Content based image and video retrieval using the QuickLook search engine

G. Ciocca\textsuperscript{a}, I. Gagliardi\textsuperscript{b}, R. Schettini\textsuperscript{a}

\textsuperscript{a}Dipartimento di Informatica Sistemistica e Comunicazione (DISCo)
Università degli studi di Milano-Bicocca
Via Bicocca degli Arcimboldi 8, 20126 Milano, Italy
ciocca@disco.unimib.it, schettini@disco.unimib.it

\textsuperscript{b}I.T.C.
Consiglio Nazionale delle Ricerche (C.N.R.)
Via Bassini 15, 20133 Milano, Italy
gagliardi@itc.mi.cnr.it

ABSTRACT

This paper describes the main features of QuickLook, a system that combines within a single framework the capabilities of alphanumeric relational query, the content-based image and video query exploiting automatically computed image features, and the textual similarity query using any textual annotations attached to database items (such as figure and video captions). The system offers the user several retrieval strategies for querying the database. He can then progressively refine the system’s response by indicating the relevance, or non-relevance of the items retrieved. Once a subset of images meeting the user’s information needs have been identified, these images can be displayed in a pseudo-3D virtual exhibition which the user can visit interactively by applying VRML technology.

Keywords: image and video indexing, image and video retrieval, multimedia system, content-based image and video retrieval, relevance feedback, textual annotations, virtual exhibition.

1. INTRODUCTION

The need to retrieve visual information from large image and video collections is shared by many application domains, and a wide variety of content-based retrieval methods and systems can be found in the literature [1][2][3][4]. Their capability as general purpose systems is, however, in large part limited by the a-priori definition and setting of:

1. the user’s aims, which may be broadly classified as:
   a) target search: the user knows exactly what image he is looking for,
   b) similarity search: the user wants to retrieve all the images resembling an example,
   c) and category search: the user aims to retrieve an image representative of a specific class;
2. the set of features (visual or textual) used for image indexing;
3. the similarity metric adopted;
4. the way in which the user may interact with the system in order to express his/her information needs.

In this paper we describe the main features of QuickLook, our system for the interactive search of items in electronic catalogues. The system is now being extended to deal with video sequences by integrating a video indexing module that summarizes the video visual contents as stills. QuickLook allows the user to query image, video and multimedia databases with the aid of sample images, or a user-made sketch, and/or textual descriptions, and the system’s response can then be progressively refined by indicating the relevance, or non-relevance of the multimedia items retrieved. In particular, we have designed a mechanism, which exploits the statistical analysis of the image feature distributions and of the textual descriptions of the retrieved items the user has judged relevant, or not relevant, to identify what features the user has taken into account (and to what extent) in formulating his judgement. It then modifies accordingly the impact of the different visual and textual features in the overall evaluation of image similarity, as well as in the formulation of a new single query representing the user's information needs.

The design of a content-based image retrieval system must address issues of efficiency in addition to those of effectiveness. Consequently another contribution has been the design and integration with the relevance feedback mechanism of an indexing scheme based on triangle inequality. QuickLook, in addition to the usual 2D representation of the retrieval results, allows users to display the results in a virtual 3D environment.

2. SISTEM OVERVIEW

The system is composed of five units [Fig. 1]: the Visual Interface Module, the entry point for user interaction; the Retrieval Module, which processes queries and retrieves the items that satisfy them; the Multimedia Database; the Indexing Module, which extracts all the information needed to perform queries on multimedia data; and the Visualization Module, which displays the images retrieved by a query in a virtual 3D exhibition.
The Visual Interface Module: The Visual Interface Module allows the user to view the catalogue’s content as a series of thumbnail images of the available items. The user can also access any textual information associated with a selected item, such as the name, location and general description (for example, “A white satin chasuble embroidered in gold and colored silks…”). He can also view the corresponding quality image to check the details. When he finds an item that resemble what he is looking for, and wants to see if there are other similar items, he can start a search session by running visual and/or textual queries, using the selected items as examples.

The Retrieval Module: The Retrieval Module processes the queries, and retrieves the items satisfying them. A generic query may be composed of visual and/or textual parts (sub-queries). Each sub-query is processed separately, the results are then combined with a similarity function to obtain a final score, and the images ranked accordingly. If the user is not satisfied by the system’s response, he can refine the search by adding textual constraints, and/or providing more examples of items relevant, or non-relevant to what he is looking for. The similarity function is dynamically adapted to the query by a relevance feedback mechanism which updates the weights (and thus the importance) assigned to each textual or pictorial feature.

