Text-Image Interaction for Image Retrieval and Semi-Automatic Indexing

Gérald Duffing

▶ **To cite this version:**

Gérald Duffing. Text-Image Interaction for Image Retrieval and Semi-Automatic Indexing. IRSG'98, 1998, Autrans, Isere, France, 17 p. inria-00107517

HAL Id: inria-00107517
https://inria.hal.science/inria-00107517

Submitted on 19 Oct 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Text-Image Interaction for Image Retrieval and Semi-Automatic Indexing

Gérald Duffing
UMR Loria
BP 239 – F-54506 Vandoeuvre-les-Nancy cedex, France.
e-mail: duffing@loria.fr

April 22, 1998

Abstract

This paper addresses the issue of retrieving images based on visual content, according a particular attention to the semantic dimension of information retrieval. A brief review of existing Image Retrieval Systems is provided, hilighting a major drawback of these prototypes, namely the lack of integration between classical "semantic search", and visual similarity retrieval (i.e. content-based retrieval). A new approach is proposed, that tries to integrate a real "semantic" dimension into visual content-based image retrieval. Images may be retrieved either by keyword matching (for manually indexed images), or by visual content matching. This approach is then particularly adapted to partially indexed databases, and may be used in a semi-automatic indexing tool.

1 Introduction

Many images can be found in several different image databases. The organization of these databases determines the way users will be able to retrieve images. Some systems allow querying by keywords, assuming that each image has been manually indexed when introduced into the database. A set of keywords is associated to the image, depicting as precisely and exhaustively as possible the semantic and/or visual content of images. This tedious task is crucial: the performance of the system depends on the quality of the indexing process. Another type of system analyses automatically each image and computes off-line a set of visual features, based on the image signal. A similarity measure, based on these indices, is used to search into the database. This "similarity retrieval" can answer questions like "find images that look like this one". No manual indexing process is needed. The former approach is time-consuming, but allows precise semantic search, whereas the latter needs no particular manual image processing, while allowing only retrieval based on *visual* similarity.

This paper addresses the issue of retrieving images based on visual content, according a particular attention to the semantic dimension of information retrieval. We will examine how this issue has been tackled so far in existing image retrieval systems. Then we present our approach, based on an image-text integration. Our model is to be applied to image retrieval and semi-automatic indexing.

2 Information Retrieval Systems: an overview of existing approaches

More and more content-based image retrieval systems are developed [AZP96]. They tackle the difficult issue of retrieving non-textual documents in large databases. Images can be retrieved either by keywords, assuming that each image has been manually indexed, or by a content analysis, based on the image signal (e.g. RGB values of each pixel), that allows visual similarity retrieval.

A visual content-based retrieval system presents some interesting features:

- *It allows retrieval of images that are not indexed by keywords.* The database can be fed with non-indexed images, but, of course, computations must be performed on images, in order to derive numerical features used for matching.

- *Query by example* is possible: users provide the system with an image or a sketch, provided they actually know what they want! Visual conditions can therefore be expressed in the query.

As users' needs are different and may be complex, queries can consist in various conditions expressed at different levels. For *image attributes* and content (like author, title, description), keywords are used as indexing and querying items. For *image composition*, spatial relationships are used to describe the layout of the image, in terms of represented objects. Note that an "abstract" layout can be computed, based on some homogeneity criterion: identifying objects is then not needed. Finally, *visual aspects* can be considered: color, texture and shape information are possible features, but many other criteria could be considered. We now give an overview of image analysis techniques that can be applied either for raw features extraction, or for content analysis. We then briefly describe three different types of systems.

2.1 Image analysis techniques

Basically, any color image is an array of pixels, having a color and a position. Many features describing colour, texture or shape can be computed from this raw data [Pra91, HS92]. The main problem is to find features that are robust (i.e. they produce similar results when applied to similar images) and that are perceptively meaningful. Invariant features address the robustness issue. Those features tend to be independent to scale, rotation, translation and intensity variations in images [RW95, SM95].

It is however often time-consuming to compute many features on each pixel of the image. Features are then computed on homogeneous region [ALO95], or on some pixels, that are of particular interest (*high-curvature points* [RS95], or *key points* [SM95]).

2.1.1 Texture analysis

Texture characterization techniques are applied to gray-scale images, and fall into two main classes of methods: statistical methods, mainly based on co-occurrence matrices, and structural methods (assuming some "regularity" properties of images). We just list here some well-known kind of indices [CP95, VDO85]:

- *Stochastic attributes* can be computed, considering images as a discrete random process. Markov fields-based techniques are a good example [HS80, HY82, PH95].

- *Co-occurrence matrices* [HSD73] assume that information are contained in relations between pixels in a given neighbourhood. Generalised co-occurrence matrices have been defined in [DJA79] and further extended in [WC92].

- In *Spectral approaches*, images are analysed in the frequency domain. The Fourier transform yields coefficients that can be considered as image features.

- *Multi-scale analysis methods* (e.g. wavelet transforms) can cope with all different types of textures, as they can further decompose some bands of the initial signal, in order to capture more precise information [MM96, CK93].

- *Surface attributes and geometric features* can help to texture classification [QNT95], or at least give some shape information [DP96].

- *Mixed approaches* try to make a unique feature encapsulating information on feature and color [CR93].

2.1.2 Color analysis

Color can be represented in different models, or color-spaces. The most common model is RGB, but various other models exist. For example, HSV models correspond to human perception of colors, but are device-dependent... L*a*b* and L*u*v* space are device-independent, and perceptually uniform. They do not correspond, however, to human perception. These remarks illustrate that no single model can be adopted as the best in every domain, as each color-space has its own properties. Moreover, color-spaces are often quantized to reduce the size of the search space, and the effectiveness of color features then depend on the quantization algorithm.

Color histograms have been extensively used to represent the color distribution of an image [SB91]. This is a basic image feature, but more complex approaches can be considered depending on the retrieval context: target search involves, for example, invariants computing, such as color angles [FCF96], or histograms [HS94], that are illumination-invariant.

2.1.3 How to use image features?

Raw image signal yields image features that should produce higher-level information, depending on our needs. Within a content-retrieval domain, image features could be helpful in determining:

- *Homogeneous color region [OKS80, UA94]*, that are automatically computed relying on texture or color homogeneity (i.e. pixel similarity must be determined [SO95, GS95] [MKNM95, HSE+95]). They have no particular meaning to the end-user, as we can't assume that these homogeneous zones actually correspond to a "real world" object;
- *Spatial relationships*, given a description of zones and their absolute or relative position, can form a first idea of the composition of the image;
- *Zones relations*: specific information about zones (inclusion, occlusion, or any higher-level information...);

Apart from these "abstracted" zones mentioned above, user-provided information may be added to cope with more subjective knowledge about objects: "real world" objects are more interesting, though more difficult to identify automatically. Image analysis methods exist, however, to identify known objects [SM95, MP97b], and to automatically annotate images [PM95, SMEK96].

