acm PRESS

Proceedings of the Tenth Annual International ACMSIGIR Conference on Research & Development in Information Retrieval

edited by C.T. Yu and C.J. Van Rijsbergen

in cooperation with

British Computer Society

AICA (Italian Computer Society)

New Orleans, Louisiana USA
June 3–5, 1987

An Approach to Image Retrieval from Large Image Databases

F. Rabitti (*) - P. Stanchev (**)

(*) Istituto di Elaborazione della Informazione
National Research Council
Via S.Maria, 46 - 56100 Pisa (Italy)

(**) Institute of Mathematics
Bulgarian Academy of Sciences
G. Bonchev Str., 8 - 1113 Sofia (Bulgaria)

ABSTRACT

In this paper we address the problem of retrieving images from large image databases, giving a partial description of the image content. This approach allows a limited automatic analysis for image belonging to a domain described in advance to the system using a formalism based on fuzzy sets. The image query processing is based on special access structures generated from the image analysis process.

1. Introduction: The problem of image retrieval

There is a growing number of application areas where digital image processing is a primary concern. These application areas include: interactive computer-aided design, geographic data processing, remote sensing of earth resources, regional economic and health data processing, cartographic and mapping applications [Chan81a]. In order to handle the growing amount of electronically stored digital images, pictorial data base systems have been proposed. A key function of these systems is image retrieval: suitable query languages and access structures for images must be implemented [Chan81b].

Most of these systems are application specific, that is, the manner in which images are stored, organized and retrived is specific of a certain application and cannot be generalized to different applications. In many systems, images are stored in files which are linked to other image files into structures for image retrieval and presentation according to the logic of the application [Hero80].

Other systems are based on a underlying database whose schema describes the image content and composition [Econ83]. These systems can exploit the flexibility of a Data Base Management System (DBMS). In particular, the query language and the access structures implemented in the DBMS are very powerful for retrieval operations. But they are limited by the fact that the content of the images must be described using data models which were developed for database systems rather than image systems, and so they lack the expressive power needed for images. In fact, images are inherently different from database records: these records can be divided in different classes according to their interpretation. The record structure (i.e. the schema) can be described at class (i.e. type) level. Since the ratio of instances per type is very high for database systems, the resulting storage structures and access methods are very efficient. On the contrary, each image may have its own particular structure, and a whole semantic network [Tsic82] may be necessary to completly describe each image instance.

A new application area where the problem of image storing and retrieval has been addressed is office automation. A new information object is defined, the multimedia document, where images are combined with attribute data, text, and voice [Tsic83]. Systems for the storage and retrieval of large volumes of these documents are under study [Bert85]. One of the main functions of these systems is the access to multimedia documents based on their content. However, while these systems incorporate efficient access methods for attribute data and text, they can do very little for images, as well as voice. Their approach is often to link images to other more structured components of the multimedia document, and then exploit the combined access to them [Tsic83].

For the retrieval of documents by content, multimedia document systems can exploit very efficient techniques borrowed from DBMS [Ullm82], when formatted data is concerned, or from Information Retrieval Systems (IRS) [Salt83], when text is concerned. Comparable techniques are not available for images.

The main conceptual problem in dealing with images derives from the difficulty to exactly define and interpret the content of images. Images can be very rich in semantics, but are subject to different interpretations according to the human perspective or the application domain. On one hand, it is difficult to recognize the objects (with the associated interpretation) contained in an image, on the other hand is difficult to determine and represent the mutual relationships among these objects, since they form structures which vary greatly from image to image.

One approach could be to face directly the problem of image understanding. This implies the combined use of sophisticated pattern recognition techniques (usually expensive in terms of computing power and special hardware required) and advanced artificial intelligence techniques to represent and organize the knowledge expressed in the images [Hans70]. In this case, access structures could be built in terms of the image representations in the image knowledge base. However, this class of expert systems is still in an experimental phase, limited to some application specific prototypes [Flee84], whose results cannot be generalized.

Another approach could be to apply DBMS or IRS techniques. However, with respect to DBMS's, it is difficult to recognize regular structures of objects contained in images, and then organize image instances into a limited number of types, to which the interpretation is associated. This is the approach required by the strictly typed data models adopted in database systems [Tsic82]. In IRS, instead, a free formatting of text is allowed, usually respecting some loose hierarchical structuring in sections, sub-sections, paragraphs and sentences. These systems do not attempt to undestand the text (unless some expert system approach is adopted), but still allow an effective retrieval on text. In fact, as opposed to image objects, they can exactly recognize words (as ASCII patterns), on which they base their retrieval capabilities, with the possible help of thesauri to support synonyms [Salt83]. This is possible, in case of text, because a common semantic is associated to the words used in the natural language. Hence, both DBMS and IRS approaches cannot be directly applied to the image retrieval.

In addressing the problem of image retrieval on large volumes of stored images, if we want to think of a system doing for images what DBMS and IRS do for formatted data and text, we must accept some indeterminateness, characteristic of images, and then deal with the inaccuracy introduced by this fact. We have recognized the exigency of adopting a non-boolean logic, allowing some form of approximate reasoning. Thus, we have decided to investigate an approach based on fuzzy sets. In fact, fuzzy sets seem appropriate for representing different degrees of similarity. This would allow one to recognize images, and contained objects, at a certain degree (i.e. certainty factor), and define classes of images/objects with unsharp boundaries.

