

Chapter III

Content-Based Visual Information Retrieval

Oge Marques and Borko Furht
Florida Atlantic University, USA

This chapter provides a survey of the state-of-the-art in the field of Visual Information Retrieval (VIR) systems, particularly Content-Based Visual Information Retrieval (CBVIR) systems. It presents the main concepts and system design issues, reviews many research prototypes and commercial solutions currently available and points out promising research directions in this area.

INTRODUCTION

The amount of audiovisual information available in digital format has grown exponentially in recent years. Gigabytes of new images, audio and video clips are generated and stored everyday, building up a huge, distributed, mostly unstructured repository of multimedia information, much of which can be accessed through the Internet.

Digitization, compression, and archival of multimedia information has become popular, inexpensive and straightforward, and there is a broad range of available hardware and software to support these tasks. Subsequent retrieval of the stored information, however, might require considerable additional work in order to be effective and efficient.

There are basically three ways of retrieving previously stored multimedia data:

- 1. Free browsing:** users browse through a collection of images, audio, and video files, and stop when they find the desired information.
- 2. Text-based retrieval:** textual information (metadata) is added to the audiovisual files during the cataloguing stage. In the retrieval phase, this additional information is used to guide conventional, text-based, query and search engines to find the desired data.
- 3. Content-based retrieval:** users search the multimedia repository providing information about the actual contents of the image, audio, or video clip. A content-based search engine translates this information in some way as to query the database and retrieve the candidates that are more likely to satisfy the users' requests.

The first two methods have serious limitations and scalability problems. Free browsing is only acceptable for the occasional user and cannot be extended to users who frequently need to retrieve specific multimedia information for professional applications. It is a tedious, inefficient, and time-consuming process and it becomes completely impractical for large databases.

Text-based retrieval has two big problems associated with the cataloguing phase:

- a) the considerable amount of time and effort needed to manually annotate each individual image or clip; and
- b) the imprecision associated with the subjective human perception of the contents being annotated.

These two problems are aggravated when the multimedia collection gets bigger and may be the cause of unrecoverable errors in later retrieval.

In order to overcome the inefficiencies and limitations of text-based retrieval of previously annotated multimedia data, many researchers, mostly from the Image Processing and Computer Vision community, started to investigate possible ways of retrieving multimedia information – particularly images and video clips – based solely on its contents. In other words, instead of being manually annotated using keywords, images and video clips would be indexed by their own visual content, such as color, texture, objects' shape and movement, among others.

Research in the field of Content-Based Visual Information Retrieval (CBVIR) started in the early 1990's and is likely to continue during the first decade of the 21st century. Many research groups in leading universities and companies are actively working in the area and a fairly large number of prototypes and commercial products are already available. Current solutions are still far from reaching the ultimate goal, namely to enable users to retrieve the desired image or video clip among massive amounts of visual data in a fast, efficient, semantically meaningful, friendly, and location-independent manner.

The remainder of this chapter is organized as follows. In Section 2, we review the fundamentals of CBVIR systems. The main issues behind the design of a CBVIR system are discussed in Section 3. Section 4 surveys several existing CBVIR systems, both commercial and research. Some of the open research problems in this field are presented in Section 5. Section 6 describes the main aspects of MUSE, a CBVIR system developed by the authors. Finally, Section 7 contains some concluding remarks.

FUNDAMENTALS OF CBVIR SYSTEMS

Preliminaries

Visual Information Retrieval (VIR) is a relatively new field of research in Computer Science and Engineering. As in conventional information retrieval, the purpose of a VIR system is to retrieve all the images (or image sequences) that are relevant to a user query while retrieving as few non-relevant images as possible. The emphasis is on the retrieval of *information* as opposed to the retrieval of *data*. Similarly to its text-based counterpart a visual information retrieval system must be able to interpret the contents of the documents (images) in a collection and rank them according to a degree of relevance to the user query. The interpretation process involves extracting (semantic) information from the documents (images) and using this information to match the user needs (Baeza-Yates, and Ribeiro-Neto, 1999)

Progress in visual information retrieval has been fostered by many research fields (Figure 1), particularly: (text-based) information retrieval, image processing and computer vision, pattern recognition, multimedia database organization, multidimensional indexing, psychological modeling of user behavior, man-machine interaction, among many others.

VIR systems can be classified in two main generations, according to the attributes used to search and retrieve a desired image or video file (Del Bimbo, 1999):

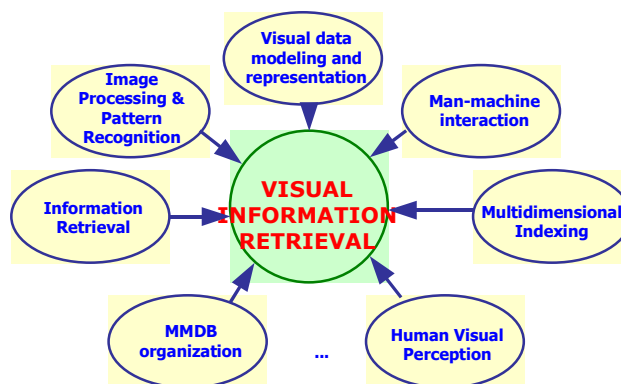
- **First-generation VIR systems:** use query by text, allowing queries such as “all pictures of red Ferraris” or “all images of Van Gogh’s paintings”. They rely strongly on metadata, which can be represented either by alphanumeric strings, keywords, or full scripts.
- **Second-generation (CB)VIR systems:** support query by content, where the notion of content, for still images, includes, in increasing level of complexity: perceptual properties (e.g., color, shape, texture), semantic primitives (abstractions such as objects, roles, and scenes), and subjective attributes such as impressions, emotions and meaning associated to the perceptual properties. Many second-generation systems use content-based techniques as a complementary component, rather than a replacement, of text-based tools.

A TYPICAL CBVIR SYSTEM ARCHITECTURE

Figure 2 shows a block diagram of a generic CBVIR system, whose main blocks are:

- **User interface:** friendly GUI that allows the user to interactively query the database, browse the results, and view the selected images / video clips.
- **Query / search engine:** responsible for searching the database according to the parameters provided by the user.
- **Digital image and video archive:** repository of digitized, compressed images and video clips.
- **Visual summaries:** representation of image and video contents in a concise way, such as thumbnails for images or keyframes for video sequences.
- **Indexes:** pointers to images or video segments.
- **Digitization and compression:** hardware and software necessary to convert images and videos into digital compressed format.

Figure 1. Visual Information Retrieval blends together many research disciplines.