Any data collected about images must be stored within a specific model. Many different models have been designed, some being domain-dependant: EMIR [MHLF93], EMIR-2 [Mec95], Meghini model [MRT91, Meg95], Kiyoki model [KKH94]; models used in VIMSYS [GWJ91], CORE [WDM+95]. The descriptive power of these models is inversely proportional to their ease of instantiation, and utilization, as a very comprehensive model claims very precise image content information. As a result, there's often a trade-off between precise information availability, and retrieval (and indexing) speed.

2.1.4 Image analysis limits

The first limit of image analysis methods is that they are not always robust, i.e. they can return non-similar results when applied to similar images. This is a strong drawback, suggesting that these methods cannot produce 100 percent reliable information for higher-level processes. Moreover, known objects identification is not always possible, as objects can have multiple representations. Visual features must be considered as "hints", or indices, that can help further processing, based on more sophisticated algorithms. Techniques applied to such problems are numerous [DCL97]. Some systems may use data and/or knowledge embedded in their description model: production rules in [GOC+92], graphs rewriting grammars [HKKZ95], natural language analysis tools [Sri95]. Many different methods could and even should be considered, as it is quite difficult to predict which combination of features is the best one, given a particular problem. This leads some authors [MP97a] to implement a "society of models". Artificial Intelligence techniques contributions are described in [CL97]. General frameworks are defined in [GYA97] and can be instantiated for a given domain.

Image analysis techniques yields non robust features, objects identification is not directly manageable in heterogeneous images : thus, automatic indexing is out of question. This brief overview has hilighted some of the most important problems we have to face when considering to implement an efficient content-based retrieval system. The next section examines how image retrieval is carried out by some of these systems.

2.2 Three classes of systems

This simple taxonomy describes three different types of systems: the first one relies on keywords to achieve indexing and querying, whereas the second one relies on visual features, that allow visual similarity computation. Finally, we describe systems that allow both keyword and visual similarity querying and retrieval.

2.2.1 Text-based systems

Keywords are used as indexing terms that describe the content of the image, and as querying items [Hal89, Sma94]. Some Information Retrieval Systems implement a complex data structure that can be browsed, as in VIMSYS [GWJ91], MMIS [GOC+92], or instantiated to formulate a query, as in MULTOS [RS92]. As natural language is powerful, it seems possible to give a good image description with keywords, especially when a data model is available, as in the systems mentioned above, to structure the information. However, their lack of pure visual similarity facilities has motivated the development of new systems.

2.2.2 Visual content-based systems

This class of systems put the emphasis on images pictorial content. The query may consist in an image, a sketch, a color histogram... IIDS [CYDA88] uses predefined iconic objects that the user can place on the screen, expressing by this way a generic image composition. TradeMark [WLS95], and Art-Museum [Kat92] use user-provided sketches to search the database. The VisualSEEk [SC97] system allows queries based on color, texture, shape, and sketches. These systems rely on visual features matching only. That is, they compute a set of indices on images, and use some similarity measures to compare images.

2.2.3 Hybrid approaches

In these systems, besides some visual indices computations, image theme and content can be expressed via keywords, which can be manually provided or semi-automatically computed by the system. Keywords can also be associated with a combination of visual predicates that define their visual representation (e.g. the keyword "sky" is associated to the blue color (The Chabot system, [OS95]). The QBIC [NBW+93] is well known: its similarity measure relies on keyword, shape, color and texture.

3 A text-image integration scheme

3.1 Motivations

Indexing is often a manual time-consuming task. Moreover, words are subjective, and it is difficult to choose the ultimate word, as describing visual impressions remains a tedious task. Users face the same problem of choosing the keyword that best suits their needs : how to describe, with keywords only, the image they want? Visual similarity-based systems allow users to retrieve "visually similar" images, provided users know what image they want (image-based query) , or provided they are great painters themselves (sketch-based query). As visual indices are numerical data, without direct meaning to users, they cannot directly be used as query items. Regarding the general information retrieval task, "visual similarity retrieval" is not sufficient: similarity is a complex concept and *Visual* similarity is only a part of it. Deeper semantic content is often forgotten. A good relevance feedback mechanism would help,

but it is rarely implemented in these systems (actually, it seems difficult to determine how to modify a "visual" query, based on user feedback).

All these remarks emphasize the importance accorded to keywords in retrieval systems, as the best mediating object between users' desires and image content. We propose an approach that bridges the gap between text- and visual-based queries, establishing a relationship between keywords (representing the *thematic* part of the image) and numerical features (representing the *visual* part of the image).

3.2 Implementing an "image-text interaction"

To take advantage from image analysis results without renouncement of the use of keywords, we want to establish a strong relationship between keywords and visual features. We must keep in mind that keywords are very important, as they are widely used as query items, and therefore as indexing terms. The problem stems from their interpretation: different users may not associate the same visual representation to a given keyword. Moreover, one single keyword can actually have multiple representation in the database. For example, a general keyword such as "sky" may correspond to a blue sky, or a cloudy sky, a red sky, etc... Our approach aims at tracking each possible representation of keywords.

3.2.1 An extended thesaurus

The first task of a thesaurus is to control and structure vocabulary: users can use it to find the most relevant word to use in a query, being sure that this word is known by the system. So a basic idea is to extend this thesaurus in a way it can store additional data, dealing with visual properties. Each keyword can be associated with one or more *realization*, which is a visual example taken from an image clip. Information as color, texture or shape features, but also spatial relationships (topological properties) are computed from this clipped image. For example, a generic description for "sky" establishes that it is often at the top of the image. A more specific "blue sky" has the following additional properties: texture with very low contrast, dominant blue color. The thesaurus becomes a dynamic multimedia structure, as it can be incrementally enriched by new realizations, as a result of the indexing process.