In this paper, an approach is presented for the analysis and retrieval of images (we will refer to examples of both graphic and bit-map images). With this approach, a limited automatic analysis of the images is performed before storing the images in the database. However it is required that the images belong to a specific domain which must be described in advance to the system using a formalism based on fuzzy sets.

The real goal of this image analysis process is not to attempt any deep image understanding, but is to support the image retrieval process. In fact, the system allows the user to query the images, already analyzed and stored, giving some specification of their content. The image query processing is based on special access structures (i.e. image indices) which are generated when the image analysis is performed. The query specification can be done using either a linear query language, or a "by example" approach, in which the user edits an "image query prototype" containing the essential features of the images to be retrieved.

In essence, this paper investigates the possibility of having systems doing for images what is today possible to do with formatted data in Database Management Systems and with text in Information Retrieval Systems.

In the following, the three fundamental phases of image analysis, image storing/indexing, and image retrieval will be presented in detail, discussing the main technical difficulties which may arise.

2. The role of fuzzy sets

Let X be a classical set of objects, called the universe, whose generic elements are denoted as x. Membership in a classical subset A of X is often viewed as a characteristic function from X to (0,1), such as

$$\mu_A(x) = \begin{cases} 0 \text{ iff } x \notin A \\ 1 \text{ iff } x \in A \end{cases}$$

{0,1} is called a valuation set. If the valuation set is measured on the real interval [0,1], A is called a fuzzy set [Zade65]. $\mu_A(x)$ is the grade of membership of x in A.

Informally, a fuzzy grammar may be viewed as a set of rules for generating the elements of a fuzzy set [Lee69]. More precisely, a fuzzy grammar is a quadruple $G=(V_T,V_N,P,S)$ where, V_T is a set of terminals, V_N is a set of non-terminals ($V_T \cup V_N \cap = \emptyset$), P is a finite set of productions, $S \in V_N$ is the initial symbol. The elements of V_N are labels of certain fuzzy sets on V_T * (The symbol "*" indicates a free monoid X* over the set X). The elements of P define fuzzy sets in $(V_T \cup V_N)^*$ (referred as fuzzy productions) and for the case of context-free fuzzy grammars are expressions of the form:

$$A \xrightarrow{\varrho} \beta, \quad A \in V_N \quad , \quad \beta \in (V_N \cup V_T)^* \; ,$$

where $0 \leq \varrho \leq 1$ is the grade of membership of β given A. In case of general grammars, let $\alpha_1, \alpha_2, \ldots, \alpha_m$ be strings in $(V_T \cup V_N)^*$ and

$$\alpha_1 \xrightarrow{\varrho_2} \alpha_2, \; \alpha_2 \xrightarrow{\varrho_3} \alpha_3, \ldots, \alpha_{m-1} \xrightarrow{\varrho_m} \alpha_m$$

$$(0 \leq \varrho_i \leq 1)$$

be productions. The expression

$$\alpha_1 \xrightarrow{\varrho_2} \alpha_2 \xrightarrow{\varrho_3} \alpha_3 \cdots \xrightarrow{\varrho_{m-1}} \alpha_{m-1} \xrightarrow{\varrho_m} \alpha_m$$

will be referred to as a derivation chain from α_1 to α_m. A string x of V_T * is said to be in the fuzzy language L(G) (that is, it is a "statement") iff x is derivable from S. The grade of membership $\mu_G(x)$ of x in L(G) is given by:

$$\mu_G(x) = \sup \min (\mu(S \xrightarrow{\varrho_1} \alpha_1), \mu(\alpha_1 \xrightarrow{\varrho_2} \alpha_2), \ldots, \mu(\alpha_m \xrightarrow{\varrho_{m+1}} x)),$$

where $\mu(\alpha_i \xrightarrow{\varrho_{i+1}} \alpha_{i+1})$ is the non-null ϱ_{i+1} such as $(\alpha_i \xrightarrow{\varrho_{i+1}} \alpha_{i+1}) \in P$, for $i=0,1,\ldots,m$, $\alpha_0 = S$ and $\alpha_{m+1} = x$.
The supremum is taken over all derivation chains from S to x. In any derivation chain, the minimum ϱ_i is taken.

For the application of fuzzy grammars in our environment we have slightly changed the definition. S is replaced with the set V_S, where the elements of V_S are labels for a syntactic category of the type "object description" (that is, they are "object statements").

3. Image analysis using fuzzy approach

Two general types of images are considered: graphic images and bit-map images. As example, two application environment can be envisaged for such images:

- CAD systems for complex engineering applications (eg. aircraft design). Large quantities of technical designs, in the form of graphic images, are produced with the support of a CAD editor. It would be of great support to the designer a system which allows him to retrieve already produced designs concerning parts of the projects related to the one under specification.

- Satellite image processing. In these systems huge amounts of bit-map images are produced. These images are usually processed by forming, framing, enlarging, filtering, brightness adjusting, etc.. Any available system supporting an automatic image analysis, even if limited and prone to errors, would be of enormous help. This system could then allow the retrieval of these images giving some specification of some wanted objects (eg. a particular building structure corresponding to an airport).

