



# Journal of Applied Sciences

ISSN 1812-5654

**science**  
alert

**ANSI***net*  
an open access publisher  
<http://ansinet.com>

## Partitioning an Image Database by K\_means Algorithm

Houaria Abed and Lynda Zaoui

Laboratory Systems Signals Data, Department of Computer Science, Faculty of Science,  
University of Science and Technology Mohamed Boudiaf, Oran B.P.1505 M'Naouar Oran, Algeria

**Abstract:** Unsupervised classification has emerged as a popular technique for pattern recognition, image processing and data mining. It has a crucial contribution in the resolution of the problems arising from content-based image retrieval. In this study, we present K\_means clustering algorithm that partitions an image database in cluster of images similar. We adapt K\_means method to a very special structure which is quadree. The goal is to minimize the search time of images similar to an image request. We associate to each image a quad-tree which represents the characteristics of the image and store a base of images in a data structure called generic quadtree. It minimizes the memory space of set of image by the sharing of common parts between quad trees and speeds up several operations applied to images. The image similarity is based on a distance computed from the differences between the quad trees encoding images.

**Key words:** Indexing of images, base of images, generic quadtree, clustering, K\_means algorithm

### INTRODUCTION

Classification is the organization of objects from other homogeneous or similar. In the literature, on distinguishes two families from machine learning: supervised and unsupervised clustering. In the approach supervised, we have objects which we know their classes of membership. We then seek classes to build a classification starting from these objects. In the approach unsupervised, we have no information on the objects to classify. In this article, we interest in a method of clustering called also K\_means. We integrate this approach in our content-based image retrieval system PicFind. This latter makes it possible to find the images of the base according to their characteristics of the low level of color and texture. From the point of view of indexing, each image is represented by a quadtree and the base of images is stored under a structure of data basis of the approach of versions of data bases, called also generic quadtree. This tree is built by sharing the common parts of the images, yielding minimum space of storage.

From the research point of view, we use the principle of similarity carried out by introduction of an image request given by the user to find out the similar images in the same class.

### ARCHITECTURE OF IMAGE RETRIEVAL SYSTEM

Research image for images similar to a query image consumes much time in a database image. To accelerate

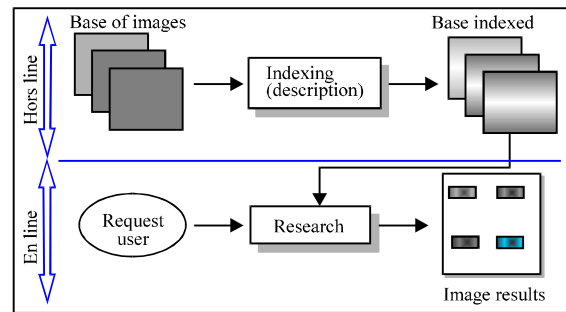


Fig. 1: Functioning of a content-based image retrieval system

the research process, several approaches have been developed (Huang and Lee, 2003).

Many image retrieval systems can be conceptually described by the framework depicted in Fig. 1. The architecture of a content-based image retrieval system is given in Fig. 1.

Two phases for interaction with the system are provided: A phase of indexing off line and a phase of research one line.

The indexing consists in calculating, starting from the image, a signature summarizing its contents and which will then be used for research. It is widely recognized that most current content-based image retrieval systems work with low level features (color, texture, shape, spatial).

Figure 1 schematizes functioning of a content-based image retrieval system.

## INDEXING

The goal of indexing is to provide a compact and significant representation of the image allowing for effective research.

A system of indexing generally includes two data processing runs.

**Logical indexing:** The logical indexing consists in extracting and modelling the characteristics of the image which are mainly the form, the colour and texture. Each one of these characteristics being able to be considered for a whole image or an area of the image.

**The color:** Color (Landre, 2005) is one of the most widely used visual features in content-based image retrieval. It is relatively robust and simple to represent. Various studies of color perception and color spaces have been proposed, in order to find color-based techniques that are more closely aligned with the ways that humans perceive. There exist several modes of representation of color. Among these models, one can quote:

**Red-green-blue model RGB:** Defined by international Company of lighting (CIE) in 1931. It represents the colors by subtractive synthesis. Each color is represented by three components: red (R), green (G) and blue (B) of which the standardized wavelengths are, respectively  $\lambda_R = 700$  nm,  $\lambda_G = 546,1$  nm et  $\lambda_B = 435,8$  nm.