- **Cataloguing:** process of extracting features from the raw images and videos and building the corresponding indexes.

Digitization and compression have become fairly simple tasks thanks to the wide range of hardware and software available. In many cases, images and videos are generated and stored directly in digital compressed format. The cataloguing stage is responsible for extracting features from the visual contents of the images and video clips. In the particular case of video, the original video segment is broken down into smaller pieces, called *scenes*, which are further subdivided into *shots*. Each meaningful video unit is indexed and a corresponding visual summary, typically a *keyframe*, is stored. In the case of images the equivalent process could be object segmentation, which just a few systems implement. In either case, the cataloguing stage is also where metadata gets added to the visual contents. Manually adding metadata to image and video files is mandatory for text-based visual information retrieval systems. CBVIR systems, however, typically rely on minimum amount of metadata or none at all.

Digitization, compression, and cataloguing typically happen off-line. Once these three steps have been performed, the database will contain the image and video files themselves, possible simplified representations of each file or segment, and a collection of indexes that act as pointers to the corresponding images or video segments.

The online interaction between a user and a CBVIR system is represented on the upper half of the diagram in Figure 2. Users express their query using a GUI. That query is translated and a search engine looks for the index that corresponds to the desired image or video. The results are sent back to the user in a way that allows easy browsing, viewing, and possible refinement of the query based on the partial results.

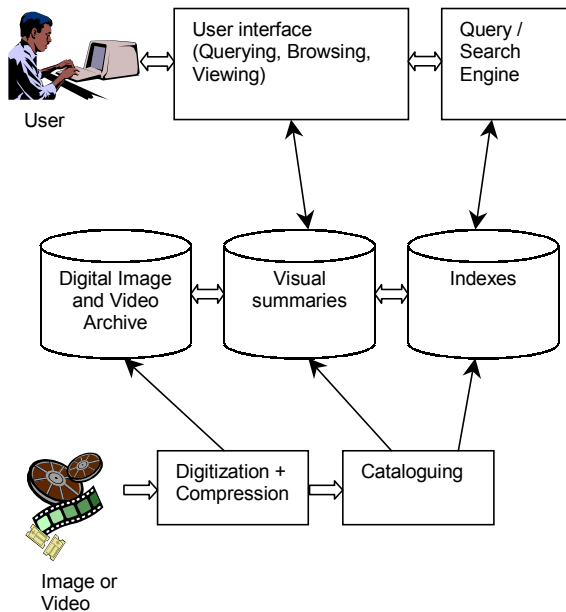
The User's Perspective

The user interface is a crucial component of a CBVIR system. Ideally such interface should be simple, easy, friendly, functional, and customizable. It should provide integrated browsing, viewing, searching, and querying capabilities in a clear and intuitive way. This integration is extremely important, since it is very likely that the user will not always stick to the best match found by the query/search engine. More often than not users will want to check the first few best matches, browse through them, preview their contents, refine their query, and eventually retrieve the desired image or video segment.

Most VIR systems allow searching the visual database contents in several different ways, described below, either alone or combined:

- **Interactive browsing:** convenient to leisure users who may not have specific ideas about the images or video clips they are searching for. Clustering techniques can be used to organize visually similar images into groups and minimize the number of undesired images shown to the user.
- **Navigation with customized categories:** leisure users often find it very convenient to navigate through a subject hierarchy to get to the target subject and then browse or search that limited subset of images.
- **Query by X** (Chang, Smith, Beigi and Benitez, 1997), where 'X' can be:
- **an image example:** several systems allow the user to specify an image (virtually anywhere in the Internet) as an example and search for the images that are most similar to it, presented in decreasing order of similarity score;
- **a visual sketch:** some systems provide users with tools that allow drawing visual sketches of the image or video clip they have in mind;

Figure 2. Block diagram of a CBVIR system.



- **specification of visual features:** direct specification of visual features (e.g., color, texture, shape, and motion properties) is possible in some systems and might appeal to more technical users;
- **a keyword or complete text:** first-generation VIR systems rely on keywords entered by the user and search for visual information that has been previously annotated using that (set of) keyword(s).

We advocate that query options should be made as simple, intuitive and close to human perception of similarity as possible. Users are more likely to prefer a system that offers the “Show me more images that look similar to this” option, rather than a sophisticated interactive tool to edit that image’s color histogram and perform a new search. While the latter approach might be useful for experienced technical users with image processing knowledge, it does not apply to the average user and therefore has limited usefulness. An ideal CBVIR system query stage should, in our opinion, hide the technical complexity of the query process from the end user. A search through visual media should be as imprecise as “I know it when I see it.” (Gupta, Santini, and Jain ,1997)

The Designer’s Perspective

Some of the main aspects in designing a CBVIR system are: feature extraction and representation, dimension reduction and multidimensional indexing, extraction of image semantics, and design of user relevance feedback mechanisms. These issues are explored in more detail in the following subsections.

Feature Extraction And Representation

CbvIR systems should be able to automatically extract visual features that are used to describe the contents of an image or video clip. Examples of such features include color, texture, size, shape, and motion information. In specific contexts the process of feature

extraction can be enhanced and/or adapted to detect other, specialized attributes, such as human faces or objects. Because of perception subjectivity, there is no best representation for a given feature (Rui, Huang, and Chang, 1999). The color information, for instance, can be represented using different color models (e.g., RGB, HSV, YCbCr) and mathematical constructs, such as color histograms (Swain and Ballard, 1990), color moments (Stricker and Orengo, 1995), color sets [Smith and Chang, 1996; Smith and Chang, 1995] color coherence vectors (Pass, Zabih, and Miller, 1996), or color correlograms (Huang, Kumar, Mitra, Zhu, and Zabih, 1997). In a similar way, texture can be represented using co-occurrence matrix (Haralick, Shanmugam, and Dinstein, 1973), Tamura texture features (Tamura, Mori, and Yamawaki, 1978) or Wavelets (Chang and Kuo, 1993; Laine and Fan, 1993; Gross, Koch, Lippert, and Dreger, 1994), to name just a few.

Dimension Reduction And Multidimensional Indexing

The extracted features are grouped into some suitable data structure or mathematical construct (e.g., a normalized feature vector), and suitable metrics (e.g., Euclidean distance) are used to measure the similarity between an image and any other image. At this stage, the main challenges are the high dimensionality of the feature vectors (typically of the order of 10^2) and the limitations of Euclidean similarity measure, which although mathematically elegant might not effectively simulate human visual perception [5].