The methodology we propose for the image analysis could be expressed in the following steps (Fig. 1.)

1) Element recognition

For both the image types, it is supposed that the image can be decomposed into known elements. For CAD systems these elements can be described in terms of the graphic primitives of the system graphical editors. For bit-map image processing, the relevant elements must be suitably described.

In this step the image space is partitioned into meaningful regions, corresponding to image elements. The elements are the basic components which will constitute the building blocks of the image structure. This step, often called segmentation in image processing [Gonz77], is made easier in our approach since we know approximately what these elements should be like. In fact admissible basic elements should correspond to some terminal symbols of the fuzzy grammar. Thus, the system will try to match the tentative elements in the image with the pictorial or graphical representation (i.e. icons stored in a catalog) of the terminal symbols of the grammar adopted for the application. In this process, different variations as changes in size, rotation, translation, using discrete Fourier transforms, can be attempted.

The main problem arises when partial element overlapping occurs in the image. The system might not be able to exactly identify the single elements. In this case, the fuzzy approach may help: elements partially seen on the image (due to overlapping) are recognized with different certainty factors, according to the degree of similarity to the stored representations of the basic graphical elements. In terms of fuzzy grammars, this means that terminal productions of the form $A \xrightarrow{\rho} a_i$, where a_i is the terminal element, will have ρ dynamically computed, case by case, during the element recognition step.

The output of this step is the set of basic elements recognized and their relative positions. Also the degree of recognition for each element is introduced. Different recognition possibilities (grammar terminal symbols associated with different certainty factors to the same image elements) will be recorded for the further image analysis. This is consistent with the fuzzy grammar approach exploited in the following steps of image analysis.

2) <u>Object recognition</u>

In this step the system tries, with a recursive procedure, to organize the image elements into meaningful relational structures called objects [Gonz77]. Objects will correspond to non-terminal symbols of the adopted fuzzy grammar. When applying a grammar production for an object the system must check if the basic elements recognized are present in the production and if their relative positions are consistent with the corresponding positional operators in the production. The value (ranging from 0 to 1) associated to the production applied will be associated to the object to represent its degree of recognition.

This procedure can be applied recursively in order to obtain higher level objects, as defined with fuzzy productions in the grammar. For every production path followed, the resulting degree of recognition will be the minimal value associated to any applied production. This process will end when no more fuzzy productions are applicable to define more complex objects. If an object, or part of it, corresponds to several object statements in the fuzzy language with several levels of membership, we will choose a set of statements with the highest membership values. The solution of keeping more object statements associated to any image object, allows the system to keep more options open in further steps of the analysis process, realizing a limited form of back-tracking.

It can also happen that an object cannot be correctly recognized by the system if there is no corresponding description for it in the grammar. In this case, either new productions are inserted in the grammar for the object, or this object is disregarded. In fact, in this approach the system must know in advance the fuzzy description of the images to be stored. The result of this step is a composed fuzzy set (set of fuzzy sets) containing the non-terminals for the recognized objects and the associated membership values for the image.

3) <u>Added information definition</u>

In this step more information is added to the image, in the form of attributes logically connected. In fact the user can define attributes describing properties associated to image objects, in the form of attribute name, type and value and certainty factor. This fourth attribute component is optional, but is here allowed due to the fuzzy set approach (if not otherwise specified, it is set to 1).

These attribute can be entered by a user when a new image is compiled or when an already existing image is retrieved and commented. We could also think of an automatic process doing this work, implementing some expert system technique based on rules specific of the application domain.

4) <u>Image clustering</u>

A useful technique applied in text document retrieval is the "clustering" [Salt83]. This technique can be also applied to image, once they have been analyzed according to the previous steps. Since it is possible to define a distance measure between fuzzy sets (eg. a relative Hamming distance between fuzzy sets can be defined as a distance measure) it is possible to define the distance between image objects, which are fuzzy sets, and the distance between images, as the weighted distance between the constituting objects. This important consequence allows the clustering of images.

According to the application, several

image classes are defined in order to nearly cover the application domain. For each class, an image representing that class is defined (i.e. a centroid). The distance from an image to the centroid image represents the inverse of the degree of membership of that image to the class.

The image description is extended with degrees according which the image belongs to the classes of images preliminary defined in the system. This task is performed by accounting the distance between the image description and the description of the image representative of each class. This step may reduce significantly the comparisons in the image retrieval process.

3.1. Problems in the analysis process

A few problems arise during the image analysis process. First of all, it is necessary to refer to a specific application domain. The proposed image analysis method requires the basic elements and possible objects to be known in advance by the system. Otherwise the results of the analysis becomes inacceptable. Hence, a system set-up phase is necessary to describe the application domain to the system.

Furtrermore, it is necessity to implement a form of approximate reasoning, in particular regarding the following points:

1) Recognition of image elements

Necessity of approximate reasonig in recognizing image elements occurs mainly in bit map image processing. After image segmentation, the image elements are matched with the defined basic elements. In this process, differences in size, colour, view angle, overlapping parts, etc. can be accepted, but may modify the certainty factor associated to the terminal grammar production of that element. This fact often lead to identifying an image element as more than one basic element with different degree of recognition.