3.2.2 Thesaurus utilization

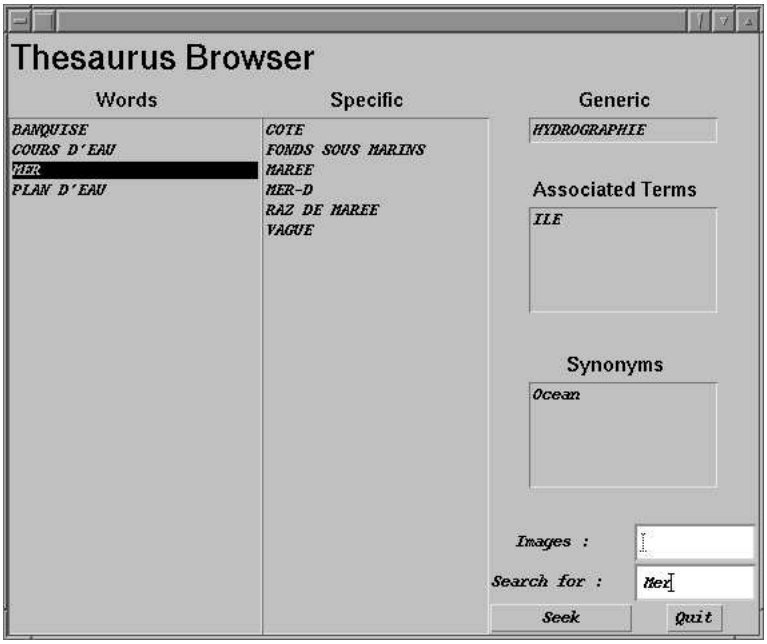

Figure 1: Thesaurus Panel

Our thesaurus provides users with links, that allow browsing through keywords. Classical links are "more general", "more specific", "see also", ... Figure 1 shows our prototype window that allows thesaurus browsing. New links could however be added, such as "same color", "same texture", "representation of", etc... The latter link is important: it gives the ability to link a *concrete word* (i.e. a word that's been assigned visual properties) to a more abstract word (i.e. a word that has no direct visual representation, like "trade"). Thus, the system is able to retrieve an image featuring an abstract concept by searching for concrete concepts, that are known to be likely to be representations of the abstract word.

By selecting a keyword in the thesaurus, users can visualize and choose a specific representation of that keyword, so that they can build more precise queries. This "pre-visualization" at the query definition step can be helpful, especially when dealing with very large databases where too many images could be retrieved by an imprecise query.

In query refinement or expansion step, the system can also take advantage of the thesaurus, as other features such as color and texture should be considered to derive a new query.

As this thesaurus implements a relation between a keyword and visual properties, images that are not indexed by keywords can also be retrieved. Suppose that we are searching for images containing "blue sky". From this keyword, and its associated visual features, we derive that images featuring mainly blue pixels in the top match the query, even if there is no indexing record available for these images.

3.3 Taking into account different levels of abstraction

The approach we described above is however not sufficient, as not every keyword may have a visual representation. Abstract keywords, that have a great power of description cannot be reduced to simple "realization", that is simply based on an image clip. However, a set of more concrete words, along with their visual representation, can be, in some circumstances, used to illustrate this abstract concept. To organize the known concepts, and to give the system a certain "knowledge", we designed a specialized data structure, that could store a *concept hierarchy*, as represented in figure 2.

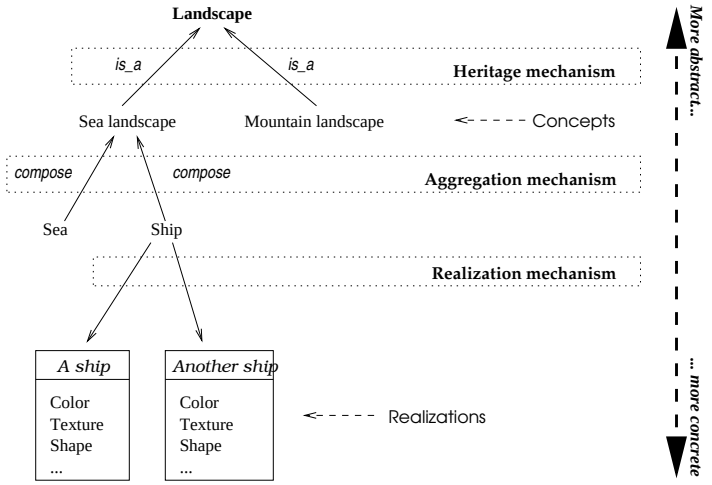

Figure 2: A simple concept hierarchy

In this example, a general and abstract *landscape* has two more specialized versions. One of them has been further described, as being possibly composed of concepts "sea" and "ship". The last concept has two known realizations, with associated visual features that can be actually used for retrieval purposes. Thus, if a user submits a query with "landscape", the system may try to analyze images in order to identify "sea" and/or "ship" in them, using its visual similarity capabilities and the image features stored in "sea" and "ship" realizations.

3.4 Integration in a classical Information Retrieval System

We now describe our approach architecture. Figure 3 sketches its main features. In our approach, we intend to take advantage of the man-machine interaction during the retrieval session, from which we think that valuable information can be extracted and reused as hints or knowledge for indexing purposes. The indexing process takes place at the end of a session (We make the assumption that the corpus is *partially* indexed).

