Content-Based Retrieval (CBR)

- *In Multimedia Systems, a mini-handbook*

Author: Chao Cai
ID: 0227216
Date: 03/31/2006

Table of Contents

Abstract

This paper explores one of the growing research areas in multimedia systems: Content-Based Retrieval. The paper reviews a number of recently available techniques used in Content-Based Retrieval for a various multimedia types. Some CBR systems are introduced and future work will be presented

1.0 Background

1.1 History & Overview

With the recent developments in multimedia and telecommunication technologies, content-based information is becoming increasingly important for various areas such as digital libraries, interactive video, and multimedia publishing. Thus, there is a clear need for automatic analysis tools which are able to extract representation data from the documents.

The researchers involved in content processing efforts come from various backgrounds, for instance:

- the publishing, entertainment, retail or document industry where researchers try to extend their activity to visual documents, or to integrate them in hypertext-based new document types,

- the AV hardware and software industry, primarily interested by digital editing tools and other programma production tools,

- academic laboratories where research had been conducted for some time on computer analysis and access to existing visual media,

- large telecommunication company laboratories, where researchers are primarily interesting in cooperative work and remote access to visual media,

- the robotics vision, signal processing, image sequence processing for security, or data compression research communities who try to find new applications for their models of images or human perception.

- computer hardware manufacturers developing digital library or visual media research programs.

1.2 Digital Library

Evolution has been taken from small databases, to image databases and now onto digital libraries for multimedia storage, representation and retrieval. A **digital library** is a library in which a significant proportion of the resources are available in machine-readable format (as opposed to print or microform), accessible by means of computers. The digital content may be locally held or accessed remotely via computer networks. In libraries, the process of digitization began with the catalog, moved to periodical indexes and abstracting services, then to periodicals and large reference works, and finally to book publishing.

Advantages of Digital Library:

- **No physical boundary**. People from all over the world can gain access to the same information.

- **Multiple accesses**. The same resources can be used at the same time by a number of users.

- **Information retrieval**. The user is able to use any search term bellowing to the word or phrase of the entire collection. Digital library can provide very user friendly interfaces, giving click able access to its resources.

- **Space**. Whereas traditional libraries are limited by storage space, digital libraries have the potential to store much more information, simply because digital information requires very little physical space to contain them.

- **Cost**. In theory, the cost of maintaining a digital library is lower than that of a traditional library.

Retrieval in Digital Library:

Digital libraries must store and retrieve multimedia data on the basis of feature similarity. A feature is a set of characteristics. Content-based retrieval uses content-representative metadata to both store data and retrieve it in response to user queries.

1.3 Metadata

Metadata is data about the media objects stored. Manually collecting metadata is not only inefficient but also infeasible for large document spaces, so we need automatic metadata generation. Once collected, these content descriptors are linked to the physical location of data. Data storage strategies are key to efficient retrieval.

Metadata Classification:

- **Content-dependent.** Metadata based on some characteristics specific to the content of the media objects. For example, text strings in text documents; the color, texture, and position of objects in an image; and individual frame characteristics, such as color histograms, for video objects.

- **Content-descriptive.** Metadata that is not based on the content. For example, names of authors and years of publication.

- **Content-independent.** Metadata that describes the characteristics of the media objects but cannot be generated automatically. For example, image characteristics like the mood reflected by a facial expression and camera shot distance.

2.0 Content-Based Retrieval of Image (CBIR)

2.1 Similarities

Retrieval of still images by similarity, i.e. retrieving images which are similar to an already retrieved image (retrieval by example) or to a model or schema is retrieval by similarity. From the start, it was clear that retrieval by similarity called for specific definitions of what it means to be similar.

A system for retrieval by similarity rest on **3 components**:

- extraction of features or image signatures from the images, and an efficient representation and storage strategy for this pre-computed data,

- a set of similarity measures, each of which captures some perceptively meaningful definition of similarity, and which should be efficiently computable when matching an example with the whole database,

- a user interface for the choice of which definition(s) of similarity should be applied for retrieval, and for the ordered and visually efficient presentation of retrieved images and for supporting relevance feedback.

2.2 Color Similarity

Concept:

Color distribution similarity has been one of the first choices because if one chooses a proper representation and measure it can be partially reliable even in presence of changes in lighting, view angle, and scale. For the capture of properties of the global color distribution in images, the need for a perceptively meaningful color model leads to the choice of HLS (Hue-Luminosity-Saturation) models, and of measures based on the 3first moments of color distributions preferably to histogram distances.

Difficulty:

One important difficulty with color similarity is that when using it for retrieval, an user will often be looking for an image "with a red object such as this one". This problem of restricting color similarity to a spatial component, and more generally of combining spatial similarity and color similarity is also present for texture similarity. It explains why prototype and commercial systems have included complex ad-hoc mechanisms in their user interfaces to combine various similarity functions.

Case-Study:

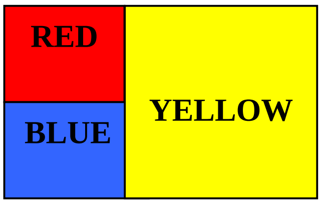


Image1

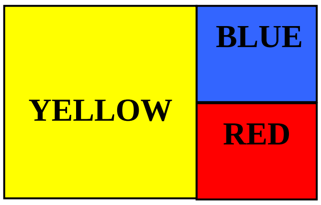


Image2

Image 1 and Image 2 are the same size and are filled with solid colors.

In Image 1, the top left quarter (25%) is red, the bottom left quarter (25%) is blue, and the right half (50%) is yellow. In Image 2, the top right quarter (25%) is blue, the bottom right quarter (25%) is red, and the left half (50%) is yellow.