The Multimedia Database: The Database contains three different types of data [Fig. 2]:
- Binary data: thumbnail images or stills extracted from video sequences (key-frames also presented for display in thumbnail form), and video streams;
- Numerical data: numerical information about the color, texture, shape, and color distribution of image contents; these features are automatically extracted without user supervision;
- Alphanumeric data: alphanumerical information given in the textual description associated with the image, which can be divided into: i) content-independent data i.e. data not directly concerned with image content, but are in some way related to it (the author, date, location, ownership, …); these data are also known as keywords; ii) content-descriptive data i.e. data concerning image semantics, in the form of text of a certain length describing the images in natural language; iii) text automatically extracted from video sources, such as captions or subtitles.

The Indexing Module: the different data sources (text, images and video) are processed separately and the information are extracted and stored in the database. A detailed description of the information extraction process is given in the next section.

Visualization Module: most CBIR systems to date present a 2D representation of the results of a retrieval session. We have extended the basic capabilities of our system to include a more natural presentation. While maintaining the 2D view for the browsing / querying of database data and as the
standard visualization method, we have also provided a 3D environment in which to display the images retrieved by the system can be more realistically displayed. In this virtual exhibition the images are “hung” on gallery walls in the same order in which they have been retrieved. The user can navigate freely through the “rooms” and if desired read the information associated with each picture. To ensure compatibility with currently employed web browsers and limit costs in terms of data transmission and processing time, we have employed the standard VRML 2.0 data specification language.

3. MULTIMEDIA OBJECT INDEXING

3.1. Content-independent data: Textual Keywords
Textual keywords do not deals directly with image content, but are in some way related to it. Examples are: format, author, date, location, ownership. This type of data is handled with traditional DBMS techniques. The system retrieves only the subset of objects that satisfies all the constraints defined by the keywords chosen by the user.

3.2 Content-descriptive data: Textual Descriptions
Images and video frames are sometimes accompanied by textual annotations describing their semantic contents. These annotations are used in indexing and retrieving images automatically: significant terms are extracted to create a dictionary, and a suitable similarity function among sets of significant terms is defined [5].

In QuickLook, designed as a general purpose system, the dictionaries are created automatically, and are composed of all the terms present in the textual annotations (excepting those on a standard Italian stop list). No stemming procedure is applied, as no satisfactory algorithm is available for the Italian language. Most morphological variations (singular/plural, feminine/masculine, ...) are automatically eliminated. Each index term of a document is automatically assigned a weight TW reflecting its importance, based on the number of times the term itself occurs in the single document, and in the entire archive. The textual annotation associated with the generic image i is indexed therefore by a set of its relevant terms with which TW weights have been associated. We call such a set T_i. Text similarity, TS, between the textual annotations T_i and T_j, is defined as follows:

$$TS(T_i, T_j) = \frac{\sum_{k \in T_i \cap T_j} TW_k}{\sqrt{\sum_{k \in T_i} (TW_k)^2 \sum_{k \in T_j} (TW_k)^2}}$$

TS can assume any value in the range of [0, 1]. The greater the value of TS, the greater the similarity between the two textual annotations.

3.3. Content-dependent data: Pictorial Features
Content-dependent data refer to the visual content of images. Examples are color, texture, shape, spatial relationship, and combinations of these. This type of information is very difficult to define
and index using natural language: it requires the design of algorithms that extract suitable surrogates of the visual features from the images; these surrogates can then be processed automatically. Since we did not design QuickLook for any specific application, we have constituted a general purpose library of low-level features to use in image indexing. This library is continuously extended and updated with new features. The default visual index contains: the ratio between the dimensions of the image, the Color Coherence Vector (CCV) [6], the Spatial Chromatic Histogram (SCH) [7], the moments of inertia (mean, variance, skewness and kurtosis) [8], the histogram of contour directions [9], the mean and variance of the absolute values of the coefficients of the sub-images at the first three levels of the multi-resolution Daubechies wavelet transform of the luminance image [9], and the spatial composition of the color regions identified by a process of quantization in eleven colors [10]. These features are all calculated on the global image and on sub-images obtained by dividing the original image in different ways.

Two pictorial indices are compared by computing a similarity distance for each pair of corresponding features (each feature has its own evaluation function), and then combining the results in a final score. The score is obtained by weighting each distance with the relevance weight calculated by the relevance feedback algorithm. To cope with the problem of distances defined on different intervals of values, we apply a normalization derived from Gaussian normalization [11][12].

