A .Net Implementation of a Content-Based Image Search Component

Nádia Puchalski Kozievitch[1], Tiago Falcão[1], Ricardo da Silva Torres[1]

[1]Instituto de Computação – Universidade Estadual de Campinas (Unicamp)
CEP 13081-970 – Campinas – SP – Brasil

{nadiapk,rtorres}@ic.unicamp.br, tiagofalcao@gmail.com

Abstract. *This work describes a .Net implementation of a content-based image search component, used for deployment and assessment of an image annotation and retrieval tool. It has been designed in a flexible and portable way, under the technology perspective, where different descriptors can be easily configured for different contexts.*

1. Introduction

Advances in hardware for image acquisition and storage of image technology have enabled the dissemination of large image datasets in several applications, like remote sensing, biodiversity and biometry. In order to deal with these huge amount of data, it is necessary to find ways to perform an efficient and reliable image retrieval.

Traditionally, the first approach used for image retrieval is the one based on image textual metadata, using keywords. The annotation is defined by users and it is usually inefficient because it is not done in a systematic way (for example, similar words are not used as previous annotated). Another issue is that different users (or even the same user at a different moment) might use different words for the same visual characteristic. This lack of systematization in the annotation process decreases the performance of the keyword-based search.

These shortcomings have been addressed by the so-called Content-Based Image Retrieval (CBIR) systems. Basically, these systems extract feature vectors that represent image properties such as color, texture and shape, using automatic image processing algorithms.

In this paper we describe a .Net implementation of a content-based image search component. This component is a Dynamic-Link Library (DLL) written in C#, using different types of descriptors, resulting in a flexible and portable application under the technology perspective. It can be used for assisting users in creating complex applications that consider tasks related to managing, searching, annotating and combining image features.

2. Related Work

2.1. CBIR

Content-Based Image Retrieval (CBIR) systems aim to help and retrieve images based on their visual properties. This solution relies on using algorithms to extract feature vectors that represent image properties such as color, texture and shape. One main advantage is the possibility of an automatic retrieval process, contrasting with the manually effort to annotate images in textual searches. CBIR systems are based on the main idea of similarity search: the user needs to retrieve images that are relevant and similar to a query pattern, usually an image. The comparison of two images is handled with descriptors, that may be characterized by [Torres et al. 2006b] (i) an extraction algorithm to encode image features into feature vectors; and (ii) a similarity measure to compare two images (for example, a distance function between the feature vectors). In content-based image retrieval domain, a specific descriptor is better than a second one if a higher number of returned images are relevant, compared with the source image previously used. During the retrieval process, the user may select the most relevant feature (color or shape, for example) for a contextual domain.

CBIR has some particular main issues when designing their system: definition of appropriate image, feature vectors representation and indexing, descriptors and interaction mechanisms among others [Torres et al. 2006b]. Basically these systems try to retrieve similar information, using a pattern (e.g., shape sketch or image example).

2.2. CBISC

The Content-Based Image Search Component (CBISC) is a component created to handle queries based on an image content. It supports collections of image information as a CBIR system. It retrieves images similar to a user-defined pattern (e.g., image sketch, color of an image, etc.) based on content properties (e.g., texture or color), which are often encoded in terms of image descriptors [Murthy et al. 2006].

CBISC encapsulates multidimensional index structures to speed up the search process. Furthermore, different image descriptors like metric and non-metric; color, texture and shape; with 1D or 2D feature vectors can be combined to improve effectiveness, allowing progressive extension of the descriptor base [Torres et al. 2004].

2.3. Use of CBISC

In archeological collections domain, CBISC contributed for searching on digital library objects based on image context [Vemuri et al. 2006]. The heterogeneous information can range from different sources, like image data, geo-spatial information, chronological data and other relevant metadata.