If the two images are compared first solely on color and then color and location, the following are the similarity results:

Color: complete similarity (score = 0.0), because each color (red, blue, yellow) occupies the same percentage of the total image in each one

Color and location: no similarity (score = 100), because there is no overlap in the placement of any of the colors between the two images

Thus, if you need to select images based on the dominant color or colors (for example, to find apartments with blue interiors), give greater relative weight to color. If you need to find images with common colors in common locations (for example, red dominant in the upper portion to find sunsets), give greater relative weight to location.

2.3 Texture Similarity

Concept:

For texture as for color, it is essential to define a well-funded perceptive space. It is possible to do so using the Wold decomposition of the texture considered as a luminance field. One gets three components(periodic, evanescent and random) corresponding to the bi-dimensional periodicity, mono-dimensional orientation, and complexity of the analyzed texture. Experiments have shown that these independent components agree well with the perceptive evaluation of texture similarity. The related similarity measures has lead to remarkably efficient results including for the retrieval of large-scale textures such as images of buildings and cars.

Difficulty:

As for color, one important difficulty with texture similarity is that when using it for retrieval, an user will often be looking for an image “with a texture such as this one”. This problem of restricting texture similarity to a spatial component, and more generally of combining spatial similarity and texture similarity. It explains why prototype and commercial systems have included complex ad-hoc mechanisms in their user interfaces to combine various similarity functions. So of course one is again confronted to the problem of combining texture information with the spatial organization of several textures.

Case-Study:

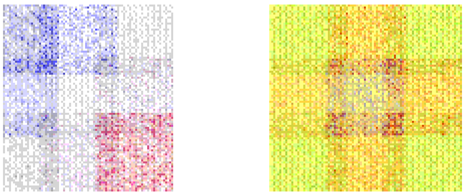


Figure 2.3 Fabric Images with Similar Texture

Texture reflects the texture of the entire image. Texture is most useful for full images of textures, such as catalogs of wood grains, marble, sand, or stones. These images are generally hard to categorize using keywords alone because our vocabulary for textures is limited. Texture can be used effectively alone (without color) for pure textures, but also with a little bit of color for some kinds of textures, like wood or fabrics. Figure 2.3 shows two similar fabric samples.

2.4 Shape Similarity

Concept:

A proper definition of shape similarity calls for the distinctions between shape similarity in images (similarity between actual geometrical shapes appearing in the images) and shape similarity between the objects depicted by the images, i.e. similarity modulo a number of geometrical transformations corresponding to changes in view angle, optical parameters and scale. In the general case, a promising approach has been proposed in which shapes are represented as canonical deformations of prototype objects. In this approach, a "physical" model of the 2D-shape is built using a new form of Galerkin's interpolation method (finite-element discretization). The possible deformation modes are analyzed using Karhunen-Loeve transform. This yields an ordered list of deformation modes corresponding to rigid body modes (translation, rotation), low-frequency non-rigid modes associated to global deformations and higher-frequency modes associated to localized deformations.

Difficulty:

As for color and texture, the present schemes for shape similarity modeling are faced with serious difficulties when images include several objects or background. A preliminary segmentation as well as modeling of spatial relationships between shapes is then necessary (are we interested in finding images where one region represent a shape similar to a given prototype or to some spatial organization of several shapes?).

Case-Study:

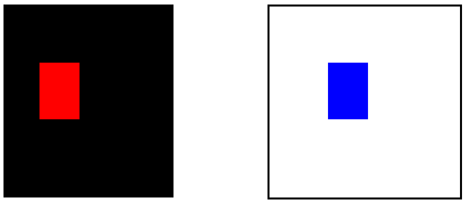


Figure 2.4 Images with Very Similar Shape

Shape represents the shapes that appear in the image. Shapes are determined by identifying regions of uniform color.

Shape is useful to capture objects such as horizon lines in landscapes, rectangular shapes in buildings, and organic shapes such as trees. Shape is very useful for querying on simple shapes (like circles, polygons, or diagonal lines) especially when the query image is drawn by hand and color is not considered important when the drawing is made. Figure 2.4 shows two images very close in shape.

2.5 Spatial Similarity

Concept:

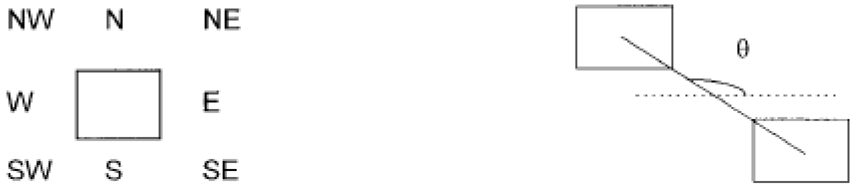
Spatial similarity assumes that images have been (automatically or manually) segmented into meaningful objects, each object being associated with is centroid and a symbolic name. This representation is called a *symbolic image*, and it is relatively easy to define similarity functions for such image modulo transformations such as rotation, scaling and translation. Also, addressing spatial similarity directly (without segmentation and object indexing) is the case, in the limited case of direct spatial similarity (without geometrical transformation), using a number of ad-hoc statistical features computed on very low resolution images.

Difficulty:

Spatial similarity does not stand-alone and usually involves object presence analysis to the image. Finding in a set of images in which a particular object or type of object appears is a particular case of similarity computation. Once again, the range of applicable methods is defined by the invariants of the object to be recognized.

Case-Study:

Spatial relations may be classified into **directional** & **topological** relations.



(a) strict and mixed directional relations

(b) slope directional relations

Figure 2.5a Directional Relations

The frequently used directional relations are the *strict* directional relations: north, south, east, and west, as shown in Figure 2.5a (a).

Another directional relation specifies the directional relation between two objects as the slope of the line joining their centroids, as shown in Figure 2.5a (b) called *slope* directional relations.