3.4. Video Analysis / Video Indexing

We summarize or abstract the visual content of the video sequences with still images and apply the same retrieval strategies developed for images to retrieve video sequences. The still images chosen to represent the video content are called key-frames. Automatic video analysis and indexing is a complex process that involves different tasks [13]:

- **Definition of the representation of visual contents.** Video indexing must capture the spatio-temporal contents of the video in a compact way. In order to do so, the first step in video indexing is the definition of a suitable representation of the visual content (i.e. a suitable representation for the video frames). We have decided to work on uncompressed data instead of compressed data in order to have access to the algorithms developed for image indexing, such as histograms, texture representation or content layouts, and to be able to choose from among them those most suitable to our purposes. To compensate for the time-consuming step of video decoding, we apply the indexing algorithm on spatial sampled frames.

- **Segmentation of the video in elementary units.** In the current implementation we segment the video into shots (a continuous sequence of frames taken over a short period of time) by detecting abrupt changes and fades between them, since these are more common than other editing effects. For abrupt changes we have implemented a threshold-based algorithm coupled with a frame difference measure computed from histograms and textures descriptors. To detect fades we have implemented a modified version of the algorithm proposed by Fernando et al. [14]. The results obtained by these algorithms are submitted for evaluation to a decision module which gives the final response. This allows us to cope with conflicting results, or with groups of frames that are not meaningful, such as whose between the end of a fade out and the start of a fade-in, increasing the robustness of the detection phase. A gradual transition detection algorithm is currently being developed; it will be integrated in a similar manner.
- **Summarization of the video.** Our system dynamically selects representative frames from the shots by analyzing the complexity of the events presented and discarding redundant information. We use a measure of the cumulative visual content change between two consecutive frames within the shot to measure the complexity. The values obtained describe how the frames change over the entire shot, high slopes indicate significant changes in the visual content, which can for the most part be attributed to moving objects, the movement of the camera itself, or highly dynamic events. These cases are considered “interesting event points” which must be included in the final shot summary [Fig. 3].

- **Identification of the overall video structure.** The set of frames obtained by summarization of the entire video can be used to represent the overall contents without having to watch the whole video. By changing the algorithm’s tuning parameters, we can change the granularity of the summary obtained and, consequently, the number of frames retrieved.

- **Definition of indexing.** The indexing task must take into account the information extracted from the previous steps in order to build a hierarchical structure and allow access to the video information at different levels of abstraction. At each step in the video analysis process, we can store different gathered information and index the video with the shots and scenes positions, their duration, the set of key frames and so on. Within QuickLook key frames are stored and treated like standard images, linked with the corresponding video segments.

An example of the summarization of a news video about three and half minutes long including the commercials is presented in [Fig. 4].

![Figure 3](image)

**Figure 3.** Shot summarization example: a) example shot; b) graph of the cumulative visual content of the sequence (the “interesting points” are identified by selecting the points of the graph that present the highest curvature); c) the representative frame is the middle frame between each pair of consecutive high curvature points; note that, if the shot does not display a dynamic behavior, i.e. the frames of the shot are highly correlated, the graph does not present evident curvature points and the shot can be summarized by a single representative frame; d) the key-frames extracted from the shot.
Figure 4. Key-frames extracted from a 3 minute and 39 second long news video with commercials (6,561 frames at 30 fps). The video has been compacted into 47 frames, and 41 shots have been detected. The system extracted more key-frames from the commercials than from the news segments, due to the highly dynamic contents of the former.

4. MULTIMEDIA OBJECT RETRIEVAL

The pictorial content and textual content of the indices allow the user of our system to submit a number of different kind of queries:

Textual Search by Keywords: the user searches for a set of images belonging to the same category, specifying a precise condition, such as, all the textiles that are 100% wool, or the set of artifacts produced by a specific manufacturer.

Query by Sample:

a. Pictorial Query by Sketch: the user makes an approximate sketch of the desired image; the sketch continues to represent his basic query while he interacts with the system.

b. Pictorial Target Search: the user imports the image depicting the visual object he wants to retrieve from the database; the imported image continues to represent his basic query while he interacts with the system.
c. **Query by Textual Sample**: the user supplies an approximate description of the desired object, i.e. “plate with flowers in the center and a blue border”; the text continues to represent his basic query while he interacts with the system while.