The Biodiversity Information Systems (BISs) involve all kinds of heterogeneous data, including ecological and geographical features, but their management have very limited support yet. In [Torres et al. 2006a], CBISC was used on an OAI-based generic digital library architecture for integrated management of image descriptors and textual information, dealing with fish specimen identification. Experimental results suggested that this approach improved the effectiveness of the fish identification process, when compared to the traditional key-based method.

In [Murthy et al. 2006], CBISC supported SIERRA, an application that combined text-based and content-based image retrieval, allowing users to link together image content with related data like annotations. To achieve this, the concept of Superimposed Information (SI) was used. SI refers to a new information laid over information [Murthy 2007]. In this scenario, a user may work with several sources, including books, papers, images, descriptions and audio, where only selected and relevant information is extracted. It can be seen as a second layer where additional data is created to reference, highlight and present information.

3. Prototype

As pointed out before, a typical CBIR solution requires the construction of image descriptors, which are characterized by (i) an *extraction algorithm* to encode image features into a feature vector*;* and (ii) a *similarity measure* to compare two images based on the distance between the corresponding feature vectors.

The CBISC starts processing a query by extracting a feature vector from the query image. This extraction process requires validating the proposed query against the CBISC configuration file and searching for the appropriate *Extraction Algorithm*. The validation process involves checking the input query parameters accordingly to the CBISC configuration. Next, the query image feature vector is used to rank the database images according to their similarity to the query image. This step relies on either a *Distance Computation Algorithm* taking into account the feature vectors of all images in the database, or using an appropriate index structure. Images are indexed in the CBISC according to their feature vectors by using the M-tree index structure to speed up retrieval and distance computation. Finally, the most similar database images are ranked and the CBISC returns a result containing this ranked list.

The prototype is now comprised of three modules: Manager, XML Database and Descriptors, implemented by classes manager, db and descriptor, as illustrated in Figure 1. All modules are integrated enabling full communication between the components.

