A Multimedia Database Server for Content-Based Retrieval and Image Mining

Cosmin Stoica Spahiu, Liana Stanescu and Dumitru Dan Burdescu

University of Craiova, Faculty of Automation, Computers and Electronics, Craiova, Romania

Abstract. This article presents a possible extension of a software tool implemented in C++ that manages multimedia data collections from medical domain. An element of originality for this database management system is that it includes a series of algorithms used for extracting visual information from images (texture and color characteristics) along with classical operations needed for databases servers. It is also presented a data mining algorithm adapted to the database system that will be included in a future version.

1 Introduction

The images are an important class of multimedia data. The WWW is one of the biggest multimedia repositories, including text data, images, video and audio data. Most of the data type exchanged in real world is images. Although there have been made efforts for developing search engines, content-based retrieval is rarely implemented.

The raw data is not always useful. The real advantage is when data mining techniques can be applied and obtained knowledge. That is why first step is to adapt the techniques used for images processing.

In order to make the mining technology to be successful, it should be developed for other types of data, especially for images. The image mining should consider automatic classification, knowledge extraction, connections between images and other new patterns. The extension of the data mining in the imagistic field is a natural extension. It is an interdisciplinary domain that includes artificial vision, patterns recognition, data mining, automatic learning, databases and artificial intelligence.

More than that, image mining must consider spatial information. The same pattern might have several interpretations. That is why the mining algorithms for images are different than the classical ones. An image pattern must be represented in a suggestive form to the users using some characteristics of images. The information can be presented at different levels of details: pixel, object, semantic concept, or pattern.

Most of the data mining activities have been made based on the similarity analysis between a query image and the images from database. There are two categories of image retrieval techniques: systems that use a text descriptor of the image and systems that use visual content.

In the first category the images are described based of a text defined by the user. They are indexed and retrieved based on basic descriptors such as: image size tags, image type, acquisition date, owner id, keywords, etc. The types of queries that can be

Spahiu C., Stanescu L. and Burdescu D.
A Multimedia Database Server for Content-Based Retrieval and Image Mining.
DOI: 10.5220/0004466001410148
In *Proceedings of the 4th International Workshop on Enterprise Systems and Technology (I-WEST 2010)*, pages 141-148
ISBN: 978-989-8425-44-7

executed are: find images from database that match the following criteria: date of capture (before 2010), size > 150 KB, and tag “clouds”.

The text descriptors are usually added by a human operator since the automatic generation is hard to be done without incorporating visual information. It is a process that is hard to be applied nowadays since the volume of information is high. More than that, these descriptors are subjective and depend on the users’ perspectives.

When using the second category, the queries that can be executed follow the next pattern: “find all the images that are most similar with the query image”.

The paper is structured as follows: Section 2 presents content-based retrieval systems. Section 3 presents an overview of the server, Section 4 presents the image data mining functionality and Section 5 presents the conclusions.

2 Content-based Retrieval

The Most important aspect in content based retrieval is to find a method to measure the similitude of the images. The properties needed for methods used to compute the similitude, are:

- Easy computing
- Correspondence with human reasoning

There are two ways to implement content based retrieval: content-based retrieval for k-nearest vicinity (retrieves the most k similar images) and domain query (retrieves all the images that have the similarity between specified ranges). In order to make fast comparisons of the images, the system has to process off-line the images and to extract and store the characteristics of the image. For example, the color histograms describe the colors distribution and they are extracted before executing any content based query.

The dissimilitude of two images must be a metric that generate small values for similar images and large value for images that have only few in common. The performance of such a system is limited by the quality of the images characteristics [4][8].

The architecture of a content based retrieval system is presented in Figure 1.

a) The content-based region query: this type of query compares the image based on their color regions. In the first step of the query, the images’ regions are being queried instead of whole images. The total similitude of two images is computed based of the distances computed for each region in part.

The content-based retrieval can be improved in quality by adding spatial information to the query. In this case it is considered both the similitude distances of the characteristics (texture, color) and the spatial values of the regions.

b) Spatial query of the images. In the last years there have been developed techniques for spatial indexing that permits to retrieve objects based on the objects positioning. These researches compare images where there have been already defined regions or objects, as in Figure 2 [7].