Directional relations are not sufficient for characterizing spatial similarity because they only consider the spatial orientation of an object while ignoring its spatial extent. In some cases, directional relations do not exist, while in other cases directional relationships may be identical in two images in spite of the fact that the images are not spatially identical. In addition, directional relations are not rotation invariant.

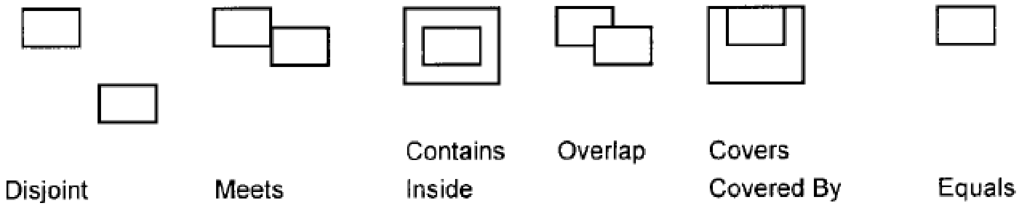


Figure 2.5b Topological Relations

Topological relations, on the other hand, always exist between any two objects. Also, topological relations are mutually exclusive, i.e., there is one and only one topological relation between any two objects in an image. Another interesting feature of topological relations is that they are preserved under perfect translation, scaling, or rotation transformations.

There are eight fundamental topological relations that can hold between two planar regions. These relations are disjoint, contains, inside, meet, equal, covers, covered-by, and overlap (Figure 2.5b).

2.6 Future Work

Low Level v.s **High Level:**

Extraction of low level visual features and establishment of related search/matching functions, extraction of higher (semantic) level image attributes (such as recognition of object, human faces and actions) and related search/matching functions are definitely a more challenging task. Only when the features extracted at both these levels are combined, can content-based image indexes be built.

Formalism:

In addition, to the success of the field, formalization of the whole paradigm of content-based image retrieval to bring it to a level of consistency and integrity available is essential. Without this formalism it will be hard to develop sufficient reliable and mission critical applications that are easy to program and evaluate.

In conclusion to this review of image similarity techniques, several main problems remain to be addressed for these techniques to be easily applicable to the full-range of access problems to large image databases:

- Study of the distribution of measures for various feature spaces on large real-word sets of image. In particular, how well is the perceptive similarity order preserved by the measure when the number of images grows?

- Study of ranking visual items that corresponding to human perception.

- Definition of methods for the segmentation of images in homogeneous regions for various feature spaces, and definition of models of this spatial organization which could be robustly combined with the similarity of the local features.

- Detection of salient features to a type of images or objects, so that to free user from specifying a particular set of features in query process.

- Combination of multiple visual features in image query and search.

- Developing efficient indexing schemes based on image similarity features for managing large databases.

3.0 Content-Based Retrieval of Video (CBVR)

3.1 Introduction

Content-based Video Retrieval (CBVR) systems appear like a natural extension (or merge) of Content-based Image Retrieval (CBIR) systems. However, there are a number of factors that are ignored when dealing with images which should be dealt with when using videos. These factors are primarily related to the temporal information available from a video document.

Temporal Information:

The temporal information firstly induces the concept of motion for the objects present in the document. When in CBIRS, it is the list or organization of such object which is search for, video retrieval may imply the retrieval of a behavior of an object throughout the document. Two video documents may therefore contain the same objects but little relevance may be found between the two in this search context. It is therefore essential to encode within the indexing of a video document the behavior of all objects throughout the document.

Structural Information:

Another aspect that does not exist in CBIRS and that should be taken into account in CBVRS is the structural organization of the document. A video document can typically be split into a hierarchical structure. Another issue in video retrieval is the complexity of the querying systems. A very elaborated retrieval system would allow flexibility for the user to specify its query parameters.

Need for CBVRS:

By definition, a CBVRS aims at assisting a human operator (user) to retrieve a video sequence (target) within a potentially large database. Three major cases may be distinguished.

- The user has a specific sequence in mind and knows it is included within the database. In this case, the target is unique and corresponds to specific criteria.
- The user has a specific video in mind and does not know if the document he is looking for exists in the database.
- The user simply searches for a document by referring to its topic or some event occurring within it.

3.2 Spatial Scene Analysis

Visual document processing operations are essential for automatically extracting an extensive description of a document. Feature extraction aims at characterizing a list of properties (called *feature vector* or *document signature*) for each component (pixel, frame region, frame, sequence) of a video document. The analysis of elements such as color and texture aim at characterizing features in the spatial space (as opposed to the temporal domain). Experience acquired from CBIR studies may be fully transferred to video in this case.

Color Feature Space:

Color is an important cue for measuring the similarity between visual documents. Color statistics are used for measuring global or local dissimilarities. Color features are analyzed through histograms. These histograms offer the advantage of being invariant under rotation, translation and many other geometric operations. Features encoding color organization within the document are often based on blocks. A feature vector is attached to each unit part of the spatial domain, and it is the relationship between neighboring image parts which is encoded as feature.

Texture Feature Space:

The analysis of textures requires the definition for a local neighbourhood corresponding to the basic texture pattern. It makes no sense to study the texture of an isolated pixel. Typically, the analysis is done via the mapping of the texture onto the response of one or a bank of pre-defined filters against the image (wavelets, Gabor filters). Another approach defines textons as the basic builders for any texture. Each texture is decomposed using these building blocks and the parameters of the local texture are obtained. Typical texture features include orientation and coarseness. Texture models are learned so that geodesic active contours are able to segment texture regions (then considered as uniform or consistent patches), thus extending the classic snake-based segmentation algorithms. Supervised learning of textures is done via texture samples.

Supervised Feature Space:

More complex features may be defined for parsing the contents of a video document. One example of such feature is the development of face detection algorithms. Another example of such feature is the retrieval of text in a video document. It is often the case that textual annotations are readily available within the document itself.