**The XYZ model:** The CIE defined a space of representation which takes into account the sensitivity of eye. It is XYZ space, where X is the integral of a function of distribution of luminous power balanced on the whole of the visible spectrum, in the same way for Y and Z but with other weights.

**Hoots-saturation-been worth model HSV:** Space dyed, saturation and value have a component of brightness, color component corresponding to color and saturation.

**Texture:** Texture can be defined as a set of local neighbourhood properties of the gray levels of an image region. Texture analysis examines the small regular variations in intensity. It examines the intensity of a point with respect to its surrounding. Quantitatively, texture describes the two dimensional arrays of variations while the qualitative definition of texture is examines the fineness or coarseness of the image. Texture can be described by uniformity, density, coarseness, roughness, regularity, linearity, directionality, frequency and phase. Texture perception has many dimensions and the method of texture extraction depends on the application.

Various approaches (Jain and Tuceryan, 1992), syntactic, structural, or statistical are used to describe the texture. They have each on their applicability. The statistical methods, mainly based on the matrixes of co-occurrence were proposed by Haralick (1979). The idea of this method is to identify the repetition of gray levels of according to a distance and a direction.

In order to estimate the similarity between the matrices of co-occurrences, Haralick (1979) proposed 14 extracted statistical characteristics starting from a matrix. Currently, only the four most suitable characteristics are largely used:

- Energy =  $\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (P_{ij}(d, \theta))^2$
- Inerty =  $\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} ((i - j)^2 P_{ij}(d, \theta))$
- Entropy =  $-\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (\log P_{ij}(d, \theta)) P_{ij}(d, \theta)$
- Contrast =  $\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} ((i - j)^2 P_{ij}(d, \theta))$

**Physical indexing:** The physical indexing consists in determining an effective structure of access to the data to find information quickly. Many techniques based on the structure of tree (B-tree, R-tree, Quadtree...).

**Basic concepts of the quadtree (AQ):** The quadtree (Samet, 1990) is an effective data structure build, representing the images with two dimensions. An image is recursively divided into four equal-size quadrants until a stopping particular criterion is met, such as homogeneity of color or texture.

Each node of the quadtree represents a quadrant of the image. The root node of a quadtree represents the initial quadrant containing the whole image. If an image does not conform to the chosen split criterion, then the root has four descendant nodes representing the four first-level image quadrants.

A node is a leaf when its corresponding image quadrant is homogeneous. Otherwise the node is internal.

The numeral 0 identifies the initial quadrant representing the whole image. Numerals 0, 1, 2 or 3, following their parent node identifier 0, identify the four first-level quadrants. Recursively, sub-quadrants of an image quadrant n are identified by nx where x {0, 1, 2, 3}.

Figure 2 shows an example of images represented by quadtree.

**Generic quadtree (QGT):** A Generic Quadtree (Manouvrier, 2000) (QGT) may be considered as a data structure for representing and managing and storing similar images, on the basis of Database Version Approach. The QTG approach is based on a principle of sharing parts of image quad-trees. This structure

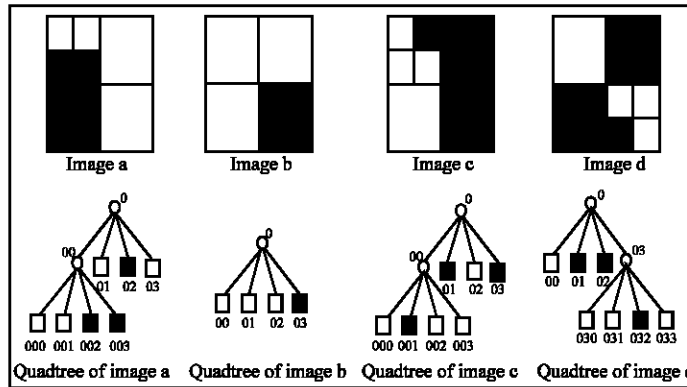


Fig. 2: Images representation by quadtree

minimizes the memory space of a set of images and speeds up several operations applied to image, such as insert new images, delete some of them, compare images, extract images, build image sequences, etc.

This structure is based on the following concepts: sharing part images of quadtrees, the similarity between the trees representing the images and the tree of images which organize the images.

**Sharing parts images of quadtrees:** The generic quadtree is based on two principles of sharing quadrant values between images: explicit and implicit. If a quadrant  $q$  has the same value in several images, this value is stored only once and is associated with the list of image identifiers. In this case, the sharing is called explicit, because the identifier of each image sharing the value is explicitly present in the list. The implicit sharing is based on the following rule: except if the identi of an image  $i$  is explicitly associated with another value  $v$ , image  $i$  shares the value with its parent image.