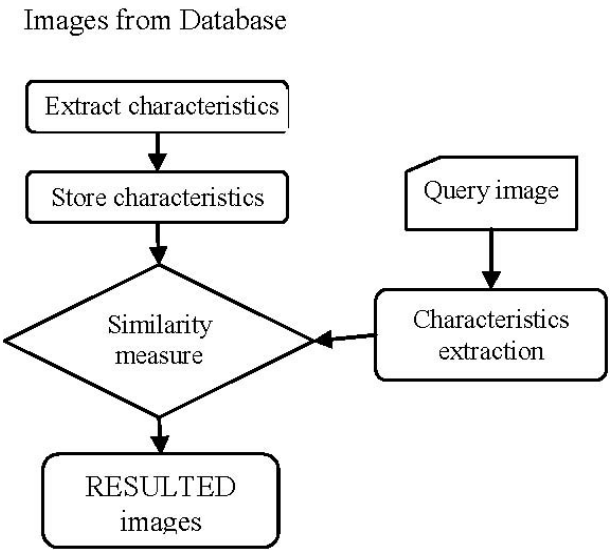
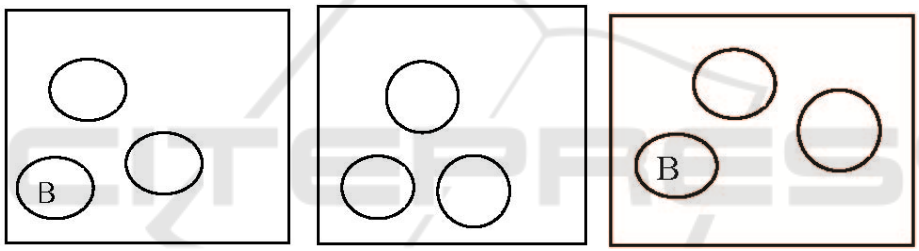


Fig. 1. Architecture of the Content-based retrieval system.



a) Query imag b) A possible relative mach c) An absolute match

Fig. 2. Spatial query of the images.

In order to extract the color regions of the images there have been implemented the Apriori algorithm proposed by Smith and Chang. For each color region that have been detected it should be stored the following information:

- Image id
- Region id
- Color set of the image
- Coordinates of the regions

All these information will be used in the content-based image query process.

The spatial information is represented by the minimum rectangle that bounds the region.

The characteristics that have been used in the proposed system are: color histogram and texture (extracted using the Gabor method) [9][10].

The similitude between images is computed using histogram intersection and quadratic distance.

3 MMDBMS Overview

The MMDBMS that have been implemented allows database creation, table and constraints adding (primary key, foreign keys), inserting images and alphanumerical information, simple text based query and content-based query using color and texture characteristics. The software tool is easy to be used because it respects the SQL standard. It does not need advanced informatics knowledge and has the advantage of low cost. It is a good alternative for a classical database management system (MS Access, MS SQL Server, Oracle10g Server and Intermedia), which would need higher costs for database server and for designing applications for content-based retrieval.

Figure 3 presents the general architecture of the MMDBMS [1][2][3].

In the first step any application that uses the server must connect to the database. This way it will be created a communication channel between them. All commands and responses will use this channel to send queries requests and receive answers.

The server has two main modules: kernel engine and database files manager.

The kernel engine includes all functions implemented in the server. It is composed from several sub-modules each of them with specific tasks[1][2]:

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 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.

