EFFICIENT USE OF CONTENT-BASED RETRIEVAL IN QUICK BROWSING: A REALIZATION

G. Votsis, Y. Xirouhakis, K. Karpouzis, and S. Kollias
Computer Science Div., Dept. of Electrical and Computer Eng.
National Technical University of Athens
Athens GR-15773, GREECE

Abstract - **Due to recent growth of interest in multimedia applications, an increasing demand has emerged for efficient storage, management and browsing in multimedia databases. This fact has made the content-based retrieval (CBR) concept very popular during the past decade. However, it has been practically shown that CBR tools respond successfully to the inexperienced user queries very rarely. It is for this reason that in commercial systems, queries are supported by the use of annotations or manual indexing. In this work, content-based query, retrieval and indexing capabilities have been combined with an intelligent agent framework over a simple database architecture. The proposed system is based on the ideas presented in [1] which are implemented and further extended in this work.**

INTRODUCTION

Multimedia applications involve storage and retrieval of voluminous data such as images and video sequences. From the inexperienced user's side, the large volume of data introduces important functional problems such as difficulty in finding desired information and time-costly browsing in databases. In view of these problems, the idea of browsing visual information in the same way as browsing text using search engines has become very popular. For this purpose, a number of prototype systems have already been developed, extracting visual features such as color, texture and shape from images or videos and using these features for visual information comparison. Some prototype systems providing such capabilities, include Virage, QBIC [2] and Photobook [3], among others, which are now in the stage of experimental evaluation.

Although the idea of automatically characterizing visual information in a unified manner and performing content-based queries has become very popular, it seems that the inexperienced user often finds it difficult to use such tools successfully. Quite often, users do not agree with the retrieved results or disagree with their ranking of relevance. This is due to the following fact;

0-7803-5610-1/99/$10.00 © 1999

the CBR concept, although powerful, has not been yet supported by tools extracting high-level information (semantics). In this sense, a system extracting low-level features such as color or texture cannot by itself facilitate browsing in a multimedia database. This problem can be tackled by extracting high-level features, such as objects, and dividing them into known object-classes [4]. Such an approach, though, can not be easily adopted, since it works under certain constraints and it is not as generic as for example the extraction of color histogram.

In this work, CBR framework is utilized 'as is' for quick browsing in the database contents. A simple intelligent agent architecture is employed in order to partition database contents into overlapping profiles, to one of which every user is appropriately assigned. For this purpose, the main intelligent agent framework [5, 6] is then combined with a domain discrimination technique. This architecture complements the CBR functionality, implemented on the basis of known techniques [2, 3, 7, 8] by limiting querying in subparts of the database. In this way, rarely will the system indicate as 'relevant' images belonging to completely different themes in the database. In the following, we consider the case of static images. In the implemented prototype, video objects have also been included in terms of characteristic frames [4]. At the same time, a number of additional features under development are to be incorporated including face detection techniques [9].

SYSTEM ARCHITECTURE

A modular representation of the system architecture is depicted in Figure 1. On the top of the block diagram, the 'Client' box denotes the web-based environment provided for user navigation into the database contents. The Web Server box establishes communication with the 'Client' and is responsible for the data exchange between the user and the application. When the user first enters the system, the 'Web Server' module collects a form of personal data, which are evaluated by the 'Agent' module in order to classify the user to one of the existing profiles. In the sequel, an initial query is automatically performed by the system retrieving the most popular images in the categories associated with the particular profile [1]. In general, all image queries are directed by the Web Server module to the Search Engine module, which in turn formulates the query and passes it through to the Database. At this point, the user can gradually browse all images in these categories, sorted in descending popularity. Alternatively, he/she may request for a content-based query using as an example any of the last retrieved images. In the 'Database' module, appropriate 'Feature Vectors' are stored for every image imported, so that feature vector comparison be performed in order to respond to user's query. The user may browse or even refresh the query in order to find the desired image. Finally, the user's preference is made clear when the particular thumbnail is downloaded or viewed in full size. This information is passed from the 'Web Server' to the 'Agent' module, which in turn informs the 'Database' for changes in popularity, updating the particular profile. Apart from the appropriate 'Feature Vectors', the Database module includes all stored visual material, the employed Feature Extraction Algorithms and classification information (existing profiles, look-up tables, indexes).