**Query by Examples:**

a. **Query by Pictorial Example**: the user selects one or more relevant (or non-relevant) multimedia objects from the system’s display focusing on image/frame attributes. In subsequent iterations both the image similarity measure adopted to compare image descriptions and the query vector representing the user’s information needs are updated on the basis of user feedback.

b. **Query by Textual Example**: the user selects one or more relevant (or not relevant) multimedia objects from the system’s display, focusing on textual description attributes. The textual descriptions associated with the images are used in the retrieval process. In subsequent iterations both the text similarity measure applied to compare image annotations and the textual query vector representing the user’s information needs are updated on the basis of user feedback.

c. **Query by Multimedia Object Example**: the user selects one or more relevant (or non-relevant) multimedia objects as a composite entity. All available information, visual and textual, is exploited in the retrieval. The query representing the user’s information needs as well as both the visual and textual similarity measures are updated and combined on the basis of user feedback.

4.1. **Relevance Feedback**

A relevance feedback mechanism is used to update the weights of the similarity function. The key concept of the relevance feedback mechanism, described in detail in [11] and [12], is that the statistical analysis of the image feature distributions and textual descriptions of the images the user has judged relevant, or not relevant, can be used to determine what features the user has taken into account (and to what extent) in formulating this judgement, and then accentuate the influence of these features in the overall evaluation of image similarity, as well as in the formulation of a new query. The structure of the relevance feedback mechanism is entirely description-independent, that is, the index can be modified, or extended to include other features without requiring any change in the algorithm.

Sub-vectors of visual features are indicated by $X^i_h$, where $i$ is the vector index, and $h$ the index of the feature; $T^i$ is the corresponding textual annotation, if available; $d_h$ is the normalized distance associated with the feature $h$-th (a normalization derived from the Gaussian normalization is applied); and $TS$, the similarity function associated to the textual annotations. The global metric used to evaluate the dissimilarity between two database items is defined as a linear combination of the distances between the individual features:

$$
\text{Dist}(X^i, X^j) = \frac{1}{p} \sum_{h=1}^{n} w_h d_h(X^i_h, X^j_h) + w_{\tau} (1 - TS(T_i, T_j)) = \frac{1}{p} \sum_{h=1}^{n} w_h d_h(X^i_h, X^j_h) + w_{\tau} d_{\tau}
$$

(2)
Textual similarity does not have, by default, a greater weight than that of pictorial similarity to reduce the risk of over-learning the user’s notion of similarity, and jamming the system. Moreover, text descriptions reflect the point of view of the annotator, which may not be that of the final user querying the system. However, we have seen in our investigation of several application domains that the similarity measure used by the system can be successfully tuned to the user’s notion of image similarity with no significant effort on the part of the user. In particular, we have observed that the algorithm makes it possible to cope with two very common problems:
- The heterogeneity of the image database queried, which may render the retrieval task particularly complex. Consequently, the user may not find enough examples in the first screens that are truly similar to the query, and, to avoid the time-consuming visual browsing of the database, may mark as relevant images that are only partially similar.
- The user's information needs may also be overly vague, such as: "find all the images containing animals".
In both cases the images judged relevant may differ widely.

4.2. Query refinement
In our experience in content-based retrieval images are sometimes considered relevant because they resemble the query image in just some limited pictorial features. Consequently, after an initial query, one retrieved image may be considered relevant because it is the same color as the query, and another be selected for its similarity in shape, although the two are actually quite different from each other. To cope with this problem, we have developed a new method for computing the query vector called query refinement [12]. On the basis of the initial results, the query processing formulates a new query that better represents the images of interest to the user, taking into account the features of the relevant images, without allowing any one particular feature value to bias query computation [Fig. 5]. A similar process is used for text: words found in relevant texts are added together to increase the weights associated with each word according to their relative frequency in the texts; instead, words present in both relevant and non-relevant texts are discarded.

![Query refinement diagram](image)

Figure 5. Query refinement: a) the scheme of the query refinement process; b) an example of how the new query representation is built by the query processing algorithm.

4.3. Image Filtering
Since comparing a query Q with every image I in the database is a time-consuming task, we have implemented a method for filtering the database before the pictorial distances are actually computed. This method is based on a variant of triangle inequality as proposed by Berman and
Shapiro [12][15], and has the advantage of being applicable to any distance measure that satisfies triangle inequality.

5. VISUALIZATION

The visualization module provides a 3D environment in which the user can automatically create a virtual exhibition in which to display the results of his query. With the help of a visual designer we have created a set of exhibition environments devised to be as neutral as possible they resemble real museums and expositions, and allow the automatic collocation of the retrieved images, without any apparent geometric distortion [16].