3.3 Temporal Analysis

The temporal dimension of a video document contains an information that is specific to this type of document. The temporal analysis of that document typically requires its partitioning into basic elements. It is now recognized that this partitioning can operate at four different levels of granularity.

Frame level: Each frame is treated separately. There is no (or little) temporal analysis at this level.

Shot-level: A *shot* is a set of contiguous frames all acquired through a continuous camera recording. The partitioning of the video into shots generally does not refer to any semantic analysis. Only the temporal information is used.

Scene-level: A *scene* is a set of contiguous shots having a common semantic significance.

Video-level: The complete video object is treated as a whole.

Three Types of Shot-level:

Cut: A sharp boundary between shots. This generally implies a peak in the difference between color or motion histograms corresponding to the two frames surrounding the cut. Cut detection may therefore simply consist in detecting such peaks. Adding any form of temporal smoothing will also improve the robustness of the detection process.

Dissolve: The content of last images of the first shots is continuously mixed with that of the first images of the second shot. The major issue here is to distinguish between dissolve effects and changes induced by global motion. Fade-in and fade-out effects are special cases of dissolve transitions where the first or the second scene, respectively is a dark frame.

Wipe: The images of the second shot continuously cover or push out of the display (coming from a given direction) that of the first shot.

The definition of a scene is based on a deep understanding of the contents of the shots. Automated scene annotation rely on a high-level clustering of shots where the indexing data derived from shots composes feature vectors (see below). Depending on the video, the segmentation of shots may lead to a small, manageable set of objects (shot representations).

Motion Feature Space:

While color and texture and their organization characterize the content of a still document, when processing video documents, it is essential to also account for the temporal dimension. The temporal features should provide the information regarding the global (temporal) organization of a video document. Temporal information is generally translated into a motion characteristic. Motion analysis is made on matching consecutive frames one with another. A (possibly directed) search is made between pixel blocks of two consecutive frames. Statistics allow for characterizing global motion (dominant or camera motion) and object motion. Using this information, one can compensate for the global motion leaving only object motion so that temporal information can eventually be used for characterizing (*e.g.*, isolating) objects within the document.

Audio Feature Space:

The audio stream attached to a video document may be of great help in understanding the document. Typically, audio processing techniques are based on the analysis of the energy contained in the audio signal. The signal is divided into *audio frames*, corresponding to few milliseconds of the signal. Features such as Mel-frequency cepstral coefficients (MFCC) and associated statistics are then derived for characterizing and classifying the audio frames. One task consists in distinguishing between speech and music or background noise in the audio signal.

Indexing:

Once the video segmentation is operated at a desired level, the indexing of the document is performed by creating some meta-data which will be attached to this document for quick reference. The content of the meta-data varies, depending on the application towards which the database is oriented. For generic video documents, this data generally includes video object (shot, scene) boundaries along with some characteristic and visual representation. One common representation is the choice of one or more key frames within the shot or the sequence. Depending the assumptions under which the segmentation has been performed, all frames within a basic shot should normally be consistent with each one another. Heuristics for choosing one or more key-frames can therefore be derived. The simplest relies on the global position of the frames within the shot (first, middle, end). Some other characteristics such as the corresponding audio stream may also be used for efficient key frame detection.. The process of re-segmenting the shots with respect to some heuristics reflecting a comprehension of the video content is referred to as video *micro*-segmentation.

3.4 Video Query

The problem of searching a video document calls for that of formulating a meaningful and clear query. For content-based image retrieval, the system of query-by-example is relatively intuitive since it caters for cases which would be difficult to solve using simple text queries. For video documents however, things are no so straightforward since a query-by-example would require the user to have a video at hand already. One of the major problem is the excessive dimensionality of the search space induced by the temporal information. In order to reduce this dimensionality, different approaches are taken which all introduce advantages and shortcomings.

Visual Query:

Content-based retrieval systems address the search for visual objects. Query-by-example (QBE)-based CBIRS can be divided into different groups. In the most basic category, the user is asked to choose one image which is supposed to resemble the one he is searching for. As a result, the system returns in decreasing order the images it finds the most similar to the example given. An enhancement of such a system allows for the user to choose more than one example so that the query combines all common features in these images. Another refinement of this principle consists in using only parts of the images for the query. This requires the definition of a perceptually-meaningful segmentation technique through which images in the database are pre-processed. Less examples of QBE-based video search tools exist. One major reason may be the difficulty in describing a video document in a simple and easy-to-represent way. Following earlier analysis, a video example may include either visual still information (*e.g.*, key-frames) or motion information.

Motion Query:

Motion is essentially the only way of representing the temporal information contained in a video document. Motion-based query is therefore an attractive feature of a video search engine. The problem is the formulation of such a query. Motion-based queries can be seen as counter-intuitive in the sense that the user is asked to represent a motion in some still fashion. It seems clear that solving the problem of the formulation of a motion-query is to be made in parallel with that of representing the motion information in the indexing task. Such an approach will facilitate the comparison of the user demand with the available information.

Textual Query:

Textual query is very important since it offers to the user the possibility to complete the weaknesses of the querying interface. QBE-based systems have demonstrated their superior descriptive power, when compared to text-only querying systems. However, it turns out that both querying systems are needed. In other words, textual query seems to be a necessary complement to a QBE-CBIRS. The reason for this is that textual query allows the user to insert some ``personalized" data within the query. Using keywords, the user may express high level concepts which would be difficult to express through QBE. For images, all is working as if the query was formulated like: ``Retrieve a document which contains [keywords] like these [example document]". For video documents, textual querying may be of even more comparative importance since it allows for completing (or replacing) the expression of the motion component of the query.

3.5 Future Work

The majority of the techniques reviewed here address problems in recovering low level structure of video sequences, though these techniques are basic and very useful in facilitating intelligent video browsing and search.