Solutions to the high dimensional indexing problem include reducing the dimension of the feature vectors and the use of efficient multi-dimensional indexing techniques. Dimension reduction is typically obtained using either the Karhunen-Loeve Transform or clustering techniques. Examples of multi-dimensional indexing techniques include specialized data structures (e.g., k-d tree, R-tree and its variants). To overcome the limitations of Euclidean similarity measures, researchers have proposed the use of clustering and neural networks.

Extraction Of Image Semantics

The human perception of visual contents is strongly associated to high-level, semantic information about the scene. Current Computer Vision techniques work at a lower level (as low as individual pixels). CBVIR systems that rely on low-level features only can answer queries such as:

- Find all images that have 30% of red, 10% of orange and 60% of white pixels, where orange is defined as having a mean value of red = 255, green = 130, and blue = 0.
- Find all images that have a blue sky above a green grass.
- Find all images that are rotated versions of this particular image.

In general case, the user is looking for higher-level semantic features of the desired image, such as “a beautiful rose garden”, “a batter hitting a baseball”, or “an expensive sports car”. There is no easy or direct mapping between the low-level features and the high-level concepts. The distance between these two worlds is normally known as “semantic gap.”

Currently there are two ways of minimizing the semantic gap. The first consists of adding as much metadata as possible to the images, which was already discussed and shown to be impractical. The second suggests the use of rich user interaction with relevance feedback combined with learning algorithms to make the system understand and learn the semantic context of a query operation.

Relevance Feedback

Early attempts in the field of CBVIR aimed at fully automated, open-loop systems. It was hoped that current Computer Vision and Image Processing techniques would be good enough for image search and retrieval. The modest success rates experienced by such systems encouraged researchers to try a different approach, emphasizing interactivity and explicitly including the human user in the loop. An example of this shift can be seen in the work of MIT Media Lab researchers in this field, when they moved from the “automated” Photobook (Pentland, Picard, and Sclaroff, 1996) to the “interactive” FourEyes (Minka, 1996).

The expression “relevance feedback” has been used to describe the process by which a system gathers information from its users about the relevance of features, images, image regions, or partial retrieval results obtained so far. Such feedback might be provided in many different ways and each system might use it in a particular manner to improve its performance. The effect of relevance feedback is to “move” the query in the direction of relevant images and away from the non-relevant ones (Gevers and Smeulders, 1999). Relevance feedback has been used in contemporary CBVIR systems, such as MIT’s FourEyes (Minka, 1996). UIUC’s MARS (Rui, Huang, Ortega, and Mehrotra, 1998; Rui, Huang, Mehrotra, and M. Ortega, 1997; Rui, Huang, and Mehrotra, 1997.; Rui, Huang, Mehrotra and Ortega; Rui, Huang, and Mehrotra, 1998; Ortega, Rui, Chakrabarti, Mehrotra, and Huang), and NEC’s PicHunter (Cox, Miller, Minka, Papathomas, and Yianilos, 2000; Cox, Miller, Omohundro, and Yianilos, 1996; Cox, Miller, Omohundro, and Yianilos, 1996; Cox, Miller, Papathomas, Ghosn, and Yianilos, 1997), among others.

In CBVIR systems that support relevance feedback a search typically consists of a query followed by repeated user feedback, where the user comments on the items that were retrieved. The use of relevance feedback makes the user interactions with the system simpler and more natural. By selecting images, image regions, and/or image features, the user is in one way or another telling the system what she wants without the burden of having to describe it using sketches or keywords, for instance.

There are many ways of using the information provided by the user interactions and refining the subsequent retrieval results of a CBVIR system. One approach concentrates on the query phase and attempts to use the information provided by relevance feedback to refine the queries. Another option is to use relevance feedback information to modify feature weights, such as in the MARS project (Rui, Huang, Ortega, and Mehrotra, 1998; Rui, Huang, Mehrotra, and M. Ortega, 1997; Rui, Huang, and Mehrotra, 1997.; Rui, Huang, Mehrotra and Ortega; Rui, Huang, and Mehrotra, 1998; Ortega, Rui, Chakrabarti, Mehrotra, and Huang). A third idea is to use relevance feedback to construct new features on the fly, as exemplified by (Minka and Picard, 1995). A fourth possibility is to use the relevance feedback information to update the probability of each image in a database being the target image, in other words, to predict the goal image given the user’s interactions with the system. The latter is the approach taken by Cox et al. (Cox, Miller, Minka, Papathomas, and Yianilos, 2000; Cox, Miller, Omohundro, and Yianilos, 1996; Cox, Miller, Omohundro, and Yianilos, 1996; Cox, Miller, Papathomas, Ghosn, and Yianilos, 1997) in the PicHunter project.

SYSTEM DESIGN ISSUES

The design of CBVIR systems brings up many interesting problems and challenges, some of which are summarized in (Marques and Furht, 1999). Based on our experience

designing the MUSE system (see Section 6) we have compiled a list of questions that designers of CBVIR systems should attempt to answer before starting to implement their prototypes.

- *Which features should it use and how should they be represented?*
The feature extraction stage is a critical piece of the puzzle. Granted, good feature extraction algorithms alone do not guarantee the overall success of a CBVIR system. However, no system will exhibit a good performance if its knowledge about the images' low-level contents is less than the minimum required to establish the notion of visual similarity between images. Most systems will extract and encode color and texture information. Some systems might also extract frequency-related information, e.g., using mathematical transforms. Specific applications might call for specialized features and algorithms such as face detection. Information about the extracted features is typically organized into feature vectors and distance measurements are used to express similarity between images - the larger the distance the smaller the similarity.
- *How can the system know which features to use or give preference to in a particular query?*
Knowing which (set of) feature(s) to take into account and assigning a particular weight to each as a way of indicating its importance is not an easy task if the CBVIR system works with an unconstrained repository of images. What is very important in one query might be completely irrelevant in the next. Two possible ways of dealing with this issue are: (a) let the user explicitly indicate which features are important before submitting the query; (b) use machine learning techniques to understand the importance of each (set of) feature(s) based on the users' interactions and relevance feedback. The former approach is used by QBIC (Flickner, Sawhney, Niblack, Ashley, Huang, Dom, Gorkani, Hafner, Lee, Petkovic, Steele, and Yanker, 1997; Niblack, Barber, Equitz, Flickner, Glasman, Petkovic, Yanker, Faloutsos, and Taubin, 1993; Ashley, Barber, Flickner, Hafner, Lee, Niblack, and Petkovic, 1995), while the latter is employed in MARS (Rui, Huang, Ortega, and Mehrotra, 1998; Rui, Huang, Mehrotra, and M. Ortega, 1997; Rui, Huang, and Mehrotra, 1997.; Rui, Huang, Mehrotra and Ortega; Rui, Huang, and Mehrotra, 1998; Ortega, Rui, Chakrabarti, Mehrotra, and Huang).
- *Which measure of dissimilarity should it use?*
The most widely adopted similarity model is metric and assumes that human similarity perception can be approximated by measuring the (typically Euclidean) distance between feature vectors. The use of non-Euclidean similarity measures has not yet been deeply explored (Rui, Huang, and Chang, 1999) and research is under way to find better similarity models.
- *Which techniques should it use for dimension reduction and indexing?*
While the Karhunen-Loeve Transform (KLT) is a well-established technique for dimension reduction of the feature vector, the search for an optimal multidimensional indexing technique is still going on and new tree-based approaches have been proposed in the last few years. The survey by Rui, Huang, and Chang (Rui, Huang, and Chang, 1999), contains many pointers to specific algorithms.
- *Which types of queries should it support?*
Deciding upon which query options to support requires a tradeoff between users' needs and preferences and the complexity of implementation behind each mode. Supporting text-based search, for instance, will call for extra effort annotating images