Let  $I_m$  be a set of images. If a quadrant  $q$  has the same value in a set of images  $I'_m \subset I_m$ , this value is stored only once in the base and is associated with the set of image identifiers of  $I'_m$ . In that case the sharing is called explicit.

If the set of images  $I_m$  are organized in tree structure, each image has a unique parent and an indefinite number of children, except the tree root. Thus, the following rule of implicit sharing may be introduced: except if the identifier of an image  $i$  is explicitly associated with another value  $v$ , image  $i$  shares the value of its parent image.

**Similarity between images:** To evaluate the sharing of common parts between two image quadtrees, a distance or a similarity measure between their quadtrees can be defined. This distance is based on several criteria such as: the structure of the quadtree and values of the nodes.

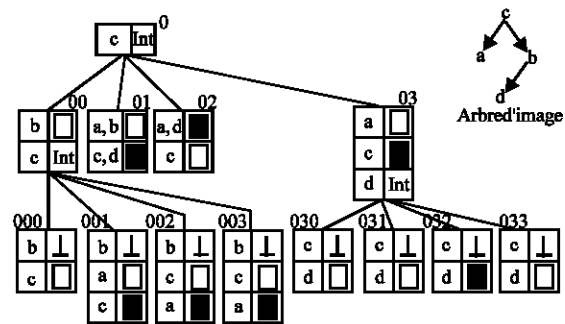


Fig. 3: Organisation of images of Fig. 2 by generic quadtree

The images are gathered, in the data base, according to similarity distance between the quadtree representing them. This distance is based on several criteria such as: the structure of the quadtree and values of the nodes.

**Image tree:** The images represented by a generic quad-tree are organized in particular structure called image tree. An image tree stores the ancestors and descendants of an image, like a version hierarchy. Using the image tree, the generic quad-tree allows an image to share common parts with its ancestors and descendants. An image  $j$  is inserted in the Image Tree as a child of an image  $i$  if the similarity between  $i$  and  $j$  has the smallest value in the database

**Example 1:** Figure 3 presents a Image tree organizing the images a, b, c and d. We suppose that the image c is original to which were apply the treatments from which the images b and d result.

**Generics nodes:** The Generic Quadtree is a quadtree-based structure whose nodes are called generic nodes. A

generic node  $n$  represents all nodes  $n$  of the quadtrees of images belonging to the base. It contains the whole information necessary to rebuild the value of the node with the same identifier  $n$  in each image quadtree. Each generic node may be seen as a table with two columns and one or several lines. Each line  $l$  of a generic node  $n$  contains a list of image identifiers and a value  $v$  of quadtree node:  $v$  is the value of node  $n$  in each image quadtree whose identifier  $i$  appears in line  $l$ .

**Example 2:** Figure 3 represents the Generic quadtree of the images of the Fig. 2; the generic node contains a single consecutive line Int: it means that node 0 are internal in the quadtree of all the images of the set.

### IMAGE RETRIEVAL

In Content-based image retrieval, there are several ways of measuring the resemblance between images in a base of images, with an image request. The definition of the similarity depends much on the manner with which the image is required.

We gave the general standard of the distance between images represented by quadtree. This distance, noted  $\Delta(i, j)$  (Oria *et al.*, 2004), allows to measure the similarity between images  $I$  and  $J$ .

**Definition of the distance between images:** To compare images represented by quadtree, the  $\Delta$  distance has been defined as follows:

$$\Delta(i, j) = \sum c_k \delta_k(i, j) / \sum c_k$$

The  $\Delta$  distance between two images  $I$  and  $J$  is defined by a sum of distance  $\delta_k(i, j)$  between the nodes of the quadtree representing images  $I$  and  $J$ , balanced by coefficients  $C_k$  with  $C_k \geq 0$ .

Where,  $\delta_k(i, j)$  is the normalized distance  $\in [0, 1]$  between the homologous nodes  $K$  of the quadtree  $I$  and  $J$ .  $k$  is the identifier of the nodes of the quadtree of images  $I$  and  $J$ .  $C_k$  is the coefficient of weight distance  $\Delta$  between homologous nodes  $n$  in the distance computation. Is a plus coefficient representing the weight of the node  $K$  in the calculation of distance. Each weight  $C_k$  is chose according to the importance which one wishes to give to certain quadrants of image compared to others in the calculation of the  $\Delta$  distance.