In conclusion to this review of content-based retrieval of video techniques, several main problems remain to be addressed for these techniques to be easily applicable to the full-range of access problems to large video databases and future work might include:

- Combining query types. The necessity of using a combined query system appears clearly from the analysis. Querying systems should typically be organized so as to cater at maximum for all possible users' needs.

- Automatic transcription generation. It removes the need for subtitles or other manually-generated annotation, meaning nearly any spoken data may be retrieved by its content.

- Developing video analysis tools. Video content analysis, retrieval and management should not be thought of a fully automatic process. We should focus in developing video analysis tools to facilitate human analysts, editors and end users to manage video more intelligently and efficiently.

4.0 Content-Based Retrieval of Audio (CBRA)

4.1 Introduction

Audio content analysis, classification, and retrieval have a wide range of applications in the entertainment industry, audio archive management, commercial musical usage, surveillance, etc. While the use of keywords for sound browsing and retrieving provides a possible solution, it is however time- and labor-consuming in indexing. Moreover, an objective and consistent description of these sounds is lacking, since features of sounds are very difficult to describe. Consequently, content-based audio retrieval would be the ideal approach for sound indexing and searching. Furthermore, content analysis of audio is useful in audio-assisted video analysis. Possible applications include video scene classification, automatic segmentation and indexing of raw audiovisual recordings, and audiovisual database browsing.

4.2 Classification

Hierarchical system is being developed for audio content analysis and classification. With such a system, audio data can be archived appropriately for the ease of retrieval at the query state.

- **Coarse-level classification.** The first stage, audio signals are classified into basic types, including speech, music, several types of environmental sounds and silence. For this level, relatively simple features such as the energy function, the average zero-crossing rate, and the fundamental frequency to ensure the feasibility of real-time processing. A rule-based heuristic procedure is built to classify audio signals based on these features. An on-line segmentation and indexing of audio/video recordings is achieved based on the coarse-level classification.

- **File-level classification.** The second stage, further classification is conducted. For speech, we can differentiate it into voices of man, woman, child as well as speech with a music background. For music, we classify it according to playing instruments and types (for example, classics, blues, jazz, rock and roll, music with singing and the plain song). For environmental sounds, we divide them into finer classes such as applause, bell ring, footstep, windstorm, laughter, bird’s cry, and so on. Based on this result, a finer segmentation and indexing result of audio material can be achieved.

4.3 Retrieval

In the third stage, an audio retrieval system is built based on the archiving scheme in previous two steps.

Two Retrieval Approaches:

- One is **query-by-example**, where the input is an example sound, and the output is a rank list of sounds in the database which shows the similarity of retrieval sounds to the input query. Audio clips may also be retrieved with distinct features such as timbre, pitch, and rhythm. The user may choose one feature or a combination of features with respect to the sample audio clip.

- The other one is **query-by-keywords (or features),** where various aspects of audio features are defined in a special keyword list. The keywords include both conceptual definitions (such as violin, applause, or cough) and perceptual descriptions (such as fastness, brightness, and pitch) of sounds.

4.4 Process

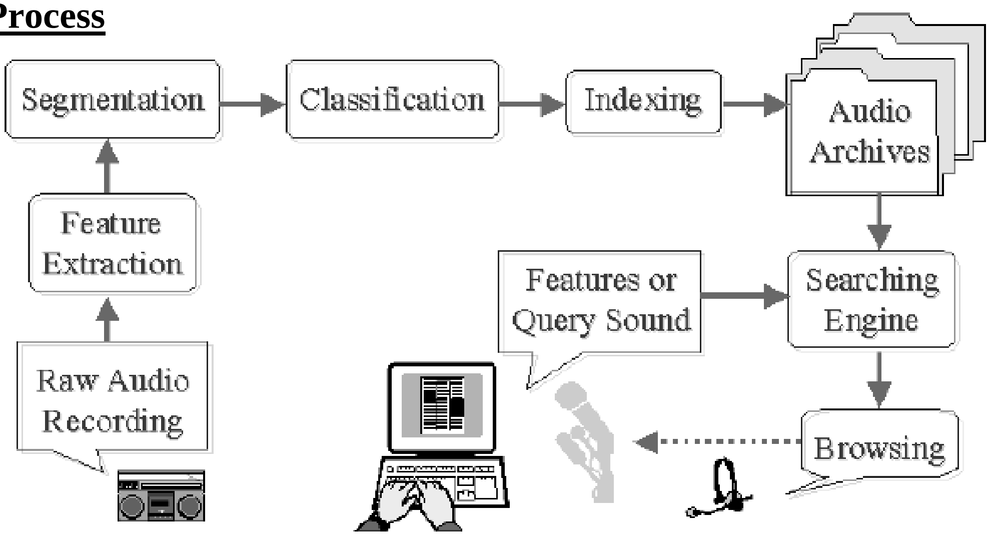


Figure 4.4 Content-Based Audio Classification and Retrieval

The procedure of the audio classfication and retrieval approach in an audio achive management system: Raw audio recordings are analyzed and segmented based on abrupt changes of features. Then, audio segments are classfied and indexed. They are stored in corresponding archives. The audio archives are organized in a hierarchical way for the ease of the storage and retrieval of audio clips. When a user wants to browse the audio samples in the archives, he may put a set of features or a query sound into the computer. The search engine will then find the best matched sounds and present them to the user. The user may also refines the query to get more audio material relevant to his interest.

5.0 Content-Based Retrieval Multimedia Systems

In this chapter, a few commercial and/or non-commercial content-based retrieval multimedia systems (digital library/ digital database) will be presented.