as they are entered into the database while supporting query-by-example (QBE) operations will require more sophisticated measurements of image similarity. Some researchers have claimed that an interesting balance can be achieved combining navigation by categories with content-based search. By the time the user performs a visual query, the subset of images has already been restricted to a particular category, which improves speed (less images need to be considered) and adds semantic knowledge about the query (the category and its parents in the hierarchy tree tell which subject the user is interested in).

- *How to evaluate the quality of the results?*
Benchmarking visual information retrieval solutions is an open problem and the research community is still debating on how to come up with a suite of images, a set of queries, and evaluation criteria for that purpose (C.H.C. Leung and H.H.S. Ip). While a standardized way of comparing two solutions against each other is not yet available, each system relies on its own set of quantitative (e.g., recall, precision, response time) and qualitative measures.
- *Where will the image files be located?*
The knowledge about where the image files are actually stored (in a local hard drive or spread over the Internet) makes a big difference in the design of the system. Among the many issues that need to be taken into account when the images are not stored locally, we can mention:
 - a) the need to store locally either a thumbnail version or a mirrored copy of each image in the remote database;
 - b) the possibility that the actual image files might be (temporarily or permanently) unavailable;
 - c) possible performance degradation caused by network congestion;
 - d) different strategies to update the indexes according to changes in the image repository.
- *How can the user provide relevance feedback and what should the system do with it?*
In CBVIR systems that support relevance feedback, these are very important issues. The first has to do with the user interface and to how do we want the users to interact with the system and express their opinion about the image used as an example (if the system follows a QBE paradigm), the features used to measure similarity, and the partial results obtained so far. While some systems will require minimal action by the user (telling if the results so far are good, bad, or neither), others will ask the user to specify numerical values that convey a measure of goodness of those results. The second issue relates to the complex calculations that take into account the user's relevance feedback information and translate it into an adjustment on the query, the importance of each feature, the probability of each image being the target image, or a combination of those.
- *Which learning capabilities, if any, should the system have?*
CBVIR systems might use unsupervised learning algorithms for a number of reasons:
 - a) to learn how the feature vectors corresponding to each image naturally group together in clusters, and maybe label those clusters;
 - b) to find out which features are useful to categorize an image as belonging to a particular cluster;
 - c) to refine the probability of each image being the desired (target) image based on a set of *a priori* probabilities and the calculations performed so far, taking relevance feedback information into account.

Unsupervised Bayesian learning and clustering are some of the most widely used learning techniques in CBVIR systems.

- *Which supporting tools should the system contain?*

CBVIR systems can be enhanced with a set of supporting tools, such as those suggested in [46]. One example of such tool is a collection of basic image processing functions that would allow users of a QBE-based system to do some simple editing (e.g., cropping, color manipulation, blurring or sharpening, darkening or lightening) on the sample image before submitting their query.

EXAMPLES OF CBVIR SYSTEMS

Numerous CBVIR systems, both commercial and research, have been developed in recent years. Some of the currently available CBVIR systems are briefly described below. More details can be obtained by following the pointers to Web sites and bibliography.

QBIC

QBIC (Query By Image Content) (Flickner, Sawhney, Niblack, Ashley, Huang, Dom, Gorkani, Hafner, Lee, Petkovic, Steele, and Yanker ,1997 ; Niblack, Barber, Equitz, Flickner, Glasman, Petkovic, Yanker, Faloutsos, and Taubin ,1993; Ashley, Barber, Flickner, Hafner, Lee, Niblack, and Petkovic ,1995) was developed by IBM Almaden Research Center. Its framework and techniques have influenced many later systems. QBIC supports queries based on example images, user-constructed sketches, and selected colors and texture patterns. In its most recent version, it allows text-based keyword search to be combined with content-based similarity search. The online QBIC demo can be found at: <http://www.qbic.almaden.ibm.com>.

Photobook

Photobook (Pentland, Picard, and Sclaroff ,1996) is a set of interactive tools for browsing and searching images developed at MIT Media Lab. Photobook consists of three sub-books, from which shape, texture, and face features are extracted respectively. Users can query the system based on features from each of the three sub-blocks. Additional information about Photobook can be found at: <http://www-white.media.mit.edu/vismod/demos/photobook/index.html>.

FourEyes

FourEyes (Minka ,1996) is an improved version of Photobook that includes user relevance feedback. Given a set of positive and negative examples, it decides upon which models or combinations of models to use and learns which combinations work best for solving particular types of problems. When presented with a new problem similar to one it has solved before, FourEyes can solve it more quickly than it could the first time. More details about the system can be found at: <http://www-white.media.mit.edu/vismod/demos/photobook/foureyes/>.

Netra

Netra is a prototype CBVIR system developed in the UCSB Alexandria Digital Library (ADL) project (Deng and Manjunath ,1998). It uses color, shape, texture, and spatial

location information in the segmented image regions to search and retrieve similar images from the database. An online demo is available at <http://vivaldi.ece.ucsb.edu/Netra/>. A new version of Netra, Netra 2, which emphasizes the group's latest work on color image segmentation and local color feature, is available at <http://maya.ece.ucsb.edu/Netra/index2.html>.

