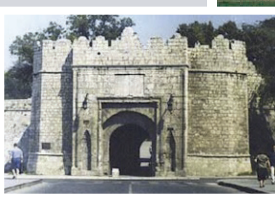
XLIII INTERNATIONAL SCIENTIFIC CONFERENCE ON INFORMATION, COMMUNICATION AND ENERGY SYSTEMS AND TECHNOLOGIES

ICEST

ISBN 978-86-85195-61-7

Welcome
Conference
Authors
Papers



Proceedings of Papers

Serbia, Niš, June 25 - 27, 2008
Faculty of Electronic Engineering

XLIII INTERNATIONAL SCIENTIFIC CONFERENCE ON INFORMATION, COMMUNICATION AND ENERGY SYSTEMS AND TECHNOLOGIES

ICEST 2008

Proceedings of Papers

Serbia, Niš, June 25 - 27, 2007

XLIII INTERNATIONAL SCIENTIFIC CONFERENCE ON INFORMATION, COMMUNICATION AND ENERGY SYSTEMS AND TECHNOLOGIES

ICEST 2008

organized by

Faculty of Electronic Engineering,
Niš, Serbia

Faculty of Telecommunications,
Sofia, Bulgaria

Faculty of Technical Sciences,
Bitola, Macedonia

under auspices of

- **Serbian Ministry of Science**
- **Serbian Ministry of Telecommunications and Information Society**

in cooperation with

- **Academy of Engineering Sciences of Serbia and Montenegro**
- **IEEE Serbia and Montenegro Section**
- **SCG IEEE Communications Chapter**
- **SCG IEEE MTT Chapter**
- **National Society for Telecommunications**
- **ETRAN Society**

CONFERENCE ORGANIZING COMMITTEE

Chairman:

D. Antić	University of Niš, Serbia

Members:

N. Acevski	University of Bitola, Macedonia
V. Česelkoska	University of Bitola, Macedonia
G. Iliev	Technical University of Sofia, Bulgaria
D. Janković	University of Niš, Serbia
V. Jović	University of Niš, Serbia
M. Kostov	University of Bitola, Macedonia
L. Lubih	Technical University of Sofia, Bulgaria
B. Milovanović	University of Niš, Serbia
D. Mitić	University of Niš, Serbia
Z. Perić	University of Niš, Serbia
Z. Stanković	University of Niš, Serbia
D. Tasić	University of Niš, Serbia

TECHNICAL SUPPORT

A. Atanasković	University of Niš, Serbia
Z. Đorđević	University of Niš, Serbia
T. Dimitrijević	University of Niš, Serbia
M. Miletić	University of Niš, Serbia
M. Milijić	University of Niš, Serbia
V. Minić	University of Niš, Serbia
M. Stanković	University of Niš, Serbia
B. Stošić	University of Niš, Serbia

CONFERENCE SECRETARIAT

Z. Stanković	**Conference Technical Editor**
B. Stošić	**Technical Coordinator**
M. Milijić	**General Technical Secretary**
T. Dimitrijević	**Technical Secretary**

Address:

ICEST 2008 Conference
Faculty of Electronic Engineering
Aleksandra Medvedeva 14
18000 Niš
Serbia

phone: +381 18 529 205, 529 303
fax: +381 18 588 399
e-mail: icest@elfak.ni.ac.yu