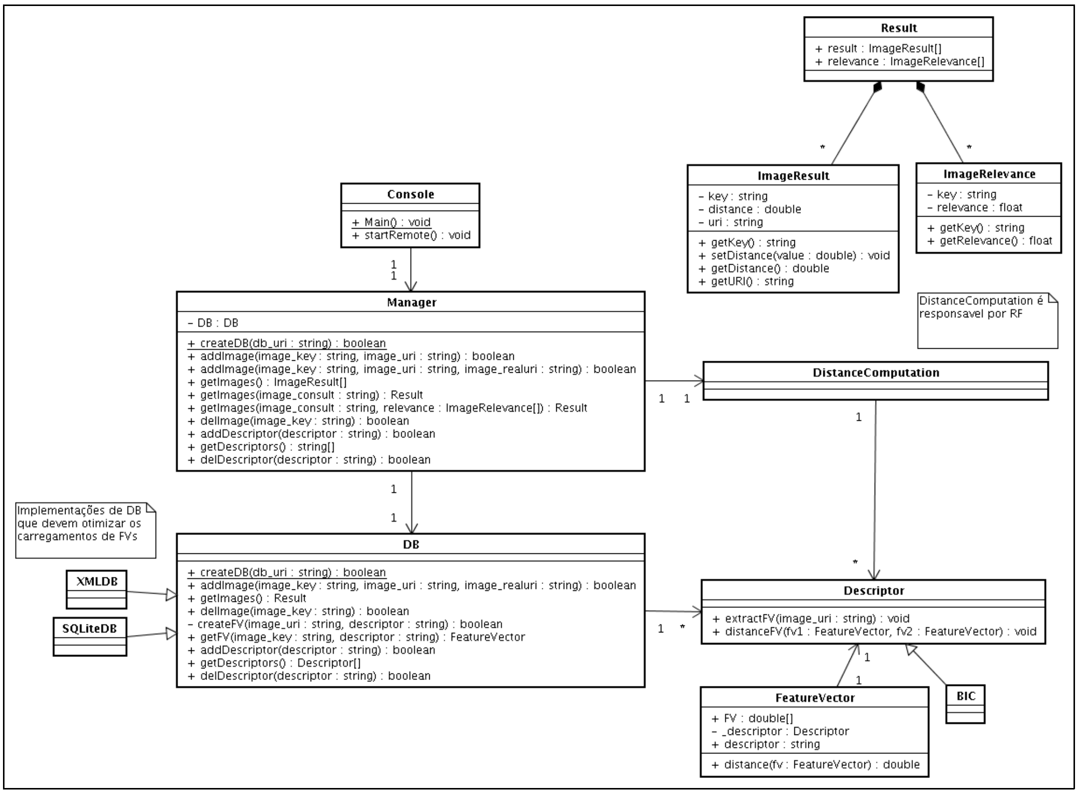

Figure 1. Class Diagram

The manager module manages requisitions and converts data. XML database manages the data in XML with image metadata, feature vectors and image regions. The descriptor module manages the descriptor library (DLLs), extracts feature vectors and calculates distances. The DLL library was written in .Net C# language. The component receives an image as an input and returns vector objects as output, containing information like image data and distance.

Different applications and images might have different image descriptors for a better effectiveness. The best image descriptor should be applied in all images, resulting in a set of feature vectors. Our component enables the installation of different descriptors, but for the tests presented, the BIC (Border/Interior pixel Classification) descriptor was used. BIC has three main components: (1) a simple and powerful image analysis algorithm that classifies image pixels as either border or interior, (2) a logarithmic distance (dLog) for comparing histograms (mainly responsable for the feature vectors comparison) and (3) a compact representation for the visual features extracted from images.

4. Case Study

The component is used in collaborating projects with the Department of Computer Science of Virginia Tech, USA. The main objective of this project is deploy and access a Tablet-PC image annotation and retrieval tool that will help scholars work with images and part of images, associate them with multimedia information land, and later, retrieve information based on text-and-content-based retrieval techniques. The main objective is to increase the educational value of a Tablet-PC by enhancing and enabling many different courses involving use of images and annotations. The evaluation of this component will take place in a field setting, involving observation and identification of fish species using an identification key.

Figure 2 shows the high-level architecture design of the Tablet-PC project. It consists of an interface, advanced digital-libraries services/knowledge management, basic digital-libraries services, information processing and data repositories. The design is such that other existing modules may be plugged in to facilitate richer image description and retrieval.