In CAD systems there is no such problem. Usually the graphic images are constructed by CAD editors which by themselves use basic elements. So the image elements are already available (although they may appear with some changes in the image: in size, in view angle, etc.).

2) Object definition incompletness

Sometimes incompletness can occur in object definition. The reasons could be a partial overlapping of some objects or the lack a suitable object description. In this latter case, new grammar productions must be entered or the object is diregarded.

3) Hierarchy of complex objects

In many cases there can be complex objects, that is, objects composed of simpler objects. Hierarchies of more than one level of objects can be established. In the analysis phase, this implies that there can be several recognition paths for the complex objects, with different certainty factors. Different recognition paths must be preserved, in order to allow some form of backtracking when more complex objects are recognized.

3.2. Setting-up the analysis process

Before the analysis is actually performed on the images to be entered into the system, it is necessary to set up particular strucures which describe to the system the elements and the object to be recognized in the images. They are obviously related to the particular application domain of the system. The system installation phase should prepare the following steps of the analysis process:

1) Element recognition

The set of basic elements $a_1, a_2, \ldots, a_n$ is to be constructed.

In CAD systems the elements are constructed from several graphic primitives, such as indented line, rectangle, arc, circle, which are stored in graphical form.

In bit map images, the basic elements are entered by some available image input devices and are stored in the form of a vector for every resolution point.

2) Recursive object recognition

The objects which could be recognized by the system are introduced by a fuzzy grammar. The fuzzy grammar represents a fuzzy description of the objects. Terminal symbols for the grammar are: (1) the names of the basic elements $a_1, a_2, \ldots, a_n$, constructed as described in the previous step; (2) the symbols used for describing the connections between the elements.

An example of the positional symbols is presented in Fig.2., where I,W,N,S, denote the four cardinal points and "in" and "out" denote that an object is inside or outside another object. Other positional terminal symbols can be "intersect" for intersection, "(...)" for inclosure, etc.. Nonterminal symbols are the names of the objects and object parts chosen from by the user and consist of several mutually connected elements. The grammar productions describe the connection between the elements of the object, and between different parts of the object. The degree of recognition of the terminal elements participating in the production means that the production rule is applicable iff the correspondent element in the string has degree of recognition greater or equal to the given one.

3) Added information definition

The purpose of attributes is to add more information to the object descriptions. Some attributes (name and type) can be defined for certain objects of the fuzzy grammar. Other attributes, for internal use (eg. relative position, color, relative size,etc.), can be calculated during the object recognition phase. Specific rules must be given during this set-up phase.

4) Image clustering

The definition of image classes is often required also in standard pattern recognition approaches [Verh80]. In our case, p different images, particularly meaningful for the application domain, will be initially defined. They will constitute the representative images, or centroids, of the p image classes $K_1, K_2, \ldots, K_p$.

3.3. Execution of the analysis process

The analysis process is executed on the images to be entered into the system. It is structured into these steps.

1) Element recognition

In CAD systems the picture is created by a CAD editor from basic elements, with the help of operations such as: translation, scaling, rotation, invisable line deleting, etc.

In bit map images, after image segmentation, the degree of similarity between every obtained element and the basic elements (using some suitable rules, for example, the percentage of the figure that can be matched) are calculated, normalized and attached as certainty factors (i.e. degrees of recognition of the element).

2) Recursive object recognition

It is performed in two steps:

STEP-1:
Organization of basic elements into meaningful relational structures called objects

The point of the step is the composition of the elements in preliminary defined objects $O_1, O_2, \ldots, O_l$. The step starts by obtaining a string for every group of elements which contain terminal symbols, corresponding to elements and their position in the image.

Different algorithms could be applied for the generation of the fuzzy language strings. One of such algorithms can be: First, all elements which are situated one under the other are described. Second, the strings obtained are connected from left to right. In case some elements are contained in another element, the same procedure is recursively applied and a string is generated which is included in the external one by parenthesis.

A top-down analysis is then applied on the obtained strings, using the following rules:

- A production rule is applicable iff the correspondent element in the string has degree of recognition greater or equal to the given value;
- All possible ways to obtain V_s elements (i.e. objects) are followed. Among these recognition paths, those which have the highest degree of recognition are chosen;
- In building the parse tree, the degrees of recognition of all productions are considered. The resulting degree of recognition is the minimum of the degrees of recognition of the applied productions;
- The analysis is applied on the subparts of the obtained strings (case of overlapping or when more than one object corresponds to the string);
- The analysis is performed

until the string is decomposed completely into object strings. If this is not possible, either the system stops at this point (i.e. some objects are not recognized) or the system set-up phase is entered again and productions for a new object or new productions for existing objects are defined.

In graphic images, a certain set of elements could be recognized as more than one object, with different certainty factors. In such cases, no more than k admissible descriptions of the images are stored. These image description have the higher sum of the degrees of recognition of the included objects.

In bit map images, several acceptable strings may correspond to the same group of elements (in all of these strings, the elements participate with different degrees of recognition). After the analysis of all strings, a set of recognized objects is obtained. As for graphic images, at most k admissible descriptions of the image are stored.