CONFERENCE INTERNET SITE

http://www.telsiks.org.yu

LIST OF ICEST 2008 REVIEWERS

Prof. Dr. Vladanka Aćimović-Raspopović
University of Belgrade, Serbia

Prof. Dr. Dragan Antić
University of Niš, Serbia

Mr. Aleksandar Atanasković
University of Niš, Serbia

Prof. Dr. Alexander Bekiarski
Technical University – Sofia, Bulgaria

Prof. Dr. Zoran Bojković
University of Belgrade, Serbia

Doc. Dr. Zlata Cvetković
University of Niš, Serbia

Doc. Dr. Dejan Ćirić
University of Niš, Serbia

Prof. Dr. Bratislav Danković
University of Niš, Serbia

Prof. Dr Dimiter Dimitrov
Technical University – Sofia, Bulgaria

Prof. Dr. Kalin Dimitrov
Technical University – Sofia, Bulgaria

Doc. Dr. Nebojša Dončov
University of Niš, Serbia

Prof. Dr. Goran S. Đorđević
University of Niš, Serbia

Doc. Dr. Goran T. Đorđević
University of Niš, Serbia

Mr. Predrag Eferica
University of Niš, Serbia

Prof. Dr. Christo Hinov
Technical University - Gabrovo, Bulgaria

Prof. Dr. Elissaveta Gadjeva
Technical University – Sofia, Bulgaria

Prof. Dr. Miodrag Gmitrović
IMTEL "Komunikacije" AD, Serbia

Prof. Dr. Nataša Gospić
University of Belgrade, Serbia

Prof. Dr. Vassil Guliashki
Institute of Information Technologies – BAS, Bulgaria

Prof. Dr. Ilia Iliev
Technical University – Sofia, Bulgaria

Dr. Predrag Ivaniš
University of Belgrade, Serbia

Prof. Dr. Dragan Janković
University of Niš, Serbia

Dr. Jugoslav Joković
University of Niš, Serbia

Dr. Aleksandra Jovanović
University of Niš, Serbia

Doc. Dr. Zoran Jovanović
University of Niš, Serbia

Prof. Dr. Milena Karova
Technical University – Varna, Bulgaria

Prof. Dr. Roumen Kountchev
Technical University – Sofia, Bulgaria

Prof. Dr. Milena Lazarova
Technical University – Sofia, Bulgaria

Doc. Dr. Nataša Maleš-Ilić
University of Niš, Serbia

Doc. Dr. Ivica Manić
University of Niš, Serbia

Doc. Dr. Dragan Mančić
University of Niš, Serbia

Dr. Zlatica Marinković
University of Niš, Serbia

Prof. Dr. Galia Marinova
Technical University – Sofia, Bulgaria

Prof. Dr. Vera Marković
University of Niš, Serbia

Prof. Dr. Georgy Mihov
Technical University – Sofia, Bulgaria

Prof. Dr. Mikho Mikhov
Technical University – Sofia, Bulgaria

Prof. Dr. Nebojša Milenković
University of Niš, Serbia

Doc. Dr. Dejan Milić
University of Niš, Serbia

Retrieving Images Using Content-Based Search and Progressive Wavelet Correlation

Igor Stojanovic[1], Sofija Bogdanova[2] and Momcilo Bogdanov[2]

Abstract **The following study looks at our experience in retrieving images from a database via a combination of content based search and progressive wavelet correlation. We distinguish two typical databases. One of them is comprised of images that are visually similar to the input; the other has dissimilar images. We examine ways of choosing the threshold value in either case. In our study we use Oracle database and IBM QBIC.**

Keywords – **Content/pixel based search, database, image retrieval, normalized correlation.**

I. Introduction

Capture, processing, storage and transmission of image data have been principally facilitated by easy Web hosting and the continuous increase of available computing power, along with the declining price of storage. Nowadays, it is essential to manipulate very large repositories of digital images. There is searchable image data characterized by miscellany of visual and semantic content, spanning geographically disparate locations. Many walks of life that by tradition depend on images for communication, for example engineering, architecture, medicine, and many others, can make good use of computerized imaging.

Modern commercial tools for retrieving images are descriptor-based. These tools are being improved and now take advantage of relevance feedback [1] and some aspects of image understanding. A typical example of this type of search engine is IBM QBIC (www.qbic.almaden.ibm.com) [2].

Pixel-based search, on the other hand, is a promising approach for applications requiring high resolution, such as satellite and medical images, especially in geoscience. It selects a template corresponding with a database of images, based on parameters such as whether the search should seek to match edges, shapes, color, texture, or other measurable relations between images. The normalized wavelet correlation coefficient [3] is a well known criterion which compares the variations within a region of a database image with the variations within the template in terms of correspondence. Elimination of sensitivity to uniform differences in brightness is a special advantage of this criterion.