MARS

MARS (Multimedia Analysis and Retrieval System) (Rui, Huang, Ortega, and Mehrotra ,1998; Rui, Huang, Mehrotra, and M. Ortega ,1997; Rui, Huang, and Mehrotra,1997.; Rui, Huang, Mehrotra and Ortega; Rui, Huang, and Mehrotra ,1998; Ortega, Rui, Chakrabarti, Mehrotra, and Huang) was originally developed at University of Illinois at Urbana-Champaign. The main focus of MARS is not on finding a single “best” feature representation, but rather on how to organize the various visual features into a meaningful retrieval architecture, which can dynamically adapt to different applications and different users. MARS formally proposes a relevance feedback architecture in Image Retrieval and integrates such technique at various levels during retrieval, including query vector refinement, automatic matching tool selection, and automatic feature adaptation. More information about MARS can be obtained at: <http://www-db.ics.uci.edu/pages/research/mars.shtml>.

PicToSeek

PicToSeek (Gevers and Smeulders ,1999) is an image search engine developed at University of Amsterdam. PicToSeek uses autonomous Web crawlers to collect images on the Web. Then, the collected images are automatically catalogued and classified into predefined classes and their relevant features are extracted. The users can query PicToSeek using image features, an example image, or simply browsing the precomputed image catalog. A demo version of PicToSeek is available at: <http://www.wins.uva.nl/research/isis/zomax/>.

VisualSEEk

VisualSEEk (Smith and Chang ,1996; Smith and Chang ,1997) is part of a family of CBVIR systems developed at Columbia University. It supports queries based on both visual features and their spatial relationships. An online demo is available at: <http://www.ctr.columbia.edu/VisualSEEk/>.

PicHunter

PicHunter (Cox, Miller, Minka, Papathomas, and Yianilos ,2000; Cox, Miller, Minka, Papathomas, and Yianilos ,1996; Cox, Miller, Minka, Papathomas, and Yianilos ,1996; Cox, Miller, Minka, Papathomas, and Yianilos ,1997; Cox, Miller, Minka, Papathomas, and Yianilos ,1998) is a CBVIR system developed at NEC Research Institute, New Jersey. PicHunter uses relevance feedback and Bayes's rule to predict the goal image given the users' actions.

ImageRover

ImageRover (ImageRover home page; Sclaroff, Taycher, and Cascia ,1997) is a CBVIR system developed by Boston University that is currently available as an online demo

version. This is a Web-based tool, which gathers information about HTML pages via a fleet of automated robots. These robots gather, process, and store the image metadata in a vector format that is searched when a user queries the system. The user then receives relevance feedback with thumbnail images, and by selecting the relevant images to their search, can utilize the content-based searching capabilities of the system until they find their desired target image. More details are available at <http://cs-pub.bu.edu/groups/ivc/ImageRover/Home.html>.

WebSEEk

WebSEEk (Smith and Chang) is similar to ImageRover in its HTML collection processes through Web robots, though it has the advantage of video search and collection as well. It was developed at Columbia University, and currently has a working demo available on the Web at <http://www.ctr.columbia.edu/webseek/>. The user receives relevance feedback in the form of thumbnail images and motion icons or spatially and temporally reduced video forms given as short GIF files to the user.

Virage

Virage (Bach, Fuller, Gupta, Hampapur, Horowitz, Humphrey, Jain, and Shu,) is a commercial content-based image search engine developed at Virage, Inc. Virage supports queries based on color, composition (color layout), texture, and structure (object boundary information) in any arbitrary combination. The users inform the system which weight should be associated with each atomic feature according to their own emphasis. More information about Virage products can be found at: <http://www.virage.com>.

Visual RetrievalWare

Visual RetrievalWare is a CBVIR engine developed by Excalibur Technologies Corp. (Dowe, 1993). Similarly to Virage, it allows combinations of several visual query features, whose weights are specified by the users. At the end of 2000, Excalibur became Convera. More information about Convera products can be found at: <http://www.convera.com/>.

AMORE

AMORE (Advanced Multimedia Oriented Retrieval Engine) is a search engine with image retrieval capabilities developed by the C & C Research Laboratories (CCRL), a division of NEC USA. It does not have the ability to search the entire Web via automated robots, but it does have an automated robot (or *harvest gatherer* as they call it) which can scour and classify images from user specified URLs. The system uses the *Harvest Information Discovery and Access System* for text indexing and searching, and the content-oriented image retrieval (COIR) to index the images and retrieve them. COIR uses a region-based approach, using attributes like color, texture, size and position for indexing. Everything but the entry of the URL addresses is automated for the user. More information about AMORE can be found at: <http://www.crl.com/amore>.

Blobworld

Blobworld (Blobworld home page) is a CBVIR system developed at U.C. Berkeley. The program automatically segments an image into regions, which roughly correspond to object or parts of objects allowing users to query for photographs or images based on the

objects they contain. Their approach is useful in finding specific objects and not, as they put it, “stuff” as most systems which concentrate only on “low level” features with little regard for the spatial organization of those features. It allows for both textual and content-based searching.

This system is also useful in its feedback to the user, in that it shows the internal representation of the submitted image and the query results. Thus, unlike some of the other systems, which allow for color histogram similarity metrics, which can be adjusted, this can help the user understand why they are getting certain results.

Other companies and products

Many companies that have entered the newly-created market of visual search solutions in the past few years. Some of these companies are: **Ereo** (<http://www.ereo.com>), **Cobion** (<http://www.cobion.com>), **LookThatUp.com**, and **ImageLock** (<http://www.ImageLock.com>).

OPEN RESEARCH PROBLEMS AND FUTURE DIRECTIONS

Visual Information Retrieval is a very active research field and many open problems are still being investigated. Some of the most prominent technical challenges and research opportunities include (Bimbo, 1999; Rui, Huang, and Chang, 1999; Chang, Eleftheriadis, and McClintock, 1998; Bimbo, 1998; Petkovic, 1998; Aigrain, Zhang, and Petkovic, 1996):

- *Better exploration of the synergy of human and computer*
It is recognized that CBVIR systems will only achieve acceptable performance if they include the human users in the loop and allow them to provide relevance feedback information. The details of how should the users provide relevance feedback and what the system should do with them are still being investigated.
- *Minimizing the semantic gap between the image's low-level features and the human interpretation of the image contents*
The disparity between the high-level concepts behind a visual query and the associated low-level features extracted from the images using current Computer Vision techniques is known as “semantic gap” in the literature. The most promising way to minimize this gap is to use off-line learning algorithms combined with online relevance feedback information.
- *Make the systems Web-oriented*
The popularity of text-based search engines for Web-based search has not yet been matched by similar abilities to perform visual searches. The lack of a standard for metadata representation (to be solved with the advent and adoption of the MPEG-7 standard) and the impact of the maximum acceptable response time on the overall performance of the system are two major hurdles yet to be overcome.
- *High dimensional indexing*
Research on effective high dimensional indexing techniques with the capability of supporting non-Euclidean similarity measures and adapting themselves to changes in similarity functions at run time is very active and no final solution has been found.
- *Standardization of performance evaluation criteria and creation of standard benchmark suites and testbeds*

There is a recognized need to adopt a standardized set of images, queries, and performance measures to allow different solutions to be compared against each other (Leung and Ip). One of the International Association for Pattern Recognition's (IAPR) Technical Committees (TC 12) has been working on this issue and at the moment of this writing there has been no final decisions yet.