5.1 Oracle interMedia

Oracle *inter*Media is a feature that enables Oracle9*i* to store, manage, and retrieve geographic location information, images, audio, video, or other heterogeneous media data in an integrated fashion with other enterprise information. Oracle *inter*Media extends Oracle9*i* reliability, availability, and data management to multimedia content in Internet, electronic commerce, and media-rich applications as well as online Internet-based geo-coding services for locator applications.

Audio Concepts:

Audio may be produced by an audio recorder, an audio source such as a microphone, digitized audio, other specialized audio recording devices, or even by program algorithms. Audio recording devices take an analog or continuous signal, such as the sound picked up by a microphone or sound recorded on magnetic media, and convert it into digital values with specific audio characteristics such as format, encoding type, number of channels, sampling rate, sample size, compression type, and audio duration.

Image Concepts:

ORDImage supports two-dimensional, static, digitized raster images stored as binary representations of real-world objects or scenes. Images may be produced by a document or photograph scanner, a video source such as a camera or VCR connected to a video digitizer or frame grabber, other specialized image capture devices, or even by program algorithms. Content-based retrieval of images with extensible indexing is supported for image matching.

Video Concepts:

Video may be produced by a video recorder, a video camera, digitized animation video, other specialized video recording devices, or even by program algorithms. Some video recording devices take an analog or continuous signal, such as the video picked up by a video camera or video recorded on magnetic media, and convert it into digital values with specific video characteristics such as format, encoding type, frame rate, frame size (width and height), frame resolution, video length, compression type, number of colors, and bit rate.

Matching:

When you match images, you assign an importance measure, or weight, to each of the visual attributes, and *inter*Media calculates a similarity measure for each visual attribute.

Each **weight** value reflects how sensitive the matching process for a given attribute should be to the degree of similarity or dissimilarity between two images. For example, if you want color to be completely ignored in matching, assign a weight of 0.0 to color; in this case, any similarity or difference between the color of the two images is totally irrelevant in matching.

The similarity measure for each visual attribute is calculated as the **score** or **distance** between the two images with respect to that attribute. The score can range from 0.00 (no difference) to 100.0 (maximum possible difference). Thus, the more similar two images are with respect to a visual attribute, the *smaller* the score will be for that attribute.

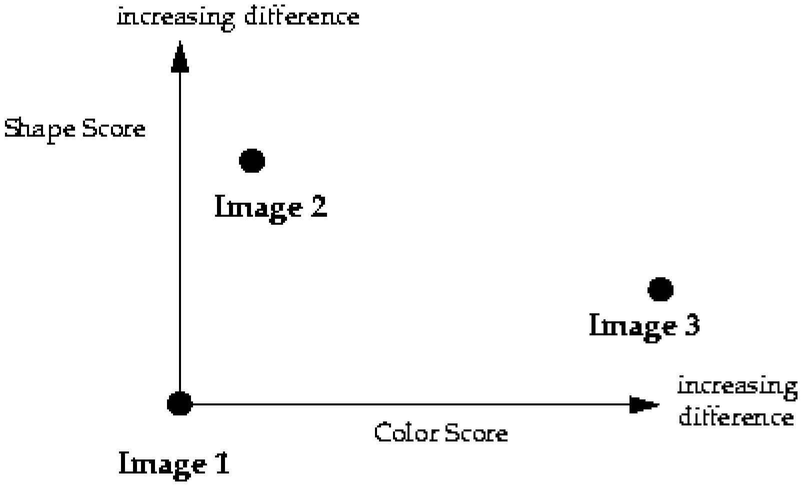
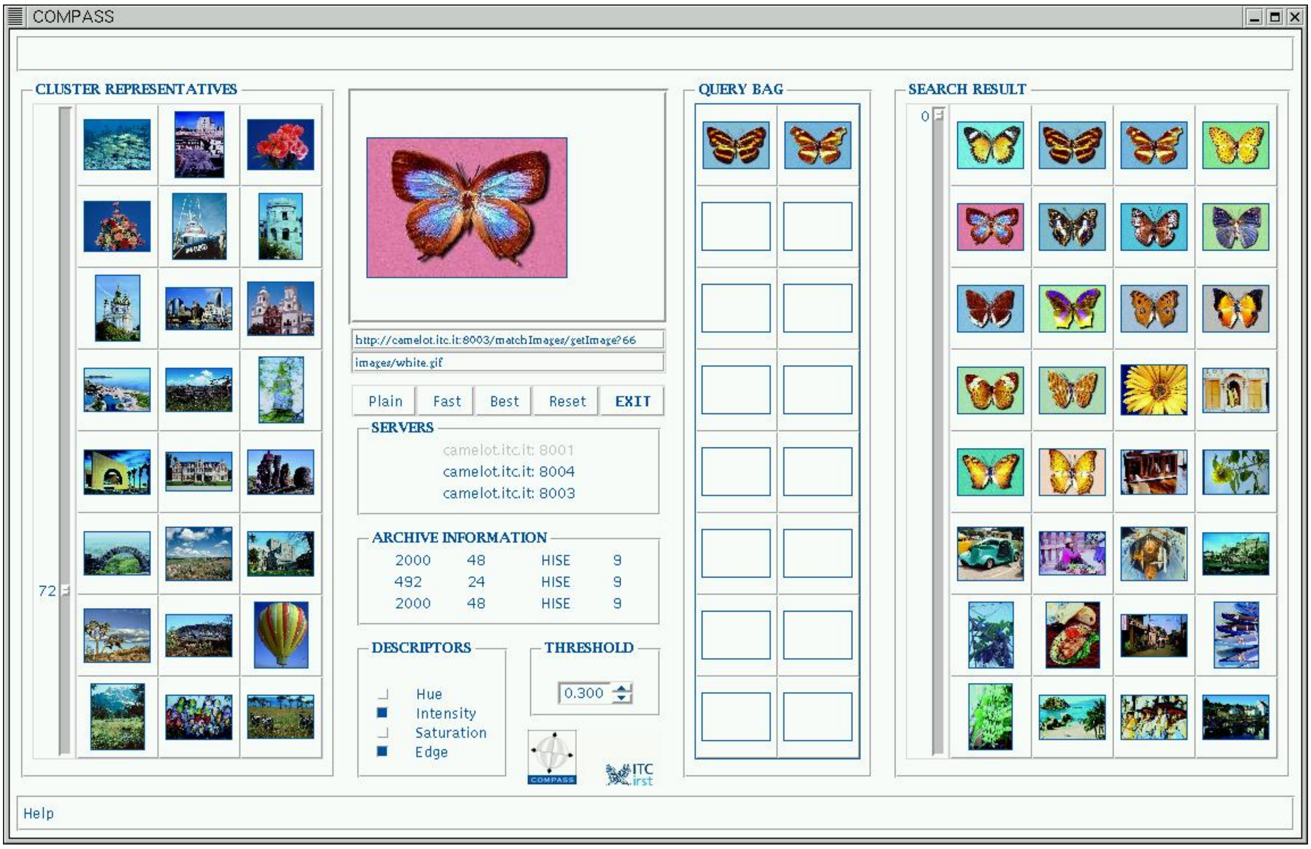