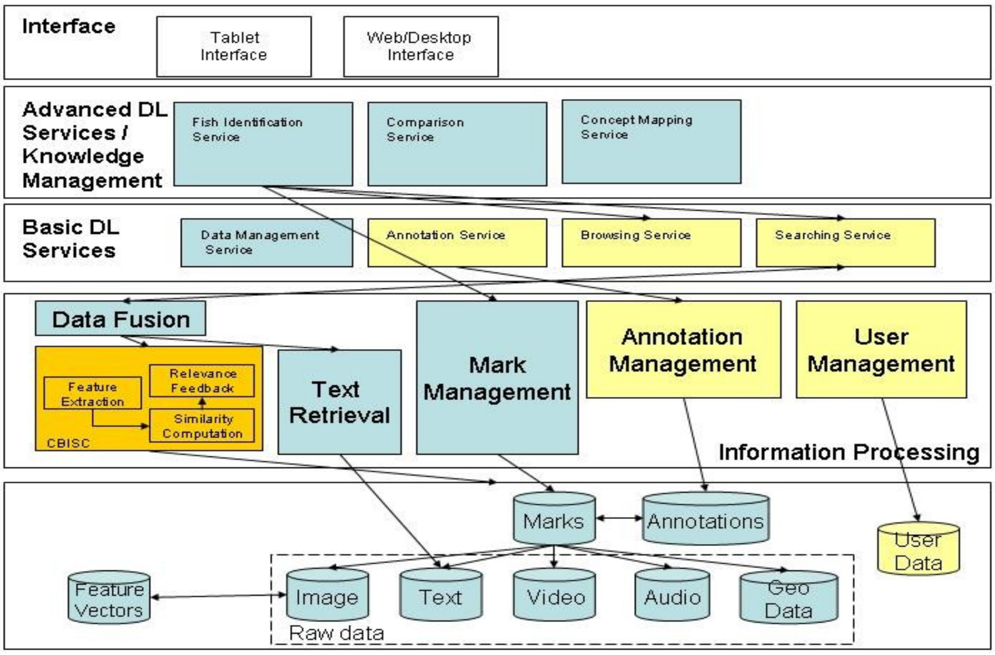

Figure 2. The Tablet-PC project architecture

Users can define queries using an image or just a part of it. In Figure 3, a user selects a fish specie image and query the application for other images that have the same characteristics (descriptor similarity search retrieval). Another approach occurs when the user selects one region at the fish image (head) and searches for other images that are similar, as demonstrates in Figure 4. Figure 5 illustrates a user selecting a fish image and applying a specific annotation related to a region. This information will be used for retrieving context-based information.

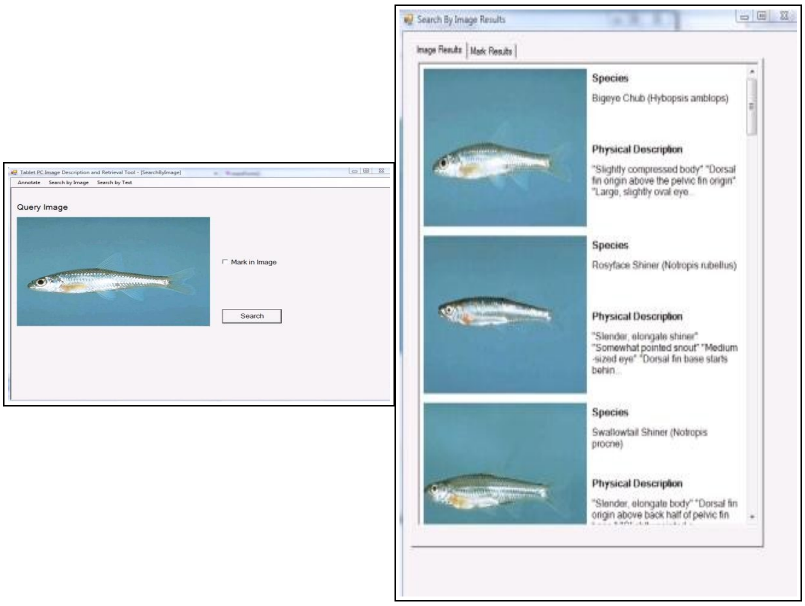

Figure 3. The descriptor similarity search retrieval

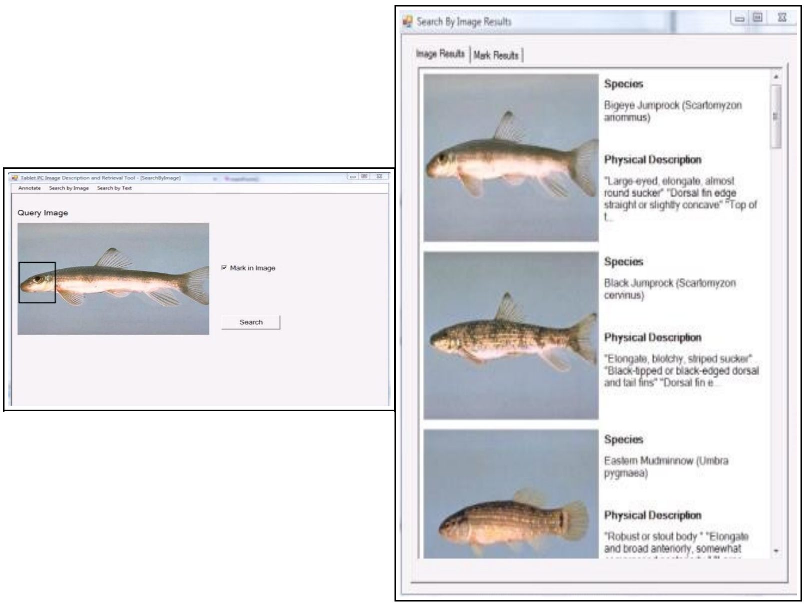

Figure 4. The search retrieval for a specific part of an image

Further tests will be done on parasitology domain at the Biology Institute in State University of Campinas (Unicamp). This study will be similar to the mentioned above, except that it will be on a different domain, involving data of parasite species, using different sizes and types of images.