- *Human perception of image contents*
A deeper investigation of the psychophysical aspects of human visual perception should provide additional insight on how the human mind judges visual similarity and help improve the performance of CBVIR systems without precluding the inclusion of the human user in the loop.
- *New visual interfaces for image/video database access*
Improved ways of querying, browsing, and navigating a visual information repository are needed, particularly when video is involved.
- *Integration of Computer Vision with other disciplines and media.*
Development of successful image database systems will require collaboration between researchers from the Computer Vision community (that deal with Image non-databases) and Database community (whose systems are typically non-image) and additional research fields.

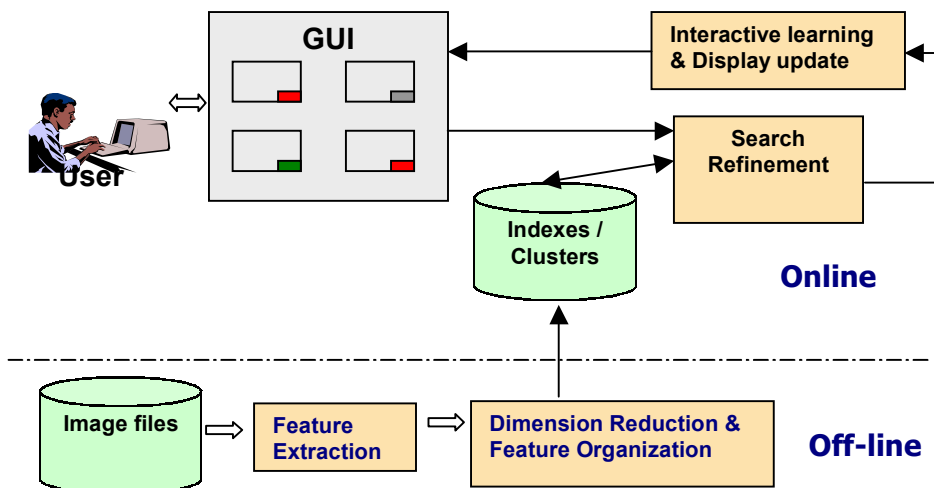
MUSE: A CONTENT-BASED IMAGE RETRIEVAL SYSTEM WITH RELEVANCE FEEDBACK

Background

For the past two years the authors have been working on MUSE, a CBVIR system with relevance feedback and learning capabilities.

The goal of this project is to build an intelligent system for searching and retrieving visual information in large repositories. Some of its objectives include:

Figure 3. MUSE: block diagram.



- Clean, simple, friendly user interface
- Ability of learning from user interaction
- User transparency: the complexity of the underlying search and retrieval engine should be hidden from the user
- Extensibility to other types of media, particularly video.

Overview Of The System

Figure 3 shows the main components of MUSE. Part of the system's operations happen off-line while some actions are executed online. The off-line stage includes feature extraction, representation, and organization for each image in the archive. The online interactions are commanded by the user through the GUI. The relevant images selected by the user have their features extracted and compared against all the other images' features. The result of the similarity comparison is the update and ranking of each image's probability of being the target image. Based on them, the system stores learning information and decides on which candidate images to display next. After a few iterations, the target image should be among those displayed on screen.

The User's Perspective

MUSE's interface is simple, clear, and intuitive (see Figure 4). It contains a menu, two tool bars and a working area divided in two parts: the left-hand side contains a selected image (optional) and the right-hand side works as a browser, whose details depend on the operation mode. MUSE supports four operation modes: free browsing, random ("slot machine") browsing, query-by-example, and relevance feedback (without an example image). In the free browsing mode (Figure 5), the browser shows thumbnail versions of the images in the currently selected directory. The random mode (Figure 6) shuffles the directory contents before showing the reduced-size version of its images, working as a baseline against which the fourth mode (relevance feedback) can be compared. The query-by-example mode (Figure 7) has been implemented to serve as a testbed for the feature extraction and similarity measurement stages. Using an image (left) as an example, the best matches (along with their normalized scores) are shown in the browser area. Last, but not least, the relevance feedback mode starts from a random subset of images and refines its understanding of which image is the target based on the user input (specifying each image as *good*, *bad*, or *neither*).

In a typical session using the relevance feedback mode, the user would initially see a

Figure 4. MUSE: the user interface.



subset of images on the browser side (Figure 8). Based on how similar or dissimilar each image is when compared to the target image (the Canadian flag, in this example), the user can select zero or more of the currently displayed images as “good” or “bad” examples before pressing the GO button. The select button associated with each image changes its color to green (when the image has been chosen as a good example) or red (when the image has been chosen as a bad example). Selecting relevant images and pressing the GO button are the only required user actions. Upon detecting that the GO button has been pressed, MUSE first verifies if one or more images have been selected. If so, it recalculates the probabilities of each image being the target image and displays a new subset of images that should be closer to the target than the ones displayed so far (Figure 9). If the user has not selected any image, the system displays four new images selected at random. After a few iterations, the system eventually converges to the target image (in this example, only one

Figure 5. MUSE: free browsing mode.



Figure 6. MUSE: random (“slot machine”) browsing mode.



Figure 7. MUSE: query-by-example mode.



Figure 8. MUSE: relevance feedback mode: initial screen.



Figure 9. MUSE: relevance feedback mode: best results after one iteration.



iteration beyond the initial images was needed).

BEHIND THE SCENES

The current prototype of MUSE supports only color-based features but the final version of MUSE is expected to support a subset of color-related features, a subset of texture-related features, and a subset of shape-related features. Color information is extracted using color correlograms (Huang, Kumar, Mitra, Zhu, and Zabih, 1997) and the correlograms of two images are compared using the L_1 distance measure. The resulting feature vectors are organized into clusters using a variation of the PAM (Partitioning Around Medoids) algorithm (Kaufman and Rousseeuw, 1990).