Some of the work done in the field of content-based image retrieval is given in Section 2. The progressive wavelet correlation [3] is outlined in Section 3. Our proposal concerning an application of this method for searching images stored in a database is presented in section 4. Results of experiments that use a combination of content and pixel-based approaches are presented in Section 5.

II. Content-Based Image Retrieval

Content-based image retrieval (CBIR) aims at inventing techniques that support effective searching and browsing of large image digital libraries based on automatically derived image features. In the past decade, many general-purpose image retrieval systems have been developed, including QBIC, Photobook, Blobworld, Virage, VisualSEEK, WebSEEK, and others. A typical CBIR system views the query image and images in the database (target images) as a collection of features, and ranks the relevance between the query image and any target images in proportion to feature similarities.

Many research works have been published in the field of CBIR. However, no universally accepted model has yet been developed. The research concentrates on image segmentation based on low-level features like color, shape, texture and spatial relations. To find the semantic meanings or high-level meanings of an image, like whether it is the image of human beings or a bus or a train and so on, is still a problem. Attempts are being made to link low-level and high-level features. However, it is proving difficult for the very simple reason that there remains a vast gap between human perception and computer perception.

CBIR systems includes two main sub-systems: the server subsystem and the client subsystem (*Fig.1*). The server subsystem handles the processes of feature extraction, database indexing/filtering, feature matching and system learning. The client subsystem handles the process of querying.

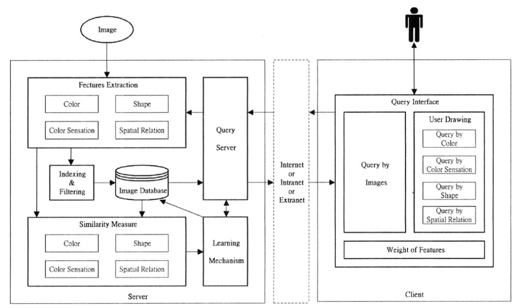


Fig. 1. System Architecture of CBIR system

In the query phase, the user can retrieve the images by giving the sample images or sketching the image via our

[1]Igor Stojanovic is with the Faculty of Computer Science, University Goce Delcev, Toso Arsov 14, 2000 Stip, Macedonia, E-mail: stojanovic.igor@gmail.com

[2]Sofija Bogdanova and Momcilo Bogdanov are with the Faculty of Electrical Engineering and Information Technologies, Karpos II b.b., P.O. Box 574, 1000 Skopje, Macedonia, E-mail: sofija@ feit.ukim.edu.mk, bogdanov@ feit.ukim.edu.mk

query interface that provides a lot of standard drawing tools. The procedures of image query are the following:

Step 1: User loads or sketches a query image, adjusts the weight of features and sends the query message.

Step 2: When the query server receives the query message, it will pass the image and weight of features to the feature extraction mechanism and the learning mechanism individually.

Step 3: After the extraction of features, the feature extraction mechanism will send the feature information of query image to the similarity measure mechanism and filtering mechanism.

Step 4: According to the feature information, filter mechanisms will retrieve the feature record of relevant images from the image database for detail similarity measure.

Step 5: The similarity measure mechanism will measure the similarity of features information between the query image and database images and return the results to the learning mechanism.

Step 6: The learning mechanism will rank the most similar database images to be the candidate images according to the similarity order that combines the weight and the similarity of features. Then, the learning mechanism will adjust the recommended weights of features according to the query image and candidate images. And, it will send the recommended weights to the query server and send the image IDs to the image database for returning the candidate images.

Step 7: Finally, the query server will send the recommended weights and candidate images to the query interface on the client side.

Step 8: When the user receives the candidate images via the query interface, he/her can pick the images that he/she wanted or choose some similar images for the next query.

In a new query with multiple images, the learning mechanism will rank the candidate images by the average of similarities between the candidate images and each of the query images. Then, the learning mechanism will adjust the recommended weights of features according to the query images and candidate images again. Repeating the above steps, the user will retrieve the desired images.

III. Progressive Wavelet Correlation

A. Overview

The basic ideas of the pixel-based search using progressive wavelet correlation include elimination in the earlier phases of searching, correlation in the frequency domain for fast computation, DCT in factorizing form, and wavelet representation of signals for efficient compression [3].