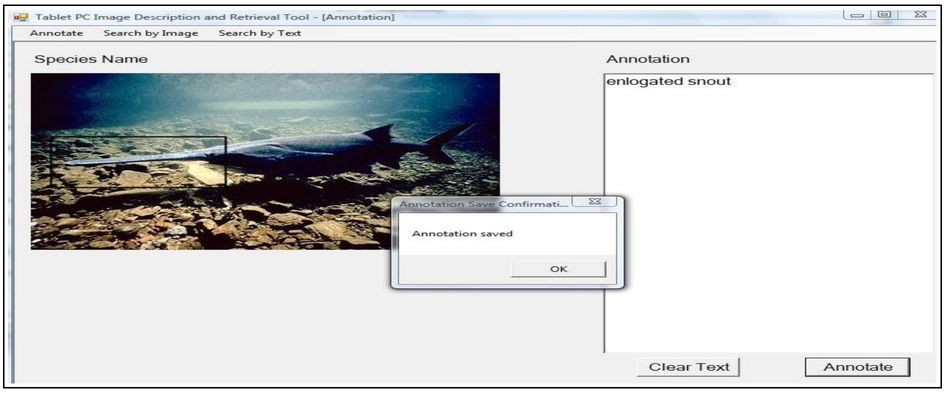

Figure 5. The context-based annotation with part of a image

5. Conclusions

We described a .Net implementation of a content-based image search component used for Superimposed-Information on annotation and retrieval of images. It has been developed to assist users managing, searching, annotating and combining features to enhance searching for contextualized information. This component is a DLL written in C#**,** using different types of descriptors, resulting in a flexible and portable application under the technology perspective.

Acknowledgments

Authors are grateful to FAPESP, CAPES, CNPq, and Microsoft Research for financial support. The authors also thank Eric Hallerman, Edward Fox, Uma Murthy and Yinlin for their support during the design and implementation of the proposed component.

References

Murthy, U., Fox, E. A. and Torres, R. da S. (2006) "Sierra - A Superimposed Application for Enhanced Image Description and Retrieval", In: European Conference on Digital Libraries, v. 4172/2006, p. 540-543, Alicante, Spain.

Murthy, U. (2007) "A Superimposed Information-Supported Digital Library", In: Doctoral Consortium at the 2007 Joint Conference on Digital Libraries. Vancouver, Canada, June.

Torres, R. da S., Gonçalves, M. A., Medeiros, C. B. and Fox, E. A. (2004) "An OAI Compliant Content-Based Image Search Component", In: ACM-IEEE Joint Conference on Digital Libraries, Tucson, AZ, 418, June.

Torres, R. da S., Medeiros, C. B., Gonçalves, M. A. and Fox, E. A. (2006a) "A Digital Library Framework for Biodiversity Information Systems", In: International Journal on Digital Libraries, v. 6, P. 3-17. Berlin.

Torres, R. da S. and Falcão, A. X. and (2006b) "Content-Based Image Retrieval: Theory and Applications", In: Revista de Informática Teórica e Aplicada, v. 13, p. 161-185.Porto Alegre, RS, Brasil

Vemuri, N. S., Torres, R. da S., Goncalves, M. A., Fan, W. and Fox., E. A. (2006) "A Content-Based Image Retrieval Service for Archaelogy Collections", In: European Conference on Digital Libraries, v. 4172/2006, p. 438-440, Alicante, Spain.