STEP-2:
Organization of objects into more complex objects

After Step 1 productions are applied again aiming at obtaining more complex objects in V_s . The procedure is repeated recursively as long as possible. No more than k admissible descriptions of the image are stored.

After Step 1 and Step 2 at most k sequences as (1) are obtained for the image:

$$(1) \quad O_1[\mu_{11}],\ldots,O_1[\mu_{s_1 1}],\ldots,$$
$$O_l[\mu_{1l}],\ldots,O_l[\mu_{s_l l}]$$

This expression denotes an image with l distinct objects. Each object may appear several times: s_j denotes the number of instance that of the object j in the image, and μ_{ij} is the degree of recognition of object j in its i-th appearance in the image. Each sequence (1) constitutes an interpretation of the image, in terms of its constituting objects.

In an exaple regarding aerial pictures, object 1 could be a tanker ship, object 2 a battle ship, object 3 an aicraft-carrier, object 4 a dock, object 5 a dockyard, etc. If s_1 is 4, it means that four tanker ships have been recognized in this interpretation of the image. If μ_{11} =0.8, it means that the first instance of the tanker ship has been recognized with a certainty factor of 0.8, etc.

3) Computation of attributes

A vector with attributes values and their certainty factor is added to every object in the image description, so that the k sequences of type (1) for the image become:

$$(2) \quad O_1[\mu_{11},\bar{a}_{11}],\ldots,O_1[\mu_{s_1 1},\bar{a}_{s_1 1}],$$
$$\ldots,O_l[\mu_{1l},\bar{a}_{1l}],\ldots,O_l[\mu_{s_l l},\bar{a}_{s_l l}]$$

where $\bar{a}_{ij}$ is the vector with the attribute values and certainty factor of object j in its i-th appearance in the image. A sequence (2) is an interpretation of the image, containing also object attributes.

Referring to the aerial picture example, an attribute (of the attribute vector) attached to the object 1 (the tanker ship), first instance, could be: size is small with certainty factor 0.9. For the object 2 (the battle ship), third instance, could be: ship class is cruiser with certainty factor 0.4.

4) Image clustering

The clustering process consists in adding to the obtained image description information the membership degrees of the image to every defined class. This is made by comparing all the obtained image interpretations (i.e. sequences (2)) with all the given class descriptions (i.e. the class centroid images). The membership degree of an image to a class is assumed to be the inverse of the minimum distance of any of the k image interpretations to the class representative image. After this computation, the clustering description of the image is expressed as a sequence (3):

$$(3) \quad K_1[\mu_1],K_2[\mu_2],\ldots,K_p[\mu_p]$$

where μ_i is the membership degree of the image to the class K_i .

Referring to the aerial picture example, the previous interpretation of the image has a membership 0.8 with the cluster "military harbour", and a membership 0.2 with the cluster "commercial harbour".

4. Storage and retrieval process

4.1. Image storage

In our approach, along with the file containing the image presentation we must also store the added information resulting from image analysis.

The image file will contain some suitable representation of the immage, according to the particular environment. The main difference exists between graphic and bit map images. In the former case, the image will be represented by the graphic primitives used for its composition. In a system based on the GKS graphic standard [ANSI84], the image file will be constituted by the GKS metafile, stored as a set of segments, each containing graphic elements with the associated attributes. In the latter case, the bit map image can be stored in raster form (possibly using some type of compression, such as run-length encoding) or using special structures such as binary trees, quad-trees, pyramids, etc. [Ball82].

The complete image description, resulting from the analysis phase, can be formally considered as a complex fuzzy set (i.e. a fuzzy set from fuzzy sets) [Kauf75]. However, it is essential to find a suitable storage representation for this added information since the efficiency of the image retrieval process is based on it. For this purpose, it is useful to define some type of indexing on fuzzy sets or on part of them.

The image access information is stored in an "image header", associated to the image file. In this header we store:

- One sequence (3), containing the image clustering description. Each term of these sequence containg the membership degree of the complete image in one of the image classes of the application. This kind of information is more synthetical, since it refers to the image as a whole.

- k sequences (2) (or (1) if image object attributes are not supported by the system), each containing an image interpretation. Only the k most plausible interpretations are stored. Each interpretation is an image descrition in terms of image objects. In fact, each term of a sequence corresponds to a particular object occurrence (in that image interpretation), with the associated certainty factor. The same object may appear more times in the sequence, one for each appearance of that object in the image interpretation. This kind of information is more analytical, since it refers to the composition of the image.

Access structures, that is, the image indices, can be built for a fast access to image headers. Two type of indices are constructed:

- The object index

  For each object in V_s of the fuzzy grammar, a list is maintained. Each element of the list is constituted by a triplet (INT, CF, IMH), where IMH is a pointer to an image header, meaning that the object is present in that image, INT identifies one of the k image interpretations containing that object, CF is the associated certainty factor. For query processing, it is very important to maitain the list in decreasing order of CF.

- The cluster index

  For each class defined in the application, a list of elements (MF, IMH) is maintained. IMH is a pointer to an image header, corresponding to an image with a non-null degree of membership to this cluster, and MF is the value of the membership factor. For query processing, it is very important to maitain the list in decreasing order of MF.

