UNIVERSITY OF CRAIOVA

FACULTY OF AUTOMATION, COMPUTERS AND ELECTRONICS

Experiments on Multimedia Databases (summary)

Implementation of a Multimedia Databases Management System for Databases with Images

Scientific Coordinator:

Prof.Ph.D.Eng. Mircea Petrescu

Ph.D Student:

asist.drd.ing. Cosmin Stoica Spahiu

Craiova 2010

From the Databases Management Servers existing on the market nowadays, only few of them offers support for multimedia content. Usually the multimedia content (images, video, audio, etc) is stored on the disk outside the database, or inside the database in BLOB data types. These solutions makes to be hard to implement content based retrieval operations.

There are two main retrieval techniques that are adopted by images search engines. The first technique was implemented since 1970 and was based on manual annotation of images with text. Although many commercial systems have adopted this technique, it has two main inconvenient:

- The images do not have always annotations. The manual annotation is time costing and cannot be applied always, especially when thousands of images are added daily.
- The annotation done by a person is not objective. It depends by the person that makes the annotation and can be different for different persons. More than that, the text tagging of images is sometimes insufficient, especially when the users are interested in small parts of the image that are hard to be described in words.

The second approach uses the image content for retrieval operations. The content can refer to color, texture, shape, etc. This technique is known as CBIR – Content Based Image Retrieval and was proposed in early 1990 once the artificial computer vision community started to develop.

Although the content-based retrieval was studied for decades it is still considered a challenge because of the well-known gap between low-level concepts and high-level semantic concepts.

The thesis is structured on six chapters:

- Chapter 1 – Introduction
- Chapter 2 – Similar implementation. State of the art.
- Chapter 3 – Personal research for content-based retrieval available methods. Methods. Algorithms.
- Chapter 4 – The Multimedia Database Management System
- Chapter 5 – Experiments. Results.
- Chapter 6 – Conclusions. Future research directions.

The thesis is started with an introduction where there are presented the thesis motivation, main subject and subjects that are studied. It is also presented a short overview with the structure and subjects discussed in each chapter.

The **Chapter 2** called **Similar implementations** makes a general overview if the Database Multimedia Servers existing nowadays on the market. They are studied from the point of view of images management.

There are presented Adaptive Server Anywhere, MySQL, Microsoft SQL Server and Oracle 10g. In this list only the solution proposed by ORACLE is complete, being able to

manage all kind of multimedia data, including DICOM files. However it is not easy all the time to be used because of the high costs involved in buying the server and developing the applications.

In the **Chapter 3,** called "Personal researches", it is studied the content based retrieval domain. There are described the color spaces, methods for translating one to another, methods for representing the images characteristics and defining the content based retrieval problems. There are also studied the files systems types available in the Microsoft Windows OS along with their advantages and disadvantages.

The studies have shown that there is no color space universally recognized as the best. This is due the fact that the color can be modeled and interpreted in different ways.

They have shown that only color two spaces can be used with success in content-based retrieval: HSV and l1l2l3 proposed by Gevers and Smeulders.

The HSV color space has the following characteristics:

- Is close to the human eye perception of colors
- Is intuitively
- Is invariant to light variations and camera direction

There were compared the results obtained using different color spaces: HSV, RGB, and CieLuv. There were considered medical images and images from nature. The HSV color space will generate the best results.

The HSV color space also has several problems:

- There is a non-linear color transformation from RGB to HSV
- It is device dependent
- The H component became instabile when S is closed to 0
- The H component is dependent of light color

Taking into account these results we decided to use the HSV color space quantized to 166 colors for the implemented system. To extract the color characteristics it is used the color histogram for quantized image.

The texture is also an important characteristic of the images, especially in the medicine domain where changes in texture and color can be manifestation of a disease. There were studied many methods for texture extraction. Among the most known studied methods we can enumerate:

Gabor filters method

Coocurrence matrix

Fourier Transform method

Markov-Gibbs

The studies have shown that each of them generates good results on different types of images. The tests made on medical images obtained the best results using Gabor filters and co-occurences matrices.

Taking these into account it was chosen the Gabor filter method for implementation into the system.

The **Chapter 4**, "Description of the Database Management System" describes the implementation. It is presented the way information is organized, the system functionality and the communication with the client applications.

The main contributions presented in this chapter are the implemented Databases Management system, the chosen solution for storing the images inside the database and the methods implemented for executing content-based retrieval queries and the tools implemented for this.