The algorithm can be described as follow:

Step 1: A candidate image is coarsely correlated with the pattern. Every eighth point of the correlation is generated.

Step 2: It is determined whether the pattern suitably matches the candidate image. If not, then another candidate image may be chosen or the search abandoned.

Step 3: If the match was suitable, then the candidate image is medium correlated with the pattern. We obtain the correlation at indices that are multiples of 4 mod 8 of the full correlation.

Step 4: Another similar match test is performed.

Step 5: A candidate image is fine correlated with the pattern. Fine correlation means to obtain the correlation at indices that are multiples of 2 mod 8 and 6 mod 8 of the full correlation.

Step 6: Another similar match test is performed.

Step 7: Full correlation: obtain the correlation at odd indices.

Step 8: If a suitable match is found for the fully correlated image, then the image searched for has been found.

B. Extension to two dimensions

Let the image size be N by N. In step 1, we have 64 subbands of length $N^2/64$. We perform one step of the inverse 2D JPEG transfer function, and one 2D step of the forward Fourier transform function. The next step includes adding the 64 subbands point by point to create a 2D array of size $N/8$ by $N/8$. Taking the inverse Fourier transform, we obtain the correlations at points that lie on a grid that is coarser than the original pixel grid by a factor of 8 in each dimension. In step 2, we obtain 16 subbands of size $N^2/16$ by adding the 16 subbands point by point, and taking the Fourier inverse. We will obtain the correlation values on a grid that is coarser than the original grid by a factor of 4 in each dimension. In step 3, we obtain 4 subbands of size $N^2/4$. Finally, in step 4, the full resolution is obtained.

Formulas for calculating normalized correlation coefficients that measure differences between images and patterns are given in [3]. Normalized correlation coefficients can be computed from the correlations described above. The normalization is very important because it allows for a threshold to be set. Such a threshold is independent of the encoding of the images.

The normalized correlation coefficient has a maximum absolute value of 1. Correlations that have absolute values above 0.9 are excellent, and almost always indicate a match found. Correlations of 0.7 are good matches. Correlations of 0.5 are usually fair or poor. Correlations of 0.3 or less are very poor. There is a tradeoff between the value of the threshold and the likelihood of finding a relevant match. Higher thresholds reduce the probability of finding something that is of interest, but they also reduce the probability of falsely matching something that is not of interest.

IV. Application of Pixel-Based Method for Searching in a Database

A. Image Store and Matlab Database Toolbox

The progressive wavelet correlation provides guidelines on how to locate an image in the image library. To make this method practical, we must first decide how to store the images. The initial choice is to store them in a disk file system. This can be seen as the quickest and simplest approach. A better alternative that should be considered is to store those images in a database. In the past five years, with changes in database technology and improvements in disk performance and storage, the rules have changed and it now makes business sense to use the database to store and manage all of an organization's digital assets. Databases offer several strengths over traditional file system storage, including manageability, security, backup/recovery, extensibility, and flexibility.

We use the Oracle Database for investigation purposes. There are two ways of storing an image into the Oracle Database. The first one is the use of Large Objects – LOB, and the second one is the use of Oracle *inter*Media.

To store images into the database we use the BLOB datatype. After creation of one BLOB column defined table we also create a PL/SQL package with loading of images procedure (load named) included. This procedure is used to store images into the database.

The implementation of the progressive wavelet correlation in Matlab and connection of the algorithm with the database are the next steps. The Database Toolbox is one of an extensive collection of toolboxes for use with Matlab. The Database Toolbox enables one to move data (both importing and exporting) between Matlab and popular relational databases. With the Database Toolbox, one can bring data from an existing database into Matlab, use any of the Matlab computational and analytic tools, and store the results back in the database or in another database.

Before the Database Toolbox is connected to a database, a data source must be set. A data source consists of data for the toolbox to access, and information about how to find the data, such as driver, directory, server, or network names. Instructions for setting up a data source depend on the type of database driver, ODBC or JDBC. For testing purposes JDBC drivers were usually used [4].