4.2. Image retrieval

The user has the option between two types of interfaces for querying images: a query language interface [Ull82] and a "by-example" interface [Zloo77].

a) Using a query language interface, the user specifies a query statement of the form:

   RETRIEVE IMAGES <cluster_clause> CONTAINING <object_clause>

   The <cluster_clause> is a boolean combination of <cluster_predecate>'s, each of the form:
   <class_name> <membership_value>
   This predicate indicates that the images in the database with a similarity to the named class higher than the <membership_value> should be retrieved (as requested in the boolean expression). The <cluster_clause> may be missing if the <object_clause> is present.

   The <object_clause> is a boolean combination of <object_predicate>'s. Each of them is a parenthesized string of <positional_operator>'s (eg. inE, outW, etc.) and <object_term>'s. Each <object_term> has the form:
   <object_name> <certainty_factor>

The <object_clause>must be evaluated, according to he boolean expression and the positional operators, taking into account only the images in the database containing those object, named (as <object_name>) in the <object_term> (as <object_name>), with a degree of recognition higher than the <certainty_factor>.

Referring to the previous example of an image database of aerial pictures, a possible query could be:

```
RETRIVE IMAGES
    ('military harbour' 0.6) OR
    ('commercial harbour' 0.4)
CONTAINING
   (AT LEAST 10 'tanker ship' 0.9
   WITH size:'large' 0.5)
   AND (('aircraft carrier' 0.8)
   OR ('battle ship' 0.7 WITH
   ship_class:'heavy cruiser'));
```

With this query, we want to retrieve the images concerning military and commercial harbours, containing: at least ten large tanker ships and either an aircraft carrier or a battle ship of type heavy cruiser.

b) Using the "by-example" interface, the user queries the image database describing to the system a prototype of the wanted image. In the prototype creation the user should put all his recollection about the objects contained in the target images. This process reflects bottom-up the image analysis process. Starting from the basic elements, as presented by the system using icons (if required, they can be manually translated, rotated, scaled, etc.), the user will build increasingly complex objects that he thinks should be contained in the image. During this process the fuzzy grammar productions are followed and the corresponding fuzzy sets are generated, like in the image analysis process. During this process, the system may tell the user that the wanted objects cannot be constructed according to the grammar rules. Thus, even in this early phase he can be sure that the wanted image is not contained in the image data base (or, if presented, it was interpreted in a different manner and so cannot be reached using the available access sructures).

As part of this query, the user will define the degree of similarity he wants between the prototype image and the retrieved images. This accuracy parameter will be translated in terms of fuzzy set distance.

The image query prototype definition is equivalent, even if more user friendly, to specifying an <object_clause> whose <object_predicate>'s are in conjunction. If the query prototype corresponds to a complete image, a <cluster_clause> is implicitely specified, whose <cluster_predicate>'s are in conjunction. A <cluster_predicate> is constructed for any image class defined in the application, with <operator> set to "higher" and <membership_value> set to the accuracy parameter of the query.

Since the use of fuzzy set techniques for image description allows one to define a distance between two images, query processing can be performed, in principle, computing the distance between the fuzzy set associated to the query specification and the fuzzy sets associated to the stored images. In practice, performances are greatly enhanced using the image access structures, that is, cluster indices to process <cluster_clause>'s and object indices to process <object_clause>'s.

All the stored images having a distance lower than the required accuracy will constitute the query answer set. With this approach, the query answers can be ordered by decreasing similarity to the query specification, so a user may limit the size of the answer and can receive a ranked output of the retrieved images (These advantages are typical of the information retrieval techniques [Salt83]).

A browsing facility becomes essential in this approach. That is, the user should have the possibility of browsing through the retrieved images. Since the image retrieval is not an exact process (there is no exact way of defining image similarity) and even the user may forget essential characteristics of the sought images, several non pertinent images can be retrieved as a result of a query. Moreover, relevance feed-back and query reformulation [Salt83] are emphasized, since at any moment the user can go back to the query formulation step, if dissatisfied by the results that he is getting, and change some aspects of the query specification.

5. Conclusions and future work

The proposed approach appears promising in dealing appears promising in dealing with some class of images (for example, satellite pictures, CAD drawings) which are stored in large image database, dealing with well sepified application domains. However, doubts remain about how a system based on the fuzzy set approach can perform in a real environment.

In order to demonstrate the proposed

approach we have realized a very simple experimental prototype, called ISA system (Image Structure Analysis) supporting the analysis, storage and retrieval of graphical images. ISA has been implemented on a IBM PC/AT, equipped by a Professional color video display (resolution 640*480) and Professional Graphic board, with 256 colors, supporting GKS graphic primitives (firmware support for segments, 2d and 3d transformations, etc.). The PASCAL programming language under MS/DOS has been used.

With respect to the complete approach presented in the paper, ISA presents important limitations:

1) image elements are exactly determined, so they can be directly associated to terminal symbols of the grammar;

2) image objects are always simple: no hierachy of complex objects can be defined in the grammar;

3) objects cannot overlap;

4) images are not clustered: image classes are not defined.