MUSE uses a probabilistic model of information retrieval based on image similarity. In this model, each image is initially assigned a probability of being the target image. The probability of each image being the target is recalculated every time the user selects good and/or bad examples, based on the distance between each image and the samples labeled as *good* or *bad*. At the end of each iteration, the probability distribution of all the images in the database is normalized and the best candidates from the most likely clusters are displayed back.

MUSE supports two types of learning. The very process of updating image probabilities during a session is in itself a way of learning the user's preferences and responding accordingly. This type of learning is what we call *intra-session learning*. MUSE is being expanded to also support *inter-session learning*, i.e., ways of using the information learned from user's interaction during one session to improve performance in similar situations in future sessions. One way to accomplish this is to allow the user to save / restore profiles. Every time the user logs into the system she would be offered the possibility of retrieving an existing login profile, starting a new profile, or ignoring profiles altogether. Examples of profiles could be as diverse and specific as: "sports car lover", "Sharon Stone fan", or "flags of the world". By logging their profiles user would indirectly provide semantic level information to the system with minimal extra burden, namely saving / restoring a profile once per session. MUSE uses Bayesian network models to assess the users' needs and profiles based on their interactions with the system. More details about MUSE's algorithms and mathematical formulation can be found in (Marques and Furht, 2002).

Possible Applications

Most of the ideas developed in this project should be general enough for any type of visual information retrieval need. During its development MUSE is being tested with general image archives as well as specialized image repositories. A possible specific application for the results of this project is management of family pictures. Since family pictures are increasingly becoming available in digital format thanks to the popularization and price drop in scanners and digital cameras, there seems to be a potentially big market for home users who want to manage and catalog their digitized pictures in an easy and intelligent way (Kuchinsky, Pering, Creech, Freeze, Serra and Gwizdka, 1999; Platt, 2000). Such a system would help minimizing the well-known phenomenon of pictures that are never retrieved, organized, and therefore enjoyed, ending up in a shoe box, or its digital equivalent, a folder in the home PC's hard disk. Improving the system's ability to deal with family picture-related situations, e.g., face detection, indoor versus outdoor classification, among others, is a possible specialization avenue we might want to pursue in future versions of

MUSE.

CONCLUSIONS

In this chapter we provided an up-to-date review of content-based visual information retrieval systems. We presented the system architecture of a CBVIR system and addressed open issues in designing and developing these systems. The potential market for CBVIR solutions has attracted many companies and universities and several commercial and research prototypes are now available. Some of these companies, research groups, and prototypes are described along the chapter with many pointers for further information for the interested reader. At the end we described the details of a CBVIR prototype being developed by the authors at Florida Atlantic University.

REFERENCES

- Del Bimbo, A. (1998). "A Perspective View on Visual Information Retrieval Systems", *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries*, Santa Barbara, California.
- Del Bimbo, A. (1999). *Visual Information Retrieval*, Morgan Kaufmann, San Francisco, CA,
- A. Gupta and R. Jain (1997). "Visual Information Retrieval," *Communications of the ACM*, 40(5).
- A. Gupta, S. Santini, and R. Jain (1997). "In Search of Information in Visual Media," *Communications of the ACM*, 40(12).
- A. Kuchinsky, C. Pering, M.L. Creech, D. Freeze, B. Serra and J. Gwizdka (1999). "FotoFile: a consumer multimedia organization and retrieval system", *Proceeding of the CHI 99 conference on Human factors in computing systems: the CHI is the limit*, May 15 - 20, Pittsburgh, PA USA.
- A. Laine and J. Fan (1993). "Texture Classification by Wavelet Packet Signatures," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, Vol. 15, No. 11.
- A. Pentland, R.W. Picard, and S. Sclaroff (1996). "Photobook: Content-Based Manipulation of Image Databases," Chapter 2 in *Multimedia Tools and Applications*, Furht, B., Ed., Kluwer Academic Publishers, Boston.
- Blobworld home page. <http://www.cs.berkeley.edu/~carson/blobworld/>.
- C.H.C. Leung and H.H.S. Ip, *Benchmarking for Content-Based Visual Information Search*.
- D. Petkovic (1998). "Challenges and Opportunities in Search and Retrieval for Media Databases", *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries*, Santa Barbara, California.
- G. Pass, R. Zabih, and J. Miller (1996). "Comparing Images Using Color Coherence Vectors," *Proc. ACM Conference on Multimedia*.
- H. Tamura, S. Mori, and T. Yamawaki (1978). "Texture Features Corresponding to Visual Perception", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 8, No. 6.
- I.J. Cox, M.L. Miller, S.M. Omohundro, and P.N. Yianilos (1996). "PicHunter: Bayesian Relevance Feedback for Image Retrieval," *Proc. Int. Conference on Pattern Recognition*, Vienna, Austria.
- I.J. Cox, M.L. Miller, S.M. Omohundro, and P.N. Yianilos (1996). "Target Testing and the PicHunter Bayesian Multimedia Retrieval System", *Advanced Digital Libraries ADL '96*