After setting up the data source for connecting to and importing data from a database we have used several standard functions of the Matlab Database Toolbox. We can retrieve BINARY or OTHER Java SQL data types. However, the data might require additional processing once retrieved. For example, data can be retrieved from a MAT-file or from an image file. Matlab cannot process these data types directly. One needs knowledge of the content and might need to massage the data in order to work with it in Matlab, such as stripping off leading entries added by the driver during data retrieval.

For the purpose of saving the extracted data into file `testfile`, we created the Matlab file `parsebin.m`. Using the `imread` function, we stored the file date into a two-dimensional output variable *x*.

In working with the Microsoft Access database, ODBC drivers are used for extracting data from the database. The extracting process must be adjusted by cutting the header created by the driver. Let *m* be quantity of bytes attached to the beginning of the data package by the ODBC driver. The quantity of bytes depends on the file type (file extension). To discover the value of *m* we created a procedure in Matlab. That procedure helps us to find the adequate value for *m* when linking with the image format. When working with Oracle databases and extracting data with JDBC and ODBC drivers there is no need for adjustment, so *m* has always the value "1" (*m*="1").

B. HTTP Application

The last step in the adaptation is to create Matlab applications that use the capabilities of the World Wide Web to send data to Matlab for computation and to display the results in a Web browser. The Matlab Web Server depends on TCP/IP networking for transmission of data between the client system and Matlab. In the simplest configuration, a Web browser runs on your client workstation, while Matlab, the Matlab Web Server (matlabserver), and the Web server daemon (httpd) run on another machine. In a more complex network, the Web server daemon can run on a separate machine [4].

The input mask of our application consists of three parameters: the image size *N*, the threshold *thr*, and the name of the image that we are looking for (*Fig.2*).

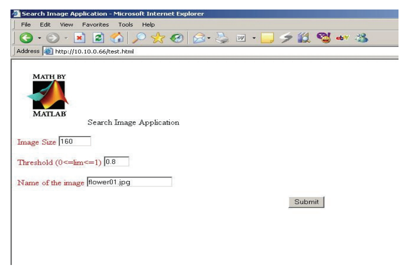


Fig. 2. Input mask.

The practical implementation of progressive wavelet correlation includes two main subsystems: the server subsystem and the client subsystem. The server subsystem handles the processes of image storing in a database and similarity measure. The client subsystem handles the process of querying. In addition, we present here the system architecture of the system (*Fig.3*).

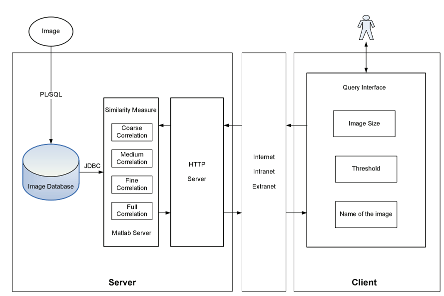


Fig. 3. System Architecture.

V. PIXEL-BASED SEARCH FOLLOWING CONTENT-BASED SEARCH

Pixel-based search using progressive wavelet correlation is too expensive computationally to apply to large collections of images, especially when it is possible to discover in advance that no match is likely. It has to be combined with descriptor-based search or some other means of reducing the search space. After descriptor matches narrow the search, pixel-based search can find matches based on detailed content.

Results obtained from a number of performed experiments by retrieving images from a database via application of both CBIR and progressive wavelet correlation led us to the idea of combining them in order to make the best use of their positive features. The initial idea was extended into a developed proposal for a new algorithm intended for searching and retrieving images from a database. The modular scheme of this algorithm is given in *Fig.4*.

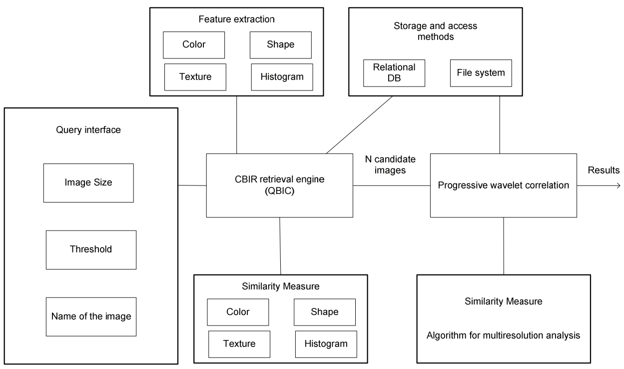


Fig. 4. A modular scheme for the proposed retrieval system.