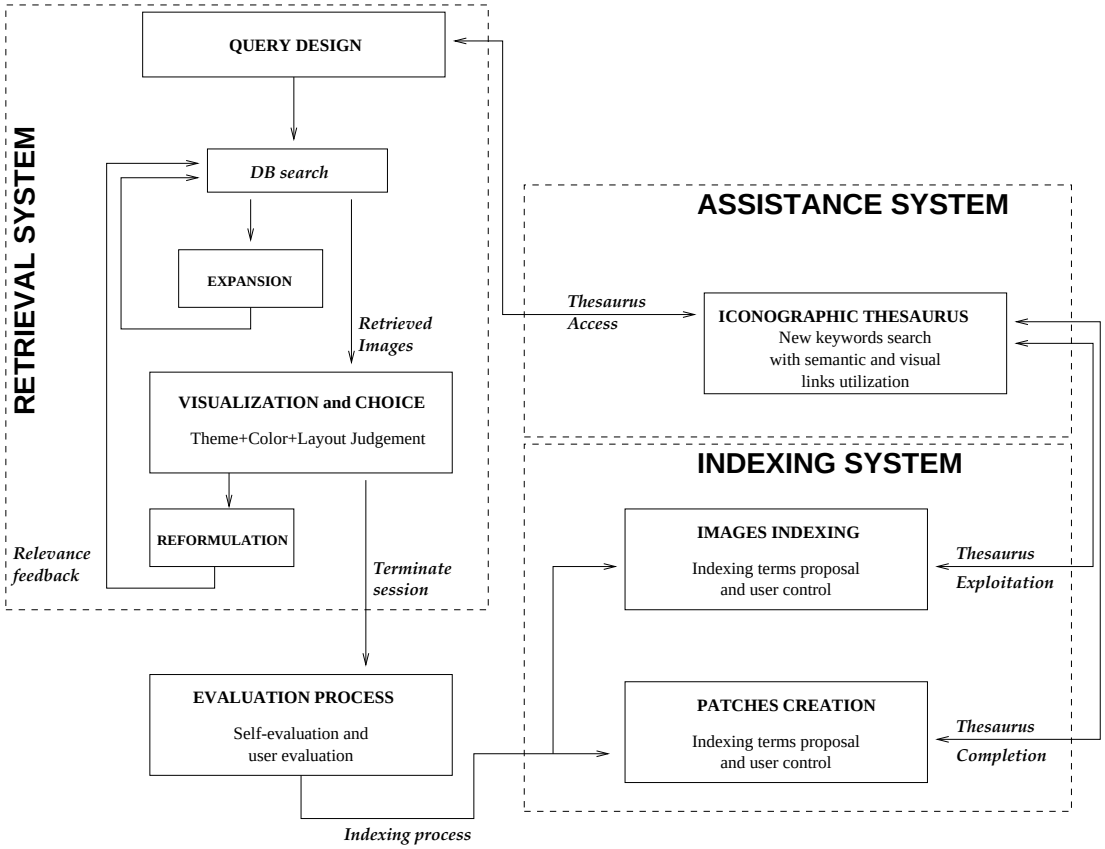

Figure 3: System architecture

3.4.1 Query formulation

Users can convey their query to the system using keywords. The vocabulary is controlled by the thesaurus. A short example of query appears in figure 4. Keywords are selected in a thesaurus, and imported in the query frame. User can formulate for each query item some constraints: the concept depicted by the keyword should be "absolutely", "rather" or "possibly" absent or present in images.

Clicking on the *Thesaurus* button causes the thesaurus frame to synchronize, so that user can get more information on the selected keyword, and related concepts. Clicking on the *Visualize* button raises a frame that displays all possible keyword realizations, so that the user can possibly choose the realization that best suits his needs. The buttons in the lower part of the window allow access to other query tuning facilities: *Conditions* button deals with search strategy tuning (i.e. the contribution of different agents), whereas *Other criteria* button allows user to express additional conditions on image attributes like author, title, etc.

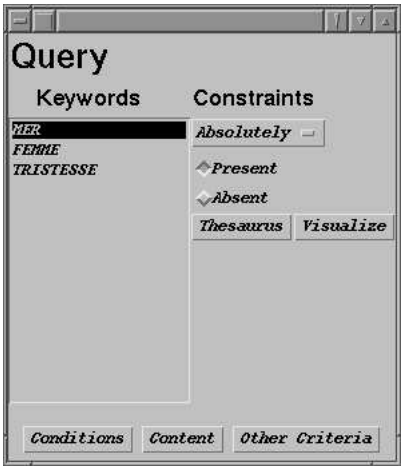

Figure 4: A simple query

3.4.2 Database search

Retrieving images involves both *keywords and visual features* matching. This allows different ways of image database access, so that non-indexed images are not left aside. Depending on the available information for each image (keyword as indexing items, or raw visual features computed off-line), more or less semantic is known about the image, and can be taken into account in the matching process. We will present in this section how this idea can be achieved in our image retrieval system. Figure 5 gives an overview of the global strategy.

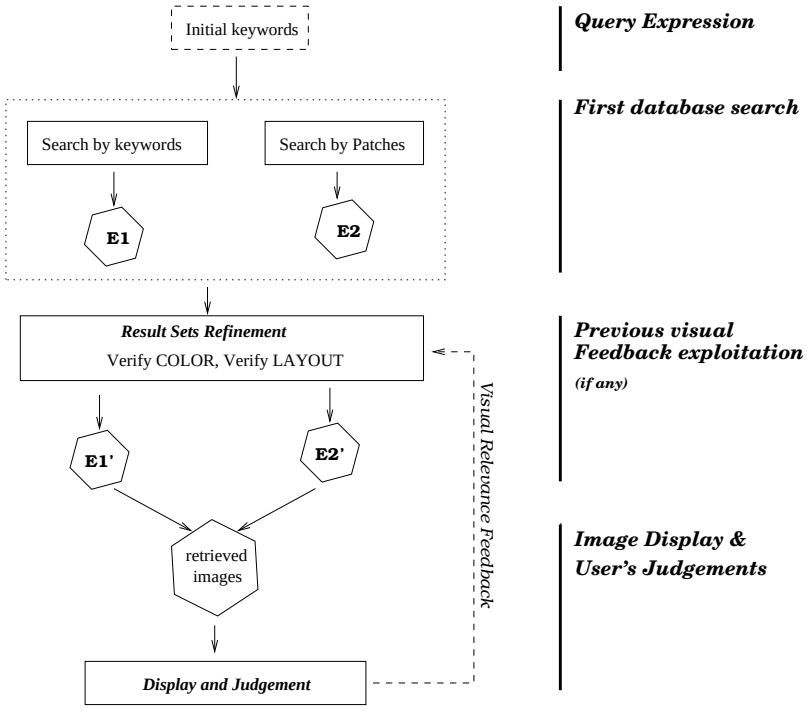

Figure 5: Search strategy

Searching by keywords. – In a first step, the keywords given as query items are used to search the database, within a simple keyword matching process. A first set of images is then extracted (a subset of

the manually indexed images satisfying the "textual" query). This yield to the set labelled E1, in which each image is scored according to its degree of relevance. If no images are retrieved by the search engine, an expansion process tries to modify the query. This step aims at finding new query items (keywords) using one or more modules described in section 3.5.1. The retrieval process is the re-initiated with these new keywords.

Searching by visual features. – Visual features give the system the ability to deal with non-indexed images: they are used to produce a numerical description of some keywords and are stored in the thesaurus. Figure 6 shows some relationships between keywords, image zones, and the image itself, that can be used in the search process. "Annotated zones" are image clips that have been proposed by user as "interesting zones", and related to a known keyword (it becomes a *realization* of that keyword, and this relationship is stored in the thesaurus). The keywords present in the query, and which have *realizations* in the thesaurus can then be used to search the database again, using a search technique based on visual similarity (set E2).