Along with the classical methods available in a DBMS, an element of originality for this system is the fact that the images are stored directly in the database along with the other relevant information defined in the table structure. The images are stored in the database in a special data type called IMAGE. This type is used to store all the information regarding the image: color characteristics, texture characteristics, wide, height, tall, image type, etc.

The next figure presents the general architecture of the system:

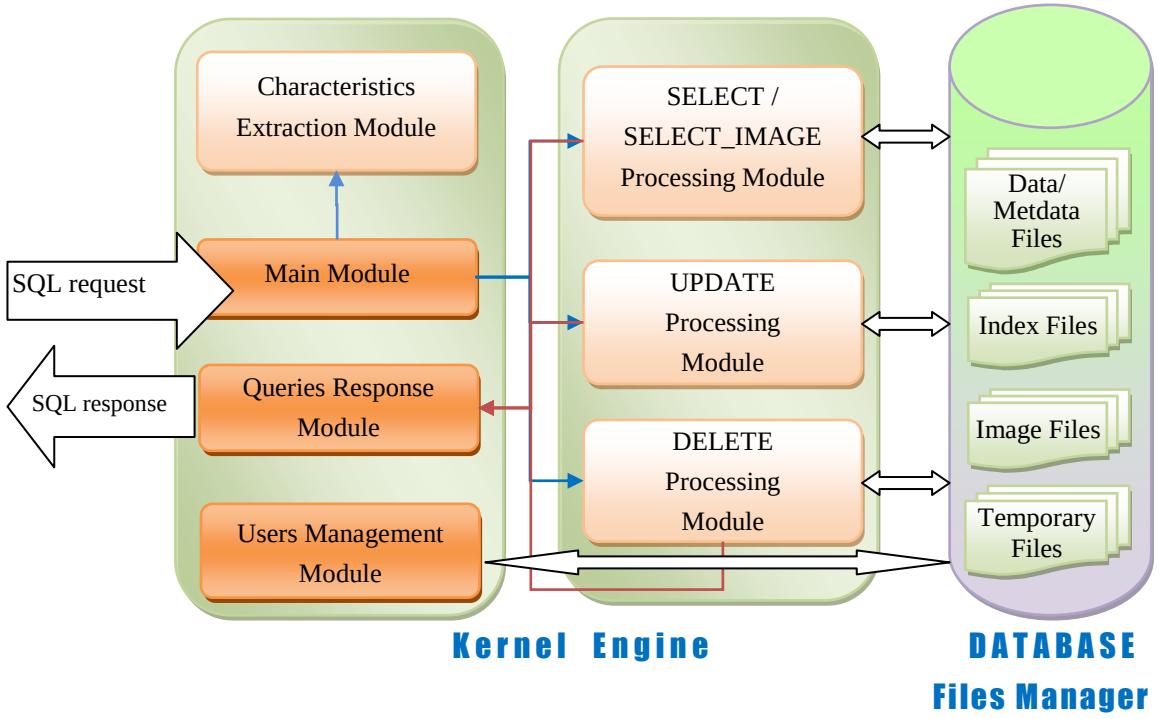

Figura 1: Arhitectura generală a MMDBMS

The first step is to establish a connection between client application and the server. All the commands and the results will use this channel for sending and receiving answers.

The server contains two main modules: the search engine (the KE|RNEL) and the files manager.

The kernel includes all the methods implemented for the server. It contains several sub-modules each of them with a well defined functionality.

The main module. It is the module, which manages all communications with the client. It is the one that receives all queries requests, check what is the type of query requested, extracts the parameters of the query and calls the specific module to execute it.

Queries response module. After the query is executed, the results will be sent to the Queries Response Module. It will compact the result using a standard format and then return it to the client. The client will receive it on the same communication channel used to send the request.

Select/Select/Image Processing module. If the main module concludes that is a SELECT SQL command, it will call the Select Processing module. This module extracts the parameters from the query and then search in the database files for specific information. If the query is a SELECT IMAGE query, it will use for comparison the similitude of characteristics instead equality of parameters.

Characteristics Extraction Module. When the main module receives a SELECT IMAGE or a UPDATE query which uses an image that is not already in the database it is needed first to process it. This module is called to extract the color and texture characteristics of the image. The data of the results will be used to initialize a variable of IMAGE data type.

Update Processing Module. When the query received from the user is an UPDATE command, it will be called to execute it.

Delete Processing Module. It is called when the user executes a DELETE command. The kernel executes only logic deletes. It never executes physical deletes. The physical deletes are executed only when a “Compact Database” command is sent by the user.