Figure 5.1 interMedia Score and Distance Relationship

For matching, assume Image 1 is the comparison image, and Image 2 and Image 3 are each being compared with Image 1. With respect to the color attribute plotted on the x-axis, the distance between Image 1 and Image 2 is relatively small (for example, 15), whereas the distance between Image 1 and Image 3 is much greater (for example, 75). If the color attribute is given more weight, then the fact that the two distance values differ by a great deal will probably be very important in determining whether or not Image 2 and Image 3 match Image 1. However, if color is minimized and the shape attribute is emphasized instead, then Image 3 will match Image 1 better than Image 2 matches Image 1.

5.2 COMPASS

COMPASS is a distributed application for content-based image retrieval using remote databases. The COMPASS system can be used for two main activities: to browse still image databases, and to search image databases similar to a query image. Therefore, the user formulates queries by examples and not a mere caption textual search



http://camelot.itc.it:8003/matchImages/getImage?66

images/white.gif

SERVERS

camelot.itc.it: 8001
camelot.itc.it: 8004
camelot.itc.it: 8003

ARCHIVE INFORMATION

2000	48	HISE	9
492	24	HISE	9
2000	48	HISE	9

DESCRIPTORS

□ Hue
■ Intensity
□ Saturation
■ Edge

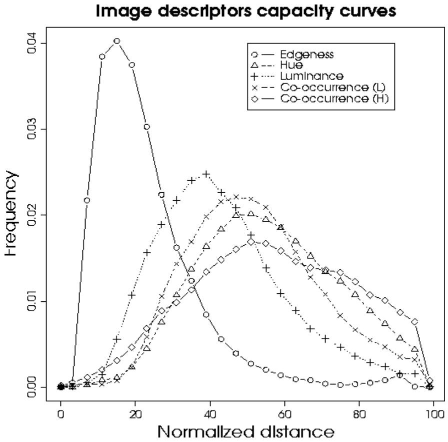
THRESHOLD

0.300

Figure 5.2 COMPASS (http://compass.itc.it/)

Concepts:

Histogram capacity curves provide a basis on which the effectiveness, of different image descriptors can be compared.



If the average value of dissimilarities is low, the curve is very sharp and the histograms are not spread out enough in the histogram space and histogram indexing is not effective. Therefore that descriptor is not effective in making the query.

6.0 Proposal

Content-Based Retrieval (CBR) in multimedia systems and query-by-example (QBE) technology are now attracting to all kinds of stake-holders including marketing people, publishers, journalists and so on. While this paper has looked the most common ways of doing CBR and a few CBR systems. There are definitely a lot of work and research will have to be done in order to produce a successful commercial CBR for people to use. I here propose 2 very preliminary ideas on CBR by QBE, my proposals havn't been proved or tested anyhow, they are just ideas from searches I have done in this area.

6.1 Idea 1: SVG/XAML text-based search

SVG stands for **S**calable **V**ector **G**raphics. SVG is used to define vector-based graphics for the Web. SVG defines the graphics in XML format. SVG graphics do NOT lose any quality if they are zoomed or resized. Every element and every attribute in SVG files can be animated. SVG is a World Wide Web Consortium (W3C) recommendation. SVG integrates with other W3C standards such as the DOM and XSL.

XAML is the user interface markup language for the Windows Presentation Foundation, which is one of the "pillars" of the WinFX API. XAML is a declarative XML-based language optimized for describing graphically rich visual user interfaces, such as those created by Macromedia Flash. XAML focused on 3D graphics, maximize flexibility in relation to 3D features and resource management.

Since, we are so expert on text-based exact match.SVG and XMAL are two perfect formats for saving images in XML format, which idealy, if we have the query-by-example image nicely converted from raster to vector, objects, color, texture, shape, edges of the image can be traced as XML-based descriptor. When image is queried as an example to the database, they will be actually doing text-based search, although won't be exact match, errors should be allowed for small discrepancy.