Moreover, a list of zones, along with spatial relationships, yields an image composition. Any feature or combination of features can be used to compute homogeneous zones in an image: this forms an abstract image layout, that can possibly be used as a clue to find similar images, depending on the previous user's judgements: this **user feedback**, used to reformulate the query, is based on a threefold judgement that will be discussed in the next section.

E1 and E2 can be refined if, in previous steps, color has been strongly accepted or rejected for some images: the system can then give more or less importance to the different colors: retrieved images' scores will be updated according to their color distributions. Moreover, if layout has been considered important, then the localisation of these colors is also taken into account, and scores are altered accordingly. This is known on figure 5 as *visual relevance feedback*. In this refinement process, however, no information about the actual semantic of the scene is available.

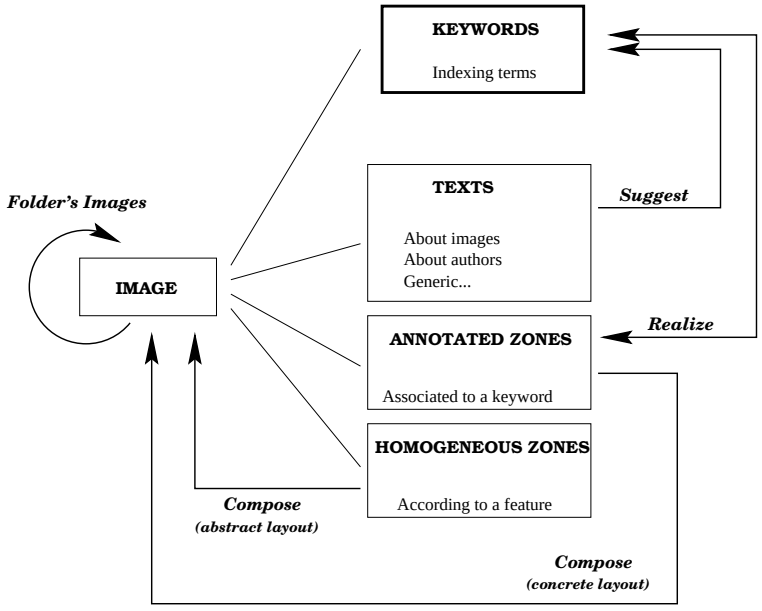

Figure 6: Some cooperation principles

Adaptive level of abstraction. – As users choose keywords in the whole thesaurus, no assumption can be made regarding the degree of abstraction of the chosen word: the more close to the root, the more abstract the keyword should be. Images may be manually indexed by abstract keywords: consequently, using an abstract word in a query will not affect image retrieval, for manually indexed images. *Realizations*, however, tend to be created for very concrete words, for which a visual representation exists in images. And as we mentioned in section 3.3, when an abstract word has been chosen as a querying item, it is highly possible that no realization has been defined for this word. The system then has to expand

the query, and therefore exploit an additional data structure (see figure 2) that "knows" a path from this abstract concept to some possible concrete words that are more likely to have a visual realization. This process might lead, however, to an alteration of the semantic of the query.

When a set of images has been defined, images are presented to the user, ordered according to their relevance score. The system then has to take into account the user's judgement, to adapt its retrieval strategy.

3.4.3 Results browsing and judgement mechanism

If the user is unsatisfied by the system's answer, the query can be refined, taking into account his judgements. The visualization window shown in figure 7 allows users to browse through retrieved images, and assess their relevance.

We believe that judgements provided by user when browsing through retrieved images are a valuable information, as it should discriminate between accepted and rejected images. Thus we allow him to give a threefold judgement on each image: theme, dominant color, and layout. *Theme* is described by keywords that may be used for image indexing. *Dominant color* can be considered at two scales: if the layout is not considered as important by user, then the dominant color addresses the whole image color distribution. On the contrary, if user expresses that the layout of the image is important to him, then we consider the color distribution of an image tile. We may consider, of course, only the most important bins in the color histogram. The *layout* of an image deals with homogeneous zones' spatial relationships. The homogeneity criterion may refer to color or texture, depending on the image. Briefly, user's judgement consists in:

- Judgement on theme: as it is impossible to take into account only the visual properties of the images in the database search, keywords are crucial. This question is the one of "aboutness" of the image, considering users' thematic needs.
- Judgement on color: this is a fundamental visual property of the image. It has a great discriminant power, and we shall only consider dominant colors.
- Judgement on layout: this allows the user to give an opinion on the whole image composition, not considering precisely each component (which are not identified).

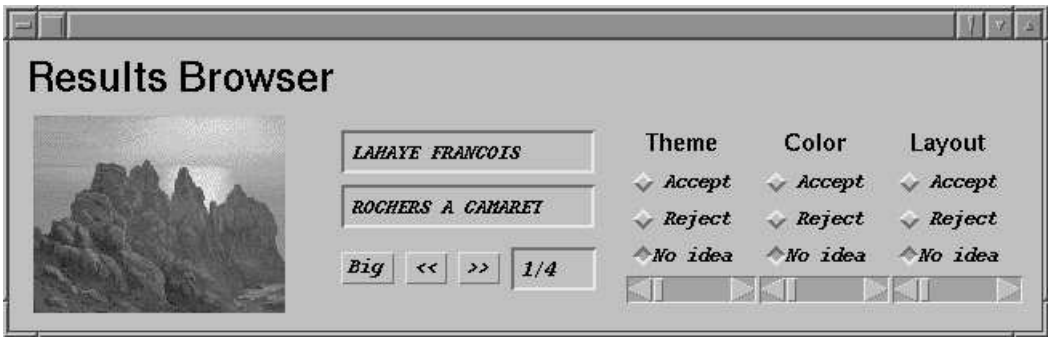

Figure 7: Image display and judgement panel

For each image, and for each aspect described above, the user can accept, reject, or have no idea. If the image is rejected or accepted, a weight ranging from 1 to 10 has to be provided. Each aspect is then analyzed to refine the query. Note that as soon as a first judgement has been proposed by the user, the system can take into account color and layout properties. From all this data, the system will try to find discriminating information on each axis: is there particular colors (theme, layout) that are found in common in accepted (rejected) images? This allows a query refinement, as the weight associated to each "feature" is updated according to user's decisions. For example, images featuring a dominant red color in upper right part of the image will be better scored, assuming that the users strongly accepted "color" and "layout" judgements on one or more images that had actually a red zone in the upper right corner...

3.5 Indexing strategy

The indexing process takes places at the end of a retrieval session. Of course, this is an optional task. In our system, indexing is twofold (see figure 3). The first possible step aims at extracting and proposing new indexing terms for images. The goal of the second step is to provide the thesaurus with new keywords realizations.