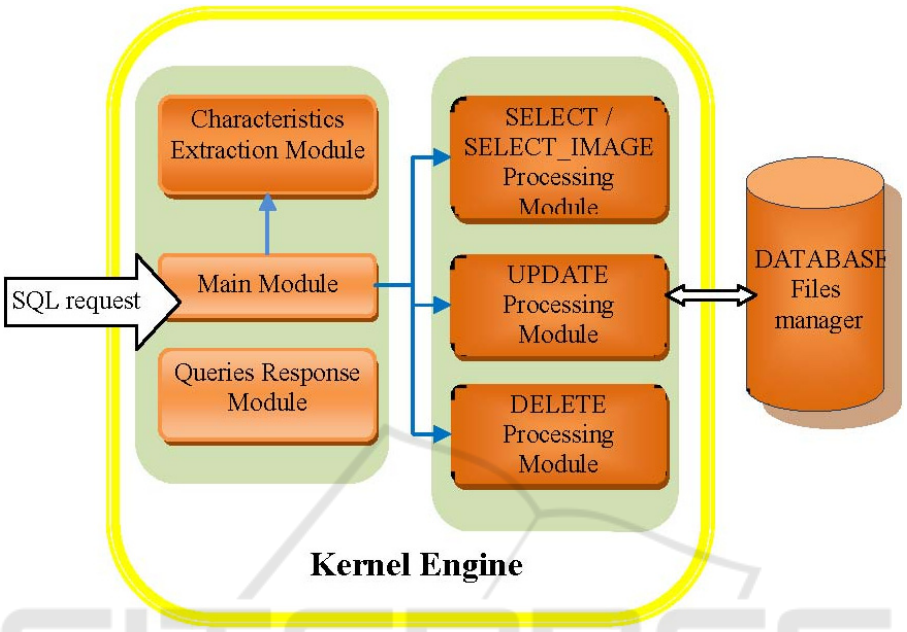


Fig. 3 General architecture of the system.

4 Image Data Mining

The information in raw form is not always useful. The real benefit comes when "interesting" patterns can be obtained based on the association rules. An association rule tells us about connections between two or several objects. It is a rule of type A => B, where A and B are objects satisfying the condition $A \cap B = \Phi$.

In order to find specific combinations for objects that appear together in different associations we have studied the Apriori algorithm. It will be implemented in a future version of the system.

This algorithm will select the most "interesting" rule, based on two parameters called support and confidence. The rule A=> B shows that anytime A appears in a transaction, it is very possible to appear B also.

The support parameter shows the statistic meaning of a rule:

$$supp(A \Rightarrow B) = \frac{|\{T \in D | A \subseteq T \wedge B \subseteq T\}|}{|D|}$$

The confidence parameter shows the strength of a rule:

$$conf(A \Rightarrow B) = \frac{|\{T \in D | A \subseteq T \wedge B \subseteq T\}|}{|\{T \in D | A \subseteq T\}|}$$

The rule probability confidence is defined as conditional probability:

$$p(B \subseteq T \mid A \subseteq T).$$

The association rule can be also between two or several objects (A,B => C) where A, B, C $\subseteq$ U.

The association rule is stronger as the confidence parameter is higher. This last parameter specifies the minimum support for frequent objects. All the subsets of frequent objects are also frequent. An object can be frequent only if it is found to be frequent in one of the algorithm's steps.
The Apriori algorithm is presented next.

Algorithm 1 Apriori Algorithm

1. **Find frequent item sets;**
$F_1 = \{u_i | \, \|u_i\| > \text{minimum support}\}$
for $(K = 2;\ F_{K-1} \neq \emptyset;\ K++)$ **do**
$C_K = \{c_k | c^{(a)} \wedge c^{(b)} \in F_{K-1}\}$, where:
$c_k = (u_{i_1}, ..., u_{i_{k-2}}, u_{i_{k-1}}, u_{i_k})$
$c^{(a)} = (u_{i_1}, ..., u_{i_{k-2}}, u_{i_{k-1}})$
$c^{(b)} = (u_{i_1}, ..., u_{i_{k-2}}, u_{i_k})$
$\|c_k\| = 0;$
for $(\forall T, T \subseteq D)$ **and** $(\forall c_k, c_k \in C_K)$ **do**
if $(c_k \in T)$ **then**
$\|c_k\| = \|c_k\| + 1;$
end if
end for
$F_K = \{c_k | \|c_k\| > \text{minimum support}\}$
end for
$F = \bigcup_K F_K$
2. **Use the frequent itemsets to generate strong association rules.**

This algorithm will be implemented in order to find different patterns used for automate classification of the images, based on the diagnosis. The system will be able to find which part of the extracted features is characteristic for each disease. It will also be able to say for example which characteristics are connected (used in early diagnosis for some diaseases).