The second main module is the Database Files Manager. It is the only module that has access for reads and writes to the files in the database. It is his job to search for information in the files, to read and write into files and to manage locks over databases. When a client module request a read form a file it is enabled a read lock for the specific file (that represents a table in the database). All other read requests will be permitted but no writes will be allowed. If the client module request a write to file, it will be enabled a write lock. No other requests will be allowed until the lock is canceled.

The results will always be returned to the module, which made the request. The data read or wrote to files is not structured in any way. This module does not modify the structure in any way. All the results will be raw data, as it is read from the files or received from client modules.

The **Chapter 5**, called „**Experiments**" makes an analysis of the system's performances. It is tested the insertion speed (processing the images), retrieval speed and the quality of the retrieval.

The first set of tests measured the time needed for inserting data containing images. Because the images processing is perform when the images are inserted, it is important to know how the images size influence the execution speed.

The table 1 presents the time needed to extract the color and texture characteristics for different images resolutions.

No	Image resolution	Color characteristic	Texture characteristic	Total Time
1	160 ×160	0,45 s	14 s	14,45 s
2	240 × 240	0,60 s	14,5 s	15,10 s
3	320 × 320	0,70 s	15,30 s	16 s
4	480 × 480	1 s	63,70 s	64,70 s
5	640 × 640	1,50 s	108,50 s	110,50 s

We can notice a liniar growth of the time needed to process the images, when the image dimension is increasing.

The second set of tests measured the retrieval time when executing different types of queries, with different complexity. There were tested both simple text-based queries and complex content-based queries.

The last set of tests referred to quality of the retrieval for content based retrieval operations. There was calculated the precision and recall parameters in different scenarios. There were three cases taken into account: if it is used only color characteristic, only texture characteristic and both characteristics, each with a weight of 50%. For each case the maximum number of images returned was first 5, 9 and 15.

Table 1 presents the medium results obtained for these tests when there were used query images having different diagnosis.

The best results were obtained when there was used the color characteristic.

Table 1. Medium no. of relevant images returned by the system

Method:	Medium no. of relevant images returned by the system when retrieving:		
	5 images	**9 images**	**15 images**
Color characteristic	4	6	8

Texture characteristic	3	3	6
Mixt	4	7	9

The results were poorer when there was used only the texture characteristic. This is due to the images type used. For the diseases taken into account, the change in tissue's color is more obvious than the changes in texture. It is possible for the results to be better for other types of medical images, from other domain when taking into account the texture characteristic.

For example, the results were better than the average when using only texture characteristic, for images with rectocolite and esofagitis diagnosis.

Taking into account the above results it is recommended, if no other special reasons, to use both characteristics for the content-based retrieval. This way, the images that are not retrieved by one method it is highly possible to be retrieved by the other method.

The last chapter presents the **Conclusions and Future developments** and the main contributions of the thesis.

The main contribution is the implementation of a system for multimedia databases management that includes algorithms for automate images processing and extraction o color and texture characteristics. These will be stored in a new special data type called IMAGE.

The list of the other contributions is:

- There were studied the main databases servers existing on the market nowadays and how they manage the multimedia files and execute content-based retrieval operations.
- There were studied different color spaces and presented the advantages and disadvantages of using each of them and how can be translated one to another.
- There were studied the methods available for extracting color characteristics (color histograms) and enhancing the images (histograms equalizations).
- There were tested the main images quantization algorithms and presented the comparative results after using each of them
- There were studied the algorithms used for extracting texture characteristics. It was noticed that the algorithm using Gabor filters generated the best results for the images types used for tests.
- There were studied different methods used for computing the similitude. The histograms intersection was chosen for computing the color similitude and the Gabor filters method was chosen for computing texture similitude during implementation.
- It was developed a software system to manage multimedia databases using SQL language. As originality elements can be mentioned the Image data type used for storing the characteristics extracted from images, and integration of methods needed for characteristics extraction and executing content-based retrieval operations.

Nowadays the Oracle 10g/11g database server is one of the few existing servers that offers something similar, but to a much larger scale.

- There were made a series of experiments using the implemented software. There were tested both execution time for different types of operations and the retrieval quality, using recall and precision parameters.

The future development directions are referring especially to enhance the execution time and the results by adding new methods.

Another chapter that can be improuved is the one referring to data security. It is recommended here to use the trazactional NTFS file system.

The results obtained in this thesis open many research perspectives for the multimedia and content-based retrieval domains. We consider these future research directions to be important and should be deeply analyzed.