3.5.1 Image indexing

As the search engine relies not only on keyword matching but also on visual features matching, the retrieval session might have retrieved images that were not indexed yet. It is therefore possible that keywords present in the query could actually be used as indexing keywords for those non indexed image. Case-based reasoning on queries found in retrieval sessions can be considered [Sma94]. Those session records can be regarded as long term memory.

Moreover, when analyzing an image, we derive a set of visual properties. A first step towards indexing is to perform a reverse search in the thesaurus. Given these visual features, we can extract a set of *possible* keywords, which become indexing terms candidates. Of course, in many situations, this will not be sufficient, as it could return too many keywords. Additional help may be considered, however, either to refine this candidates list, or to find other candidates. For example, we have, as session knowledge, a set of queries associated with retrieved images : we can perform co-occurrence analysis between the retrieved candidates and the keywords used in queries, in order to refine the candidates list. Any other specialized modules and their associated data structures may help. Some of them have been implemented as "agents" [Sim96]. [MD97] describes how agents can be organized to solve a particular task. In our field, one could imagine:

- *a concept expert*: we already mentioned the benefits of a "concept hierarchy" (section 3.3) to help building a path from an abstract concept to concrete words;
- *a full text indexing expert*: it implements full text analysis methods, in order to extract information (i.e. relevant keyword) from any textual data that comes with the image;
- *a co-occurrence expert*: given a keyword k, this module gives a list of keywords that are often associated with k in the database;
- *a corpus expert*: it keeps track of different type of paintings: "portraits", "scenes", "landscapes", and associates general features to them. This kind of information could facilitate the image analysis process;
- *a art history expert*: given a painter name, and the date of the painting, this expert could derive some "general plausible painting characteristics", or a list of painters that have the same painting techniques, etc...
- any other specialized module, whose task is to identify any feature in images;

In a first step, agents try to generate a list of possible new keywords. Agents dealing with text could isolate "interesting" words in it; agents analysing the visual content of an image could extract keywords from the numerical content of images, using the thesaurus, etc... Given this list of keywords, agents try to cooperate in order to give a score to each keyword, depending on the task. Finally, keywords that are most likely to suit the current context are chosen as candidates. Of course, it remains the user's responsibility to determine whether the choice of a keyword as an indexing term is correct or not.

3.5.2 Thesaurus completion

If the user is satisfied with the answer of the system, we want to take advantage of the session. As the session record may contain non-indexed images, what we want to do, is to ask the user to show the system which part of images could be annotated with the keywords used in the query. This is a way to enrich

the thesaurus with new keywords' realizations. The user selects a keyword, and draws a corresponding zone on the image (see figure 8). The system then computes a set of visual features on this image clip, and stores all that information in the thesaurus.

In spite of the fact that this task is mainly a manual one, we could try to extract those "interesting zones" automatically, using zone homogeneity criterion, or computing some "key points". This approach has not been experimented yet.

Figure 8: Realization creation frame

4 Experimentation

4.1 Experiments' goals

Adapting the retrieval strategy. – We are primarily concerned with an Image Retrieval System design. An adapted retrieval strategy has to be outlined, so that text querying and internal visual features use can be exploited simultaneously. We presented above our proposal, and experiments now have to be conducted to verify the feasibility of this approach. The retrieval performance will highly depend on the corpus, so we will try to conduct our experiments on different corpuses.

Selecting image features. – We need to select some image analysis techniques: among several different visual indices, and different corpuses, we have to choose a set of features that:

- *Are close to human perception*, so that users' judgements can be easily taken into account [PM95]. System's similarity shall be close to the user's one;
- *Facilitate the matching process*, to allow fast and accurate comparisons.
- *Are adapted to the corpus.* Different corpuses may demand different features: each corpus has its own strategy dealing with visual indices use;

After this selection task, and given a global retrieval strategy, a bunch of questions arises, that our experiments shall help solving. For example, when and under which conditions shall we exploit visual features ? How visual indices could cooperate to develop a real synergy ?

4.2 First results in image analysis

As a first set of indices, we chose basic texture and color features. We are planning to add some shape features. To characterize images in terms of visual, we divide each image into small 32x32 squares, called "image tiles". Every tile is then analyzed in order to compute the features we describe below.

4.2.1 Texture features

We compute Haralick's features [HSD73], which have the properties mentioned above. The algorithm's complexity depends on the size of the analyzed region, and on the number of different gray-levels it contains. We compute: angular second moment, contrast, correlation, variance, inverse differential moments, averages sum, variances sum, entropies sum, entropy, variance difference, entropy difference, measures of correlation. Each feature is computed at four different orientations. Our experiments have shown that those indices are not sufficient to provide a satisfying image analysis. Thus, we have to find additional features.

We also compute a vector of nine local invariant photometric features. These are a simplified version of indices used in [SM95] [Kv87, Ter94], as they do not consider different scales. This feature vector is computed on key points, determined from the Laplacian of the Gaussian, applying an Hessian thresholding method on each point in a given neighborhood [Tab94]. The set of point we obtain are often corner points. Such features are not helpful for the retrieval process itself, but rather for texture classification.

These two features could however be used for image tile characterization, as the size of a tile is rather "small". Haralick's feature could characterize texture, provided the image tile is homogeneous enough, whereas key points could locate "corner points" for more complex tiles.

4.2.2 Color features

To compute color distribution in the image, we chose the L*u*v* color space, as it is perceptually uniform: in this space, the just noticeable difference (JND) between two colors is defined to be equal to one, according to the Euclidean distance. It's therefore easy to find similar colors. We applied a coarse color quantization. At this point, our goal is to compute the color homogeneity of a region. A simple color histogram is computed for each image tile, and similar tiles are clustered in regions. Our experiments showed that this color distribution, though very rough (we use a fixed grid), is sufficient to produce a very coarse image segmentation that could give a sketch of the general image color layout. Color histogram are also used as a visual feature for realizations, in order to keep a more accurate idea of the colors present in the image clip on which the realization was based.

4.2.3 Shape features

We are planning to implement features described in [ALO95], i.e. compacity, region moments, boarders eccentricity and region convexity. We are facing the problem of shape extraction: the shape we want to characterize should be relevant to the user. Therefore we can't rely on image analysis techniques, that would probably not give a correct answer: we shall ask the user to outline regions of interest. This however complicates the indexing process, and it is not sure that an object can be actually represented by shape characteristics, as the object itself might not be "extractable" from complex scenes.