**Particular cases of distance  $\Delta$  distance:** Particular cases of  $\Delta$  distance can be found depending on the weights  $C_k$ .

**The T distance (T for Tree):** Allows the comparison of the structure of two quadtree representing the images, without taking into account the value of the nodes leaf.

**The distance  $\delta_k(i, j)$  takes only 2 values:** 0 when both nodes are interns or leaf. 1 when the node is leaf in a quadtree and intern in the other, or when the node  $K$  exists only in one tree.

**The Q distance (Q for quadrant):** The Q-similarity between two images is defined as the number of nodes having the same identifier but different values.

The distance  $\delta_k(i, j)$  takes value 0 when all the homologous nodes are all the two interns or all the two leaf with the same value; value 1 is given when the node is leaf in a quadtree and intern in the other or when the node  $K$  exists only in one tree and a value ranging between] 0, 1[when the two nodes are with the same position but their values are different.

**The V distance (V for visual):** During the calculation of distance  $\Delta$  between two images  $I$  and  $J$ , the quadtree of these images are supplemented to have the same structure. Its does not hold when whereas values of the nodes ( $\delta_k(i, j) = 0$  for all the internal nodes).

### CALCULATION OF SIMILARITY

Table 1 shows the various similarity T, Q and V from images represented in quadtree of Fig. 4.

If for example, we consider that two images are similar when their Q distance is lower than 0.3, then the result of the request corresponds to the images 1 and 2.

Table 1: Different similarity

	Image1	Image 2	Image 3	Image 4
T	0.00	0.00	0.38	0.38
Q	0.15	0.08	0.46	0.38
V	0.08	0.06	0.09	0.03

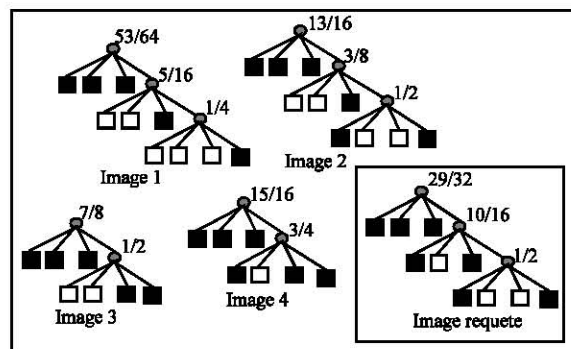


Fig. 4: The example of similarity calculation

**RESULTS**

Our system PicFind developed with C++ in 2007 in our laboratory Systems Signals Data Department of Computer Science Faculty of Science University of Science and Technology Mohamed Boudiaf, Oran. PicFind makes it possible to represent black and White images, level of gray or color, by quadtree whose criterion of cutting is the homogeneity of the color and texture. The base of images obtained is stored in the form of a generic quadtree.

The objective of our prototype PicFind is to improve the results of the search for images by the contents of our prototype REQUIT developed with C++ in 2005 in our laboratory. We conceived by improving the indexing part by the addition of a second descriptor.

**Phase off line:** It is made in two phases: The first phase made by the program is the segmentation that operation tries to establish a total partition of an image in regions. These regions will be relatively homogeneous from the point of view of the studied criteria.

**Extraction of the characteristic color:** In Picfind, every color is defined by three components: red (R), green (G) and blue (B) representing space RGB. The homogeneity of a region is determined by its variance of colors. The variance being calculated by the following formula:

$$\text{Variance}_k = \frac{\sum(\text{averagecolor}_k - \text{Value of color of pixel quadrant}(i, j))^2}{\text{No. of pixels of color } k \text{ of the quadrant} / \text{No. total pixels of the quadrant}}$$

Such as: K = red/green/blue.

**Extraction of the characteristic textures:** The second descriptor is the texture used as a mean to finalize the color segmentation. Its information is essential when making the distinction between two zones of an image of the same color. For every homogeneous region obtained from the color segmentation, 4 matrixes of cooccurrences will be calculated on regions in levels of grey for one distance =1 and for direction (0°, 45°, 90° and 135°) degrees. To estimate the similarity between the matrixes of cooccurrences, we used the first three parameters: the energy, the entropy and the slowness.