Using QBIC, we established a database for the following characteristics of images: color, text, color histogram, and texture feature. We query the database for those images in the library with the most similar characteristics to the input image. On this set of candidates, we apply the normalized correlation coefficients to obtain the desired image. As an example, we show our work on locating the image 21.jpg in a database. Using QBIC, we isolated ten candidate images based on the Color Histogram Feature. After that these images are subjected to detailed pixel-based search based on the normalized correlation coefficients.

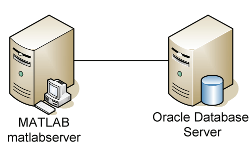


Fig. 5. Test environment.

For testing we used two PC on 3 GHz with 1 GB RAM (*Fig.5*). On one of them we installed Oracle Database Server version 10.1.0.2.0 and on the other we installed MATLAB version 7.0.4.365 (R14) Service Pack 2. We tested a different number of images stored in the database with constant value of threshold *thr*=0.8. The results of retrieval performance are shown in *Fig.6*.

Our experiments show that the retrieval performance is practically independent of the database capacity.

Investigating how the number of selected images depends on the number of images into database we obtain practically the same shape as shown in *Fig.6*.

Our experience suggests that the following minimum threshold values should be used when the two methods are combined:

- 0.5 for images not having visually similar images in the database (image 21.jpg);
- 0.7 for images having visually similar images in the database (image 10.jpg).

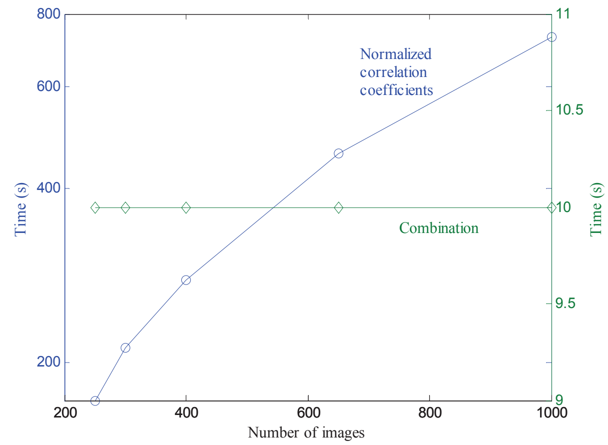


Fig. 6. Retrieval performance for different number of images.

By the use of the normalized correlation coefficients, the threshold value should be equal or higher than 0.7. An increased number of located images are produced as a result of increasing the number of images into the database by using the normalized correlation coefficients.

VI. Conclusion

The efficiency of joint content and pixel-based retrieval of images from a database can be improved in some applications such as satellite and medical images, by way of appropriate selection of the threshold value. Content-based image retrieval is fast, but it normally gives more than one image due to the vast differences in perception capacity between humans and computers. On the other hand, pixel-based retrieval using progressive wavelet correlation is impractical, due to the numerous operations per image it entails. However, a positive outcome should be expected if we combine the good features of content-based and pixel-based searches: speed and accuracy.

References

[1] Y. Rui et al., “Relevance feedback techniques in interactive content-based image retrieval,” in Storage and Retrieval for Image and Video Databases VI (Sethi, I. K. and Jain, R. C., eds.), *Proc. SPIE 3312*, , 25-36, 1997.

[2] M. Flickner et al., “Query by image and video content: The QBIC system,” *IEEE Comp.,* vol. 28, pp. 23-32, Sept.1995.

[3] H. S. Stone, “Progressive Wavelet Correlation Using Fourier Methods,” *IEEE Trans. Signal Processing,* vol. 47, pp. 97-107, Jan. 1999.

[4] I. Stojanovic, S. Bogdanova, and M. Bogdanov, “Content-Based Image Retrieving Improved by Pixel-Based Search,” *14th IWSSIP 2007* and *6th EC-SIPMCS 2007*, pp. 395-398, Slovenia, Maribor, 27-30 June, 2007.