4.3 Software development

We developed a *Content-based Image Retrieval System prototype*. In the current version, it allows keyword querying, thesaurus browsing, and realizations creation. Users can specify search conditions, and build their queries using a thesaurus that contains about 6000 words of general interest. Queries are submitted to the search engine, that currently handles keyword-based queries. It manages user sessions, query refinements and actual search in the database. It is written in C++ language. Besides query definition, the man-machine interface provides image display and judgement. This has been written in Java language.

An image analysis library that implements the features we described in this paper has been developed. It can be easily extended with new modules for additional features computations. It has been written in C++ language, and it handles color and gray-scales images ("ppm" and "pgm" file formats). This library is designed to build "visual feature files" that store features to be used by the search engine.

We developed an experimentation platform to test our image analysis library's algorithms : it allows

a quick verification of modules performances. It has been written in Java language, just as every man-machine interface of our prototype. Java modules and C++ modules communicate via Unix sockets.

5 Conclusion and perspectives

We have studied various image analysis techniques, and image representation models. We selected a first set of visual features that could be used within a content-based image retrieval system. We think, however, that priority should be given to keywords for querying and indexing. Thus, we have proposed an extended thesaurus that could associate keywords and visual features. In this system, keyword remains the preferred "mediator" between the user and the system, but visual features can be taken into account to allow more precise query definition, or to allow retrieval of non-indexed images.

This close relationship between text and image properties also facilitates the indexing process, that is considered as a part of the whole information retrieval process. We think that knowledge accumulated during the search process is a valuable information to facilitate further indexing.

We need to refine this cooperation scheme, and to find additional visual features to implement in our prototype. This will allow further experimentation. We are also planning to evaluate the robustness of the approach on other corpuses.

References

[ALO95] Y. H. Ang, Z. Li, and S. H. Ong. Image retrieval based on multidimensional feature properties. In Wayne Niblack and R. C. Jain, editors, *Storage and Retrieval for Image and Video Databases*, pages 47–57, San Jose, CA, 1995.

[AZP96] P. Aigrain, H. Zhang, and D. Petkovic. Content-Based Representation and Retrieval of Visual Media : A State-of-the-Art Review. *Multimedia Tools and Applications*, 3:179–202, 1996.

[CK93] T. Chang and C. J. Kuo. Texture analysis and classification with tree-structured wavelet transform. *IEEE Transactions on image processing*, 2(4):429–441, 1993.

[CL97] D. Crevier and R. Lepage. Knowledge-based image understanding systems : a survey. *Computer Vision and Image Understanding*, 67(2):161–185, 1997.

[CP95] J. P. Cocquerez and S. Philipp. *Analyse d'images : filtrage et segmentation*. Masson, Paris, 1995.

[CR93] T. Caelli and D. Reye. On the classification of image regions by colour, texture and shape. *Pattern Recognition*, 26(4):461–469, 1993.

[CYDA88] S.K. Chang, C.W. Yan, C. Dimitroff, and T. Arndt. An intelligent image database system. *IEEE Trans. on Software Engineering*, 14(5):681–688, 1988.

[DCL97] M. De Marsicoi, L. Cinque, and S. Levialdi. Indexing pictorial documents by their content : a survey of current techniques. *Image and VISION COMPUTING*, 15(2):119–141, 1997.

[DJA79] L. S. Davis, S. A. Johns, and J. K. Aggarwal. Texture analysis using generalized co-occurrence matrices. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-1(3):251–259, 1979.

[DP96] A. Del Bimbo and P. Pala. Image indexing using shape based visual features. In *ICPR'96 13th int. IAPR conf. on Pattern Recognition*, Vienna, Austria, August 1996.

[FCF96] G. Finlayson, S. Chatterjee, and B. Funt. Color angular indexing. In Bernard Buxton and Roberto Cipolla, editors, *Computer Vision – ECCV'96*, pages 16–27. Springer, Cambridge UK, 1996. Also in : Lecture Notes in Computer Science, volume 1065.

[GOC+92] C. Goble, M. O'Docherty, P. Crowther, M. ireton, J. Oakley, and C. Xydeas. The manchester multimedia information system. In *Lecture Notes in Computer Science no 580*, pages 39–55. Springer-Verlag, 1992.

[GS95] Y. Gong and M. Sakauchi. Detection of regions matching specified chromatic features. *Computer Vision and Image Understanding*, 61(2):263–269, 1995.

[GWJ91] A. Gupta, T. Weymouth, and R. Jain. Semantic queries with pictures : the VIMYS model. In *Proceedings of the 17th int. conf. on Very Large Data Bases*, pages 69–79, Barcelona, September 1991.

[GYA97] J. Griffioen, R. Yavatkar, and R. Adams. A framework for developing content-based retrieval systems. In Mark T. Maybury, editor, *Intelligent Multimedia Information Retrieval*, pages 295–311. AAAI Press, Menlo Park, 1997.

[Hal89] Gilles Halin. *Apprentissage pour la recherche interactive d'images : processus EXPRIM et prototype RIVAGE.* PhD thesis, Université de Nancy I - CRIN, 1989.

[HKKZ95] T. Hermes, C. Klauck, J. Kreyss, and J. Zhang. Image retrieval for information systems. In Wayne Niblack and R. C. Jain, editors, *Storage and Retrieval for Image and Video Databases*, pages 394–405, San Jose, CA, 1995.

[HS80] M. Hassner and J. Sklansky. The use of markov random fields as models of texture. *Computer Graphics and Image Processing*, 12:357–370, 1980.

[HS92] R. M. Haralick and L. G. Shapiro. *Computer and robot vision.* Addison-Wesley, 1992.

[HS94] G. Healey and D. Slater. Using illumination invariant color histogram descriptors for recognition. In *IEEE Proc. of Conf. on Computer Vision and Pattern Recognition*, pages 355–360, Seattle, WA, 1994.

[HSD73] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Trans. System, Man and Cybernetics*, SMC–3(6):610–621, 1973.

[HSE+95] J. Hafner, H. Sawhney, W. Equitz, M. Flickner, and W. Niblack. Efficient color histogram indexing for quadratic form distance functions. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(7):729–736, 1995.

[HY82] H. Haneko and E. Yodogowa. A markov random field application to texture classification. In *Proc. Pattern Recognition and Image Processing*, pages 221–225, Las Vegas, Nevada, 1982.

[Kat92] T. Kato. Database architecture for content-based image retrieval. In *Storage and Retrieval for Image and Video Databases*, pages 112–123, 1992.