- Forum, Washington D.C.
- I.J. Cox, M.L. Miller, T. Papathomas, J. Ghosn, and P.N. Yianilos (1997). "Hidden Annotation in Content Based Image Retrieval," Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries.
- I.J. Cox, M.L. Miller, T.P. Minka, T. Papathomas, and P.N. Yianilos (2000). "The Bayesian Image Retrieval System, PicHunter: Theory, Implementation and Psychophysical Experiments," IEEE Transactions on Image Processing, Vol. 9, pp. 20-37, January.
- I.J. Cox, M.L. Miller, T.P. Minka, T. Papathomas, and P.N. Yianilos (1998). "An Optimized Interaction Strategy for Bayesian Relevance Feedback," Proceedings of the International Conference on Computer Vision and Pattern Recognition.
- ImageRover home page. <http://www.cs.bu.edu/groups/ivc/ImageRover/Home.html>.
- J. Ashley, R. Barber, M. Flickner, J. Hafner, D. Lee, W. Niblack, and D. Petkovic (1995). "Automatic and Semi-Automatic Methods for Image Annotation and Retrieval in QBIC," Research Report RJ 9951 (87910), IBM Research Division, Almaden Research Center.
- J. Dowe (1993). "Content-Based Retrieval in Multimedia Imaging," Proc. SPIE Conference on Storage and Retrieval for Image and Video Databases.
- J. Huang, S. Kumar, M. Mitra, W.-J. Zhu, R. and Zabih (1997). "Image Indexing Using Color Correlogram," Proc. IEEE International Conference on Computer Vision and Pattern Recognition.
- J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih. (1997). "Image indexing using color correlograms", Proc. IEEE Comp. Soc. Conf. Comp. Vis. and Patt. Rec., pp. 762-768.
- J. R. Smith and S.-F.Chang (1996). "Tools and Techniques for Color Image Retrieval," Proc. SPIE Conference on Storage and Retrieval for Image and Video Database IV, 2670, San Jose, CA, February.
- J. R. Smith and S.-F.Chang, S.-F. (1995). "Single Color Extraction and Image Query," Proc. IEEE International Conference on Image Processing.
- J.C. Platt (2000). "AutoAlbum: Clustering Digital Photographs Using Probabistic Model Merging", in Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries 2000.
- J.R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C. Shu, "The Virage Image Search Engine: An Open Framework for Image Management," Proc. SPIE Conference on Storage and Retrieval for Image and Video Databases.
- J.R. Smith and S. Chang, "Searching for Images and Videos on the World-Wide Web," Center for Telecommunications Research Technical Report #459-96-25, <http://www.ctr.columbia.edu/webseek/paper/>.
- J.R. Smith and S.-F.Chang (1996). "VisualSEEk: A Fully Automated Content-Based Image Query System," Proc. ACM Multimedia '96, Boston, MA, November.
- J.R. Smith and S.-F.Chang (1997). "Querying by Color Regions Using the Visual SEEk Content-Based Visual Query System," in Intelligent Multimedia Information Retrieval, Maybury, M. T., Ed., American Association for Artificial Intelligence (AAAI), Menlo Park, CA.
- L. Kaufman and P.J. Rousseeuw (1990). Finding Groups in Data: an introduction to cluster analysis, Wiley.
- M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker (1997). "Query by Image and Video Content: The QBIC System," in Intelligent Multimedia Information Retrieval, Maybury,

- M. T., Ed., American Association for Artificial Intelligence (AAAI), Menlo Park, CA.
- M. H. Gross, R. Koch, L. Lippert, and A. Dreger (1994). "Multiscale Image Texture Analysis in Wavelet Spaces," Proc. IEEE International Conference on Image Processing.
- M. Ortega, Y. Rui, K. Chakrabarti, S. Mehrotra, and T.S. Huang, "Supporting Similarity Queries in MARS", Proc. of ACM Multimedia, pp. 403-413, 1997.
- M. Stricker and M. Oren (1995). "Similarity of Color Images," Proc. SPIE Storage and Retrieval for Image and Video Databases.
- M.J. Swain and D.H. Ballard (1990). "Indexing via Color Histograms", Proc. Third International Conference on Computer Vision.
- O. Marques and B. Furht (1999). "Issues in Designing Contemporary Video Database Systems," Proc. IASTED Conference on Internet and Multimedia Systems and Applications, Nassau, Bahamas, October.
- O. Marques and B. Furht (2002). "MUSE: A Content-Based Image Search and Retrieval System Using Relevance Feedback", Multimedia Tools and Applications, Kluwer Academic Publishers. (to appear)
- P. Aigrain, H. J. Zhang, and D. Petkovic (1996). "Content-Based Representation and Retrieval of Visual Media: A State-of-the-Art Review", Multimedia Tools and Applications, 3, 3.
- R. Baeza-Yates, and B. Ribeiro-Neto (1999). Modern Information Retrieval, Addison-Wesley / ACM Press, New York.
- R. Haralick, K. Shanmugam, and I. Dinstein (1973). "Texture Features for Image Classification," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 3, No. 6,
- S. Sclaroff, L. Taycher, and M. La Cascia (1997). "ImageRover: A Content-based Image browser for the World Wide Web," Proc. IEEE Workshop on Content-based Access of Image and Video Libraries, June.
- S.-F. Chang, A. Eleftheriadis, and R. McClintock (1998). "Next-generation Content Representation, Creation and Searching for New Media Applications in Education", Proceedings of the IEEE.
- S.-F. Chang, J. R. Smith, M. Beigi, and A. Benitez (1997). "Visual Information Retrieval from Large Distributed Online Repositories," Communications of the ACM, Vol. 40, No. 12, December.
- T. Chang and C.-C. Jay Kuo (1993). "Texture Analysis and Classification with Tree-Structured Wavelet Transform," IEEE Transactions on Image Processing, Vol. 2, No. 4.
- T. Gevers and A. W. M. Smeulders (1999). "The PicToSeek WWW Image Search System," Proceedings of the International Conference on Multimedia Computing and Systems, Florence, Italy.
- T. Minka (1996). "An Image Database Browser that Learns from User Interaction," MEng Thesis, MIT.
- T.P. Minka and R. Picard (1995). "Interactive learning using a 'society of models'," MIT Media Laboratory Perceptual Computing Section Technical Report No. 349.
- W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin (1993). "The QBIC Project: Querying Images By Content Using Color, Texture and Shape," Proc. SPIE Storage and Retrieval for Image and Video Databases, pp. 173-187.
- Y. Deng and B.S. Manjunath (1998). "NeTra-V: Toward an Object-Based Video Representation," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 8, No. 5.
- Y. Rui, T. S. Huang, and S. Mehrotra (1997). "Content-Based Image Retrieval with

- Relevance Feedback in MARS,” Proc. IEEE Int. Conf. Image Processing.
- Y. Rui, T. S. Huang, and S. Mehrotra (1998). “Relevance Feedback Techniques in Interactive Content-Based Image Retrieval,” Proc. S&T SPIE Storage and Retrieval of Images/ Video Databases VI, EI’98.
- Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra (1998). “Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 8, No. 5, pp. 644-655.
- Y. Rui, T. S. Huang, S. Mehrotra, and M. Ortega (1997). “A Relevance Feedback Architecture for Content-Based Multimedia Information Retrieval Systems,” Proceedings of IEEE Workshop on Content-based Access of Image and Video Libraries, pp. 82-89.
- Y. Rui, T. S. Huang, S. Mehrotra, and M. Ortega, “Automatic Matching Tool Selection Using Relevance Feedback in MARS”, Proc. 2nd Int. Conf. Visual Information Systems.
- Y. Rui, T.S. Huang, and S.-F. Chang (1999). “Image Retrieval: Current Techniques, Promising Directions, and Open Issues”, Journal of Visual Communication and Image Representation, Vol. 10, pp. 39-62, March.