes image retrieval through the **Facebook**[3] social network, and the img(Paint.Anaktisi) which performs an image search based on color rough sketches.
The img(Finder) connects to Facebook, using the user's personal account. Subsequently, it downloads an XML file from the Facebook server, which includes the addresses of the images that have been uploaded by the user's "friends". The application extracts the Joint Composite Descriptor for each image and creates a new XML index file. During the retrieval process, the user inserts an image in order to check whether this image has been used at Facebook.
The img(Paint.Anaktisi) uses the MPEG-7 Color Layout Descriptor to retrieve images using color rough sketches. Color Layout Descriptor descriptor effectively represents the spatial distribution of color of visual signals in a very compact form. The img (Paint.Anaktisi) uses the same XML files with img(Rummager) application.

[3]http://www.facebook.com

REFERENCES

[1] J. L. R. Datta, D. Joshi and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys*, vol. 40(2), pp. 1–60, January 2008.

[2] M. Lux and S. A. Chatzichristofis, "Lire: Lucene image retrieval - an extensible java cbir library," in *ACM International Conference on Multimedia 2008, ACM MM08.*, 2008, pp. 1085–1087.

[3] S. A. Chatzichristofis and Y. S. Boutalis, "Cedd: Color and edge directivity descriptor a compact descriptor for image indexing and retrieval." in *6th International Conference in advanced research on Computer Vision Systems ICVS 2008.*, 2008, pp. 312–322.

[4] S. A. Chatzichristofis and Y. Boutalis, "Fcth: Fuzzy color and texture histogram - a low level feature for accurate image retrieval," in *Proc. Ninth International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS '08*, 7–9 May 2008, pp. 191–196.

[5] S. A. Chatzichristofis, Y. S. Boutalis, and M. Lux, "Selection of the proper compact composite descriptor for improving content based image retrieval." in *The Sixth IASTED International Conference on Signal Processing, Pattern Recognition and Applications SPPRA 2009.*, 2009, pp. 134–140.

[6] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and texture descriptors," vol. 11, no. 6, pp. 703–715, June 2001.

[7] S. A. Chatzichristofis and Y. S. Boutalis, "Content based medical image indexing and retrieval using a fuzzy compact composite descriptor." in *The Sixth IASTED International Conference on Signal Processing, Pattern Recognition and Applications SPPRA 2009.*, 2009, pp. 1–6.

[8] S. M. H. Tamura and T. Yamawaki, "Textural features corresponding to visual perception." *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 8(6), pp. 460–472, 1978.

[9] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, "Image indexing using color correlograms," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 17–19 June 1997, pp. 762–768.

[10] W. G. Wang J Z, Li J, "Simplicity: Semantics- sensitive integrated matching for picture libraries," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23(9), pp. 947–963, 2001.