[KKH94] Y. Kiyoki, T. Kitagawa, and T. Hayama. A metadatabase system for semantic image search by a mathematical model of meaning. *SIGMOD RECORD*, 23(4):34–41, 1994.

[Kv87] J. J. Koenderink and A. J. van Doorn. Representation of local geometry in the visual system. *Biological Cybernetics*, 55:367–375, 1987.

[MD97] B. Merialdo and F. Dubois. An agent-based architecture for content-based multimedia browsing. In Mark T. Maybury, editor, *Intelligent Multimedia Information Retrieval*, pages 281–294. AAAI Press, Menlo Park, 1997.

[Mec95] M. Mechkour. *Un modèle étendu de représentation et de correspondance d'images pour la recherche d'informations.* PhD thesis, Université Joseph Fourier, Grenoble I, 1995.

[Meg95] C. Meghini. An image retrieval model based on classical logic. In *SIGIR'95*, pages 300–308, Seattle, WA, 1995.

[MHLF93] N. Mouaddib, G. Halin, Y. Lahlou, and O. Foucaut. Emir : modèle étendu et méthode de recherche pour objets complexes. premières spécifications. Technical report, Rapport Interne CRIN, Centre de Recherche en Informatique de Nancy, Vandoeuvre-les-nancy, France., 1993.

[MKNM95] B. Mehtre, M. Kankanhalli, A. Narasimhalu, and G. Man. Color matching for image retrieval. *Pattern Recognition Letters*, 16:325–331, 1995.

[MM96] B. Manjunath and W. Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.

[MP97a] T. P. Minka and R. W. Picard. Interactive learning with a "society of models". *Pattern Recognition*, 30(4):565–581, 1997.

[MP97b] B. Moghaddam and A. Pentland. Probalistic visual learning for object representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):696–710, 1997.

[MRT91] C. Meghini, F. Rabitti, and C. Thanos. Conceptual modeling of multimedia documents. *Computer*, pages 23–30, October 1991.

[NBW+93] W. Niblack, R. Barber, W. Wquitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC project : querying images by content using color, texture and shape. In Wayne Niblack, editor, *Storage and Retrieval for Image and Video Databases*, pages 173–181, San Jose, CA, 1993.

[OKS80] Y.-I. Ohta, T. Kanade, and T. Sakai. Color information for region segmentation. *Computer Graphics and Image Processing*, 13:222–241, 1980.

[OS95] V.E. Ogle and M. Stonebraker. Chabot : Retrieval from a relational database of images. *IEEE Computer*, 28(9):40–48, 1995.

[PH95] D. K. Panjwani and G. Healey. Markov random field models for unsupervised segmentation of textured color images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(10):939–954, 1995.

[PM95] R.W. Picard and T.P. Minka. Vision Texture for Annotation. *Multimedia Systems*, 3:3–14, 1995.

[Pra91] W. K. Pratt. *Digital Image Processing.* John Wiley and Sons, New York, second edition, 1991.

[QNT95] Y. Qiu Chen, M. S. Nixon, and D. W. Thomas. Statistical geometrical features for texture classification. *Pattern Recognition*, 28(4):537–552, 1995.

[RS92] F. Rabitti and P. Savino. Querying semantic image database. In *Storage and Retrieval for Image and Video Databases*, pages 69–78, 1992.

[RS95] N. Ramesh and I.K. Sethi. Feature identification as an aid to content-based image retrieval. In *Storage and Retrieval for Image and Video Databases*, pages 2–11, San Jose, CA, 1995.

[RW95] E. Rivlin and I. Weiss. Local invariants for recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(3):226–238, 1995.

[SB91] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[SC97] J. R. Smith and S.-F. Chang. Querying by color regions using the VisualSEEk content-based visual query system. In Mark T. Maybury, editor, *Intelligent Multimedia Information Retrieval*, pages 23–41. AAAI Press, Menlo Park, 1997.

[Sim96] B. Simonnot. *Modélisation multi-agents d'un système de recherche d'information multimédia à forte composante vidéo.* PhD thesis, Université Henri Poincaré, Nancy, 1996.

[SM95] C. Schmid and R. Mohr. Combining greyvalue invariants with local constraints for object recognition. In *CVPR*, 1995.

[Sma94] M. Smail. *Raisonnement à base de cas pour une recherche évolutive d'information ; Prototype Cabri-n. Vers la définition d'un cadre d'acquisition de connaissances.* PhD thesis, Université Henri Poincaré, 1994.

[SMEK96] E. Saber, A. Murat Tekalp, R. Eschbach, and K. Knox. Automatic image annotation using adaptative color classification. *Graphical Models and Image Processing*, 58(2):115–126, 1996.

[SO95] M. Stricker and M. Orengo. Similarity of color images. In Wayne Niblack and R. C. Jain, editors, *Storage and Retrieval for Image and Video Databases*, pages 381–392, San Jose, CA, 1995. SPIE volume 2420.

[Sri95] R. K. Srihari. Use of multimedia input in automated image annotation and content-based retrieval. In Wayne Niblack and R. C. Jain, editors, *Storage and Retrieval for Image and Video Databases*, pages 249–260, San Jose, CA, 1995.

[Tab94] A. Tabbone. Detecting junctions using properties of the laplacian of gaussian detector. In *International Conference on Pattern Recognition*, volume 2, pages 52–57, Jerusalem, Israel, October 9–13 1994.

[Ter94] B. Ter Haar Romeny. *Geometry-driven diffusion in computer vision.* Kluwer Academic Publishers, 1994.

[UA94] T. Uchiyama and M. A. Arbib. Color image segmentation using competitive learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(12):1197–1206, 1994.

[VDO85] L. Van Gool, P. Dewaele, and A. Oosterlinck. Texture analysis anno 1983. *Computer Graphics and Image Processing*, 29:336–357, 1985.

[WC92] C.-M. Wu and Y.-C. Chen. Statistical feature matrix for texture analysis. *CVGIP: Graphical Models and Image Processing*, 54(5):407–419, 1992.

[WDM+95] J.K. Wu, A. Desai-Narasimhalu, B. M. Mehtre, C.P. Lam, and Y.J. Gao. "core : a content-based retrieval engine for multimedia information systems". *Multimedia Systems*, 3:25–41, 1995.

[WLS95] T. Whalen, E.S. Lee, and F. Safayeni. The Retrieval of Images from Image Databases. *Behaviour & Information Technology*, 14(1):3–13, 1995.