Because the data volume is higher and higher in the last years it is important to find efficient algorithms for data mining. The presented algorithm scans the data for few times depending to the biggest most frequent object. New enhancements can be added by reducing the number of database parsing and the number of candidates that were generated.

The version of Apriori algorithm that is based on partitions needs only two parsing of the database. The database is divided in disjoint partitions, each of them small enough to fit the memory.

During the first scan, the algorithm reads each partition and finds the most frequent local objects. During the second parsing the algorithm computes the support for each frequent local object, from the entire database. If one object is frequent in the database it must be frequent at least in one of the partitions. That is why to the second partition there are found the supersets with all the potential frequent sets of objects.

5 Conclusions

The paper presents a possible extension of a software tool implemented in C++ that manages multimedia data collections from medical domain. An element of originality for this database management system is that along with classical operations for databases, it includes a series of algorithms used for extracting visual information from images (texture and color characteristics). It is also presented a data mining algorithm adapted to the database system that will be included in a future version.

It is created for managing and querying medium sized personal digital collections that contain both alphanumerical information and digital images (for examples the ones used in private medical consulting rooms). The software tool allows creating and deleting databases, creating and deleting tables in databases, updating data in tables and querying. The user can use several types of data as integer, char, double and image. There are also implemented the two constraints used in relational model: primary key and referential integrity.

The advantages of using this intelligent content-based query visual interface are that the specialist can see images from the medical database that are similar with the query image taking into consideration the color and texture characteristics. In this way the specialist can establish a correct medical diagnosis based on imagistic investigation frequently used nowadays.

The system will include a data mining module that will be used for automate classification of images and finding "interesting" patterns between characteristics of the images.

This software can be extended in the following directions:

Adding new types of traditional and multimedia data types (for example video type or DICOM type - because the main area where this multimedia DBMS is used it is the medical domain and the DICOM type of data is for storing alphanumerical information and images existing in a standard DICOM file provided by a medical device)

Studying and implementing new algorithms for data mining that performs faster on large image collections.

References

1. Stoica Spahiu C.: A Multimedia Database Server for information storage and querying. In: Proceedings of 2nd International Symposium on Multimedia – Applications and Processing (MMAP'09), Vol. 4, (2009), pp. 517 – 522.
2. Stoica Spahiu C., Stanescu L, Burdescu D. D., Brezovan M.: File Storage for a Multimedia Database Server for Image Retrieval. In: Proceedings of The Fourth International Multi-Conference on Computing in the Global Information Technology, (2009) pp.35-40
3. Stoica Spahiu C., Mihaescu C., Stanescu L., Burdescu D. D., Brezovan M.: Database Kernel for Image Retrieval. In: Proceedings of The First International Conference on Advances in Multimedia, (2009), pp. 169-173
4. Kratochvil M.: The Move to Store Images In the Database (2005). http://www.oracle.com/technology/products/intermedia/pdf/why_images_in_database.pdf

5. Del Bimbo A.: Visual Information Retrieval, Morgan Kaufmann Publishers. San Francisco USA (2001)
6. Smith J. R.: Integrated Spatial and Feature Image Systems: Retrieval, Compression and Analysis. PhD. thesis, Graduate School of Arts and Sciences. Columbia University (1997)
7. Gevers T.: Image Search Engines: An Overview. Emerging Topics in Computer Vision. Prentice Hall (2004)
8. Stanescu L., Burdescu D. D., Brezovan M., Stoica Spahiu C., Ion A.: A New Software Tool For Managing and Querying the Personal Medical Digital Imagery. In: Proceedings of the International Conference on Health Informatics, Porto – Portugal: (2009) 199-204
9. Samuel J. Query by example (QBE). http://pages.cs.wisc.edu/~dbbook/openAccess/thirdEdition/qbe.pdf
10. Query by Example, http://en.wikipedia.org/wiki/Query_by_Example
11. Muller H., Michoux N., Bandon D., Geissbuhler A.: A Review of Content_based Image Retrieval Systems in Medical Application – Clinical Benefits and Future Directions. Int J Med Inform 73 (2004)
12. Khoshafian S., Baker A. B.: Multimedia and Imaging Databases. Morgan Kaufmann Publishers, Inc. San Francisco California (1996)