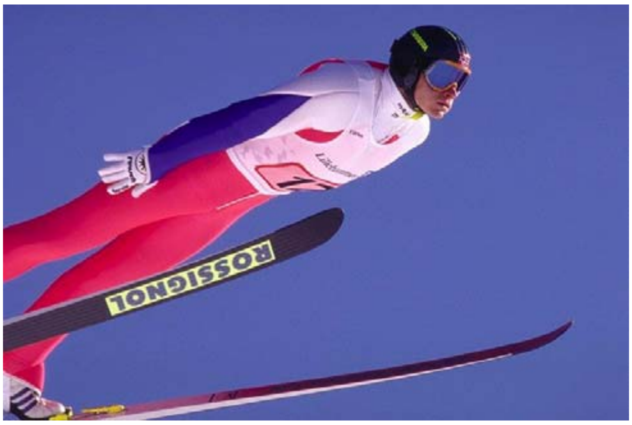


Figure 6.1a Raster Image

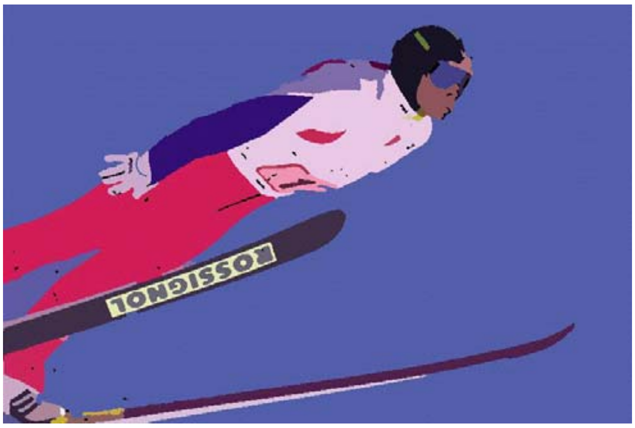
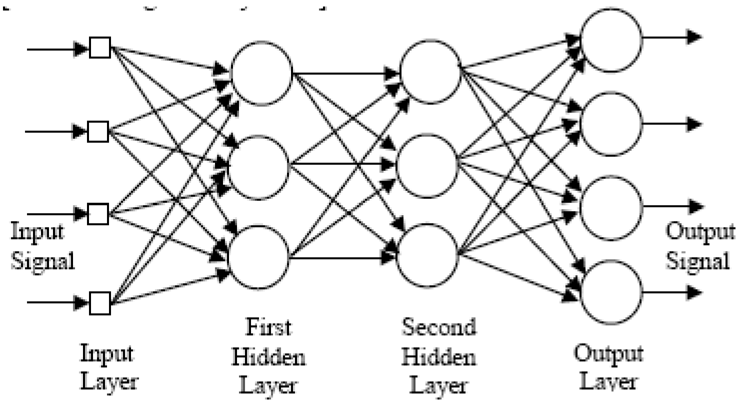


Figure 6.1b Vector Image

6.2 Idea 2: Neural Networks Approach

Artificial neural network is biologically inspired model, based on the functions and structure of biological neurons. A neural network consists of numerous computational elements (neurons or nodes), highly interconnected to each other. A weight is associated to every connection. Normally nodes are arranged into layers. A multilayer perceptron is a feedforward neural network with one or more hidden layers. Typically, the network consists of an input layer of source neurons, at least one hidden layer of computational neurons. During a training procedure input vectors are presented to the input layer with or without specifying the desired output. According to these differences neural networks can be classified as supervised or unsupervised (self-organizing). Networks can also be classified according to the input values (binary or continuous). The learning procedure contains three main steps: the presentation of the input sample, the calculation of the output and the modification of the weights by specified training rules. These steps are repeated several times, until the network is trained.



Now here is the problem and idea.

The **problem** with content-based retrieval is that no formalism is defined. Two different people when querying on a same image in the digital library may have different perception of understanding the image hence the result of matched images will differ based on what they are looking for. One people might be looking for similar color of the images, the other one might be looking for similar object of the images.

So the **idea** is to use artificial neural networks to train customized content-retrieval system. As different people has different perceptions and preferences, training image database with neural networks will result with a closet preference matched samples.

References

1. M. Flickner, et al, Query by Image and Video Content, *IEEE Computer*, September 1995, pp.23-32.

2. K. Hirata and T. Kato, Query by Visual Example: Content-Based Image Retrieval, *Proc. E.D.B.T.'92 Conf. on Advances in Database Technology*, In Pirotte and Delobel and Gottlob eidtors, Springer-Verlag, Lecture Notes in Computer Science, Vol.580, 1994, pp.56-71.

3. M. Stricker and M. Orengo, Similarity of Color Images, *Proc. Storage and Retrieval for Image and Video Databases III*, 1995, February, SPIE Conference Proceedings, Vol.2420, San Jose, CA, USA, February, 1995, pp.381--392.

4. R. Picard and Fang Liu, A New World Ordering For Image Similarity, *Proc. Int. Conf. on Acoustic Signals and Signal Processin*g, Adelaide, Australia, Vol.5, March, 1994, pp.129.

5. R. W. Picard and T.O. Minka T, Vision Texture for Annotation, *Multimedia Systems*, ACM-SPringer, Vol.3, No.3, February, 1995, pp3-14.

6. V. N. Gudivada and Vijay V. Raghavan, Design and Evaluation of Algorithms for Image Retrieval by Spatial Similarity, *ACM Transactions on Information Systems, 1995*, April, Vol.13, No. 2, pp.115-144.

7. K. I. Chang, K. Bowyer and M. Sivagurunath.
   Evaluation of texture segmentation algorithms.
   In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR99)*, volume 1, pages 294-299, Fort Collins, Colorado, 1999.

8. Jitendra Malik, Serge Belongie, Jianbo Shi, and Thomas Leung.
   Textons, contours and regions: Cue combination in image segmentation.
   In *Int. Conf. Computer Vision*, Corfu, Greece, 1999.

9. F. Idris and S. Panchanathan.
   Review of image and video indexing techniques.
   Journal of Visual Communication and Image Representation, 8:146-166, 1997.

10. Content-Based Classification and Retrieval of Audio (1998)
    Tong Zhang, C.-C. Jay Kuo

11. Oracle *inter*Media User's Guide and Reference Release 9.0.1
    Part Number A88786-01