An important simplification that derives from the previous restrictions is that only one image interpretation results as image description.

In ISA the user stores and retrieves images driven by a simple command language. As a CAD system we have used GRAFFITO [Mane85]. A simple example of an application in ISA is illustrated in Appendix A.

A more advanced prototype is under development. It will include the more advanced features outlined in this paper. This new prototype will run on SUN workstations, will be written in C and will use the SUNCORE graphic package.

Moreover, it will not be required to write the grammar rules at system set-up time for the specific application. Instead, grammar rules will be derived from the objects interactively drawn with the mouse on the workstation screen. Moreover, the creation of clusters will be done by the system using some learning by example approach.

REFERENCES

[ANSI84] ANSI X3H3/83-25r3 "Graphical Kernel System", Special issue, Computer graphics, Febr. 1984.

[Ball82] Ballard, D. and Brown, C., Computer Vision, Prentice Hall, 1982.

[Bert85] Bertino, E., Gibbs, S., Rabitti, F., Thanos, C., Tsichritzis, D., "Architecture of a Multimedia Document Server" Proc. 2nd ESPRIT Technical Week, Brussels, Sept. 1985.

[Chan81a] Chang, S.K., Kunii, T.L., "Pictorial Data-Base Systems", IEEE Computer Vol.14(11), pp.13-21, 1981.

[Chan81b] Chang, N.S., Fu, K.S., "Picture Query Languages for Pictorial Data-Base Systems", IEEE Computer Vol.14(11), pp.23-33, 1981.

[Econ83] Economopoulos, P. and Lochovsky, F., "A system for managing image data," Proc. IFIP Congress, 1983.

[Flee84] Fleet, D.J., Jepson, A.D., Hallet, P.E., "A spatio-temporal model for early visual processing", Tech. Rep. CSRI, Univ. of Toronto (Canada), 1984.

[Gonz77] Gonzalez, R., Wintz, P., "Digital Image Processing", Addison-Wesley, 1977.

[Hans78] Hanson, A.R., Riseman, E.M., "Computer vision systems", Academic Press, 1978.

[Hero80] Herot, C.F., "Spatial management of data," ACM Trans. Database Syst. Vol. 5(4), pp.493-513, Dec. 1980.

[Kauf75] Kaufmann, A., "Theory of Fuzzy Subsets", Academic Press, 1975.

[Lee69] Lee E., Zadeh L., "Note on fuzzy languages", Information Sciences Vol.1, pp.421-434, 1969.

[Mane85] Manev N., Nicos E., Banchev,B., "Interactive graphic system GRAFFITO", Proc. Conf. Problems of complex control systems, Varna, pp.523-531, 1985.

[Salt83] Salton, G. and McGill, M.J., Introduction to Modern Information Retrieval, McGraw-Hill, 1983.

[Tsic82] Tsichritzis, D. and Lochovsky, F., Data Models, Prentice-Hall, Englewood Cliffs, N.J., 1982.

[Tsic83] Tsichritzis, D., Christodoulakis, S., Economopoulos, P., Faloutsos, C., Lee, A., Lee, D., Vandenbroek, J., and Woo, C., "A multimedia office filing system," Proc. Ninth Int. Conf. on Very Large Data Bases, 1983.

[Ullm82] Ullman, J., Principles of Database Systems, Computer Science Press, 1982.

[Verh80] Verhagen C. et al., "Progress report on pattern recognition", Reports on Progress in Physics, vol.43, No.6, pp. 785-831, 1980.

[Zade65] Zadeh L. "Fuzzy sets", Inf. Control No. 8, pp.338-353, 1965.

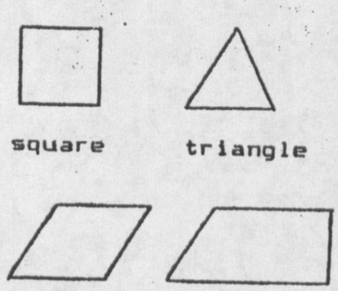
[Zloo77] Zloof, M.M., "Query-by-Example: A database language," IBM Systems J. Vol. 16(4), pp.324-343, 1977.

APPENDIX A

1) System set-up

A) Element creation

With the help of GRAFFITO we create:



square triangle

romboid trapezoid etc.

B) Object definition

With the help of ISA we define the following grammar:

```
Terminals >
in  inW  inN  inS  inE  inNW  inSW
inNE  inSE  outN  outS  outE  outNE
outSE  intersect  (  )
square  triangle  romboid  trapezoid;

Nonterminals >
house  facade  front  sideview  wall
window  door  chimney roof ;
```

```
Objects > house

Rules >
  1: house => 1.00 -->
     sideview outE house ;
  2: house => 1.00 -->
     facade ;
  3: door => 1.00 -->
     square ;
  4: window => 0.80 -->
     square ;
  5: window => 1.00 -->
     square outE square ;
  6: chimney => 0.80 -->
     trapezoid ;
  7: wall => 1.00 -->
     square ;
  8: wall => 0.80 -->
     ( window in square ) ;
  9: wall => 1.00 -->
     ( window door in square ) ;
 10: front => 1.00 -->
     triangle ;
 11: front => 1.00 -->
     chimney outS triangle ;
 12: roof => 1.00 -->
     romboid ;
 13: roof => 1.00 -->
     chimney outS romboid ;
 14: facade => 0.95 -->
     front outS wall ;
 15: sideview => 1.00 -->
     roof outS wall ;
 16: chimney => 0.90 -->
     square ;
 17: wall => 0.95 -->
     (door inS square) ;
```