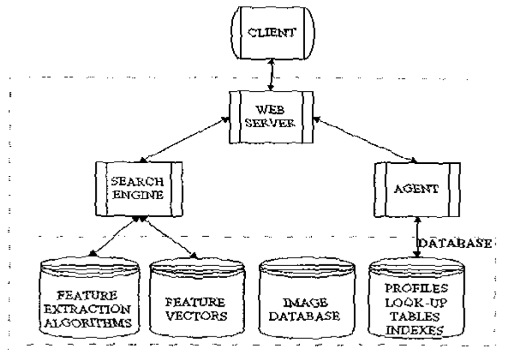

Figure 1: System schema

PROFILES ASSIGNMENT AND DATABASE ARCHITECTURE

The system consists of three abstract levels (sets): the profiles, the categories and the images set. The former consists of a number of distinct characteristic users (profiles) P_n, $n = 1 \cdots N$, where N denotes the number of profiles stored in the system. Each of these profiles corresponds to a certain subset of the categories set. In turn, the categories set consists of a number of distinct categories C_m, $m = 1 \cdots M$, where M is the total number of predetermined categories. The images set consists of all images I_k, $k = 1 \cdots K$, where K is the number of images stored in the database. It is presumed that every image belongs to all categories. In this sense, every image is saved along with a vector of weights $\mathbf{w}_k^I$ classifying it with higher or lower certainty to all categories. For example, if $\mathbf{w}_5^I(1) > \mathbf{w}_5^I(2)$, then image I_5 belongs to category C_1 with higher certainty than to C_2. This definition of weights is also a measure of image 'popularity' in a certain category. For example, if $\mathbf{w}_5^I(2) > \mathbf{w}_6^I(2)$, then image I_5 belongs to category C_2 with higher certainty than image I_6 does. Consequently, an $M \times K$ look-up table $\mathbf{I}$ with its columns corresponding to the vectors of weights, contains sufficient information for images classification to categories.

Similarly, every profile is saved along with a vector of weights $\mathbf{w}_n^P$, classifying it to all categories. For vectors $\mathbf{w}_n^P$ similar results to that of $\mathbf{w}_k^I$ can be obtained by comparing respective weights. Consequently, a sparse $M \times N$ look-up table $\mathbf{P}$ with its columns corresponding to the vectors of weights, is formed. At first, the particular user is assigned a certain profile P_a. At the same time, the user is assigned a certain number of categories C_i with respect to $\mathbf{w}_a^P$. An initial query is performed automatically retrieving the most popular images from categories C_i. A sequence of retrieved database entries (images) is available to the user, who is capable of selecting any image in the database starting with the most popular images in profile P_a.

Once the user has located an image that bears some resemblance to the one he/she has in mind, he/she is given the opportunity to perform a query-by-example using the particular image as in Figure 3. For every image imported in the database a feature vector is extracted, containing information concerning image attributes with respect to defined similarity metrics. After the user has decided on the relevant attribute similarity, by imposing certain weights for each attribute, a query is performed returning a new set of retrieved data. The user is able to refine the query until he comes up with the desired output.

Along the lines of the described procedure, the user finally comes up with the desired image. His/her choice becomes clear to the system by selecting to view the thumbnail in full size or selecting to download it. This is valuable information for the user and his/her profile that must not be wasted; in this sense, the look-up tables are updated. Supposing that the k-th image is selected, the respective weights in $\mathbf{w}_k^I$ are updated regarding vector $\mathbf{w}_a^P$, or even equally increased.

In an extended scenario, the particular user does not only select a profile, but submits a form containing personal evidence. In a field named Interests or *Profession* the user selects the appropriate profile, while at the same time he is able to fill in a number of other attributes such as *Age, Sex, Educational Level* (Figure 2). These attributes are again quantized. Then, again, a profile is assigned to the user and weights are updated as in the previous scenario. In addition, classification is performed for the existing images concerning the remaining user attributes.

CONTENT-BASED RETRIEVAL SCHEME

In visual information retrieval (VIR) applications, visual information is defined as the outcome of image processing transformations applied on visual objects [7]. For fundamental properties of an image, a variety of visual features, which are considered to effectively portray content are supported by our system [2, 3, 7, 8] (Figure 3). The mechanism applied in our system exploits the use of a vector, composed by the following features:
(a) Color: Image's color histogram is employed.
(b) Color composition: Involves topological color distribution. It is achieved through the combination of quad-trees and histogram computation.
(c) Shape: Image segmentation along with edge detection techniques enable calculation of quantitative amounts, such as area, centrality and elongation.
(d) Form: Faces the problem of object class discrimination using the Karhunen-Loeve transform.
(e) Texture: Supposing that a texture image is a homogeneous discrete 2D