The panoramic images of the virtual environments have been acquired with a virtual QTVR camera and then stored in the system archive. Once the user selects a given environment, the images to be shown are pasted “on the fly” on the panoramic image. This image is then simply mapped on a cylindrical surface for display and virtual navigation. Should the number of images to be shown exceed the maximum that can be exhibited in a single environment, “rooms” are automatically created and linked by “virtual doors”. The user can customize the display of the images by adding a passe-partout and a frame for a more realistic display; moreover the display respects the proportions between the true size (if available) of the originals (paintings, for example) and the dimensions of the architectural elements present in the room [Fig. 6]. Obviously the user may not only visit the exhibition, but also retrieve all the related information available by simply clicking on the image.

Figure 6. a) A room of the three dimensional virtual museum; a) the circular perimeter chosen to keep distortions due to the perspective at a minimum; b) floor plan, the museum has been planned to allow the viewer to roam freely through the different virtual rooms; c) a panoramic view of a room of the virtual Museum “Sky”; the walls are positioned to present no or minimal deformation; near the left edge of the display, and at the extreme right, the passages leading to the other rooms;
d) a portion of the same view with pictures hung on the walls; the user can zoom and scroll the view in all directions.

6. CONCLUSIONS

We have described the main features of the multimedia information retrieval engine of QuickLook. The system is equipped with a visualization tools that make it possible to display the results of a query session in different virtual environments. These environments can then be interactively visited by the user, exploiting VRML technology. At the moment six neutral environments are available; we plan enlarge this library in the early future.

The retrieval performance of the system has been tested on different databases ranging from photos to trademarks and objects. QuickLook performs very well with all kind of image databases, and the relevance feedback mechanism allows the user to retrieve satisfying results in a few iterations. Retrieval performance results and a more in-depth description of the system’s architecture and algorithms can be found in our previous papers [9-12].

The video processing module will be integrated in an advanced version of QuickLook and will include new effects detector algorithms. We are also studying a friendly interface that will allow the user to inform the system of the type of image database to be indexed; the system will then automatically ignore (i.e. not compute) certain pictorial features if the images belong to a predefined class. We plan to create a database preview as well by clustering both visual and textual information. An on-line demo of the QuickLook system for image databases with the 3D visualization module can be found at the address http://quicklook.itc.cnr.it.

ACKNOWLEDGEMENTS

The video indexing and analysis presented here has been developed within the Italian MURST FIRB Project MAIS (Multi-channel Adaptive Information Systems) [17].

REFERENCES


BIOGRAPHIES

**Gianluigi Ciocca** took his degree (Laurea) in Computer Science at the University of Milan in 1998. He has been since then a fellow at the Institute of Multimedia Information Technologies of the Italian National Research Council (C.N.R.), where his research has focused on the development of systems for the management of image and video databases, and the designing of new methodologies and algorithms for automatic indexing. He is currently a PhD student in computer science at DISCo (Dipartimento di Informatica, Sistemistica e Comunicazione) of the University of Milano-Bicocca, where he works on video analysis and abstraction.

**Isabella Gagliardi** took her degree in Physics at the University of Milan in 1985. She has been working at the National Research Council (C.N.R.) since 1986 where she is now in the Multimedia Information Technologies department of the Institute of Construction Technologies (I.T.C.). Dr. Gagliardi’s major areas of research include hypermedia information retrieval models and methodologies, the automatic generation of hypertextual links text to text, and between text and image, and the evaluation of algorithms of content-based image retrieval. On these topics she has produced several national and international publications, and participated in national and international research projects.

**Raimondo Schettini** is Associate Professor in the Department of Information Science, Systems Theory, and Communication (DISCo), at the University of Milano Bicocca, where he is in charge of the Imaging and Vision Lab. He has worked with Italian National Research Council (C.N.R.) since 1987. He has been team leader in several national and international research projects and published over 140 refereed papers on image processing, analysis and reproduction, and image content-based indexing and retrieval. He was General Co-Chairman of the First Workshop on Image and Video Content-based Retrieval (1998) of the First European Conference on Color in Graphics, Imaging and Vision (CGIV’2002), and of the EI Internet Imaging Conferences (2000-2004). He has been guest editor of a special issue about Internet Imaging of the Journal of Electronic Imaging, 2002, of another one on Color Image Processing and Analysis in Pattern Recognition Letters, 2003, and is currently associate editor of the Pattern Recognition Journal.