C) Attribute definition

With the help of ISA we enter the following set of attributes:

```
Attributes >
object -> house
1: color - enumerated with values:
     white red yellow ;
2: windows - rule done with sequence:
     5 ;
3: big_size - attribute done,true if:
     (windows > 20 ) and
     (colour = yellow) ;
```

2) Analysis

We create with the help of GRAFFITO the image in Fig. 3. The following string corresponds to that image:

square outS romboid outS (square outE square in square) outE triangle outS (square inS square);

After processing by the given grammar the object "house" is recognized with certainty factor 0.8.

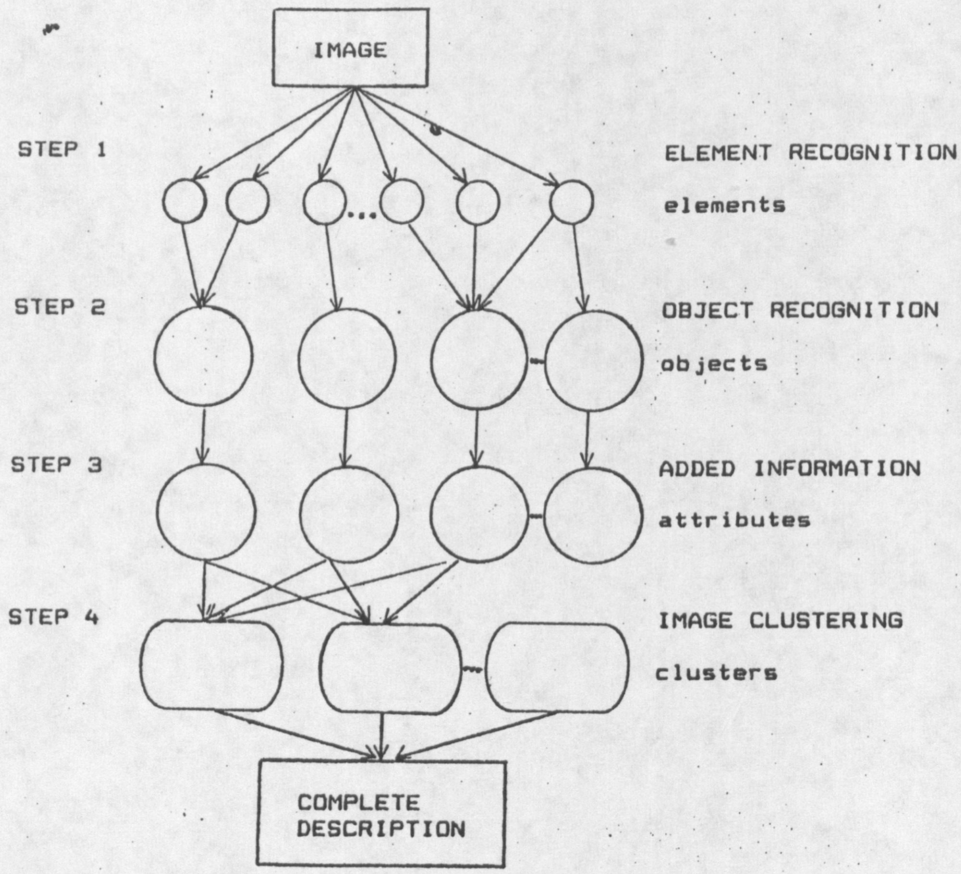


Fig. 1

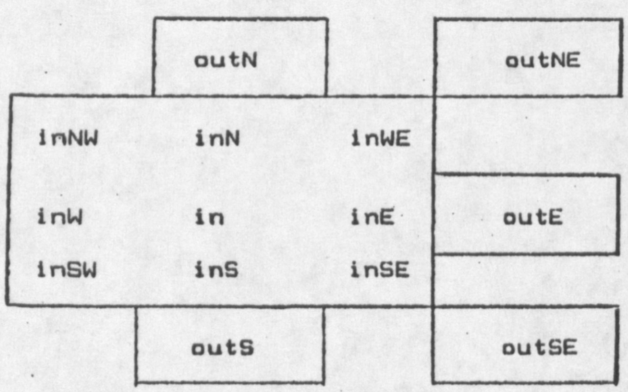


Fig. 2

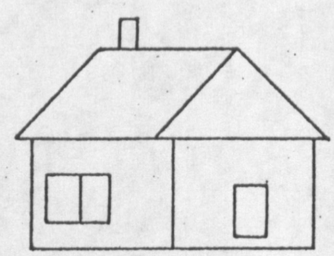


Fig. 3

(76.) Rabitti F., Stanchev P., "An Approach to Image Retrieval from Large Image Databases", Proc. **10-th Annual Intern.** ***ACMSIGIR Conf. on Research & Development in Information Retrieval*****,** New Orleans, Louisiana, USA 1987, 284-295.