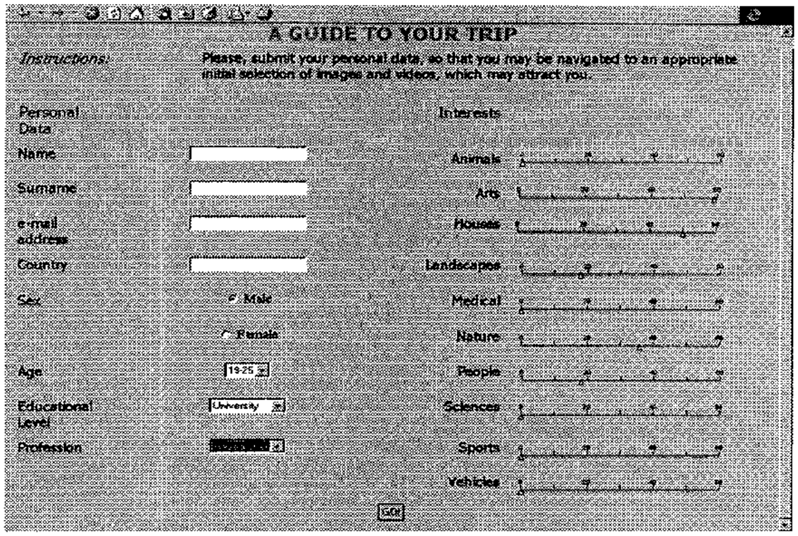

Figure 2: Determining the desired profile

random field, a decomposition model in three mutually orthogonal components is utilized [2].

(f) Texture composition: As in the case of color, a quadtree structure provides topological knowledge of texture synthesis of the image.

By using the above features, the user has the opportunity to drive his query in the feature space and not directly in the image space. Questions of the form: 'if each image is viewed as a point in the n-dimensional space, find the m closest images within distance d from the characteristic vector $\mathbf{v}$' can be answered [7].

CONCLUSIONS

In this work, the ideas first presented in [1] are implemented and further extended in order to achieve efficient image browsing and retrieval in multimedia databases. The proposed system is based on the concepts of intelligent agent management and CBR tools. A simple architecture and a voting system training strategy are implemented, along with CBR techniques, minimizing search time in voluminous databases.

References

[1] Y. Xirouhakis, G. Votsis, K. Karpouzis, and S. Kollias, "Efficient Browsing in Multimedia Databases using Intelligent Agents and Content-Based

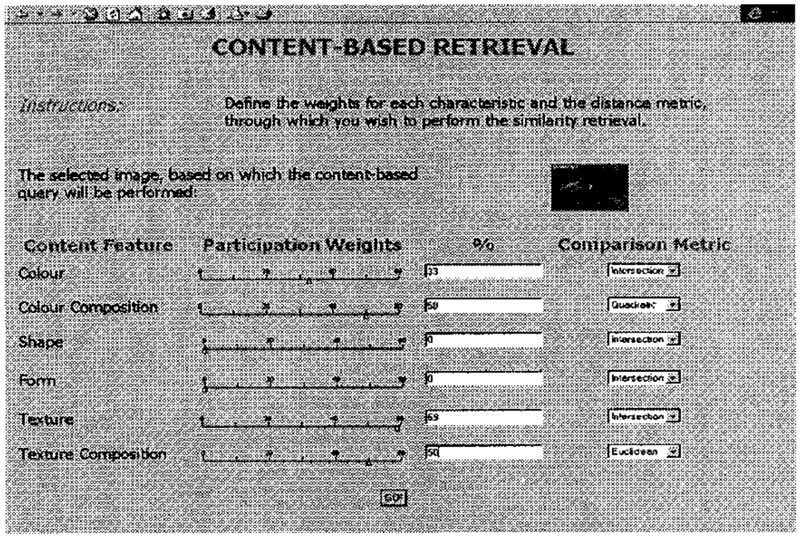

Figure 3: The content-based retrieval tool

Retrieval Schemes," in *Proc. of MMSP98 (IEEE)*, Los Angeles, California, USA, Dec. 1998.

[2] M. Flickner *et al*, "Query by Image and Video Content: The QBIC System," *Computer*, vol. 28, pp. 23–32, Sept. 1995.

[3] A. Pentland, R. W. Picard and S. Sclaroff, "Photobook: Content-Based Manipulation of Image Databases," *SPIE Storage and Retrieval Image and Video Databases II*, no. 2185, Feb. 1994.

[4] Y. Xirouhakis, Y. Avrithis, and S. Kollias, "Image Retrieval and Classification using Affine Invariant B-spline Representation and Neural Networks," in *IEE Colloq. on NN in Multimedia Interactive Systems*, London, UK, Oct. 1998.

[5] C. Guilfoyle and E. Warner, *Intelligent Agents: The New Revolution in Software*, pp. 214, 1994.

[6] C. T. Lin and C. S. G.Lee, *Neural fuzzy systems: a neurofuzzy synergism to Intelligent Systems*, Prentice Hall, 1995.

[7] A. Gupta and R. Jain, "Visual Information Retrieval," *Communications of the ACM*, vol. 40, no. 5, May 1997.

[8] C. E. Jacobs, A. Finkelstein and D. H. Salesin, "Fast Multiresolution Image Querying," in *Proc. SIGGRAPH 95*, Los Angeles, CA, Aug. 1995.

[9] K. Karpouzis, G. Votsis, N. Tsapatsoulis and S. Kollias, "Compact 3D Model Generation based on 2D Views of Human Faces: Application to Face Recognition," *Machine Graphics and Vision*, vol. 7, no. 1/2, pp. 75-85, 1998.