**Phase on line:** In this stage, the system will make a search called by the example or by the similarity. Databases image used for the tests in indexation are diverse and are chosen according to the criterion used for the search. Indeed, bases used for the indexation by the color include

most of the domains of application: the satellite images, photography, portraits etc., while indexing by the texture, present resemblances or differences at the level of the aspects of surfaces. However, we used certain bases, which became standard bases for the indexation. Three types of color similarity are defined to calculate the distance between images represented by quaternary trees: V-similarite, T-similarity and Q-similarite, while the distance used to calculate the similarity textures is the distance of Minkowski defined as follows:

$$L_r(i, j) = \frac{1}{m} \left( \sum_{d=1}^m |\text{param}_i(d) - \text{param}_j(d)|^2 \right)^{1/2}$$

where, i is an image request, j is an image of the base, m represent the total number of parameters.

The search for images by the contents is made by means of the request concerning the combination of both characteristics color and texture.

The global function of similarity used between the quadtree of both images is the is the sum of these two distances:

$$\text{Distance} = \text{weight} * \text{Dist}(\text{couleur}) + (1-\text{weight}) * \text{Dist}(\text{texture})$$

After several validation of our prototype Picfind, we noticed that there is a saving of space memory. This reduction is of the order of 40 and 50%.

The example 1 (Fig. 5) and example 2 (Fig. 6) following ones represent respectively the result of research for similarity by level and by whole image with a threshold of research that the chosen user.

**Example 1:** Table 2 shows the different distance T, Q and V of the example 1 (Fig. 4).

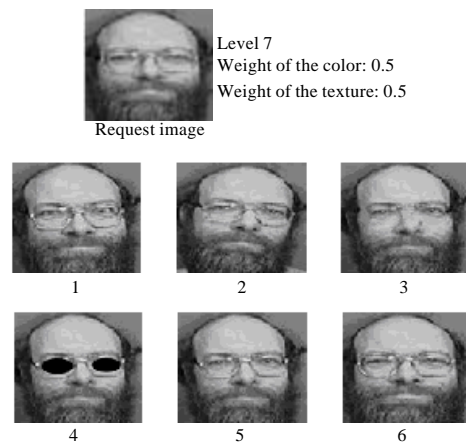


Fig. 5: Search of similarity by level

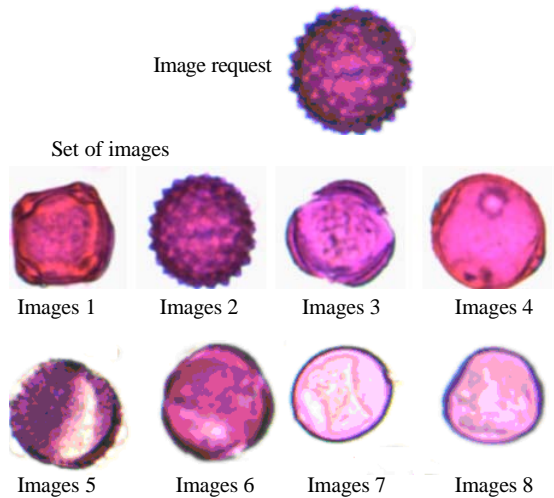


Fig. 6: Search similarity by image

Table 2: Search of similarity by level

Images	1	2	3	4	5	6
T	100	99.63	99.79	99.14	99.90	100
Q	44.39	42.81	/	/	/	100
V	/	/	/	/	/	100
V niv=1	/	/	/	/	/	100

Table 3: Search similarity by image

Images	Im 1	Im 2	Im 3	Im 4	Im 5	Im 6	Im 7	Im 8
T	99.60	100	99.59	99.62	99.24	99.35	98.35	99.43
Q	70.14	100	/	77.08	/	79.55	/	/
V	/	100	/	/	/	/	/	/

**Example 2:** Table 3 shows the distance T, Q and V between the different images of the exemple1 and request image (Fig. 4).

In short, these various distances (T, Q, V) are proposed in various optics. If we do research of images by the contents and if all the images have the same quadtree we shall use the distance V. If we do research of images by the contents with images segmented there quadtree, we shall use the distance Q if we are interested in the structure of the quadtree we shall use the distance T.

A good organization of a base of images requires a good strategy of indexing and storage in order to minimise the space of research.

The architecture, which rests on the storage of the images in a cluster and carries out research by sweeping the entire group, becomes obsolete with respect volumes of data.

Currently, the idea is to use architecture able to manage a set of images segmented in several subsets in order to reduce the space of research and improve the performances in terms of answer time. An organization in cluster will enable us to reduce the volume of the data to

be explored by carrying out research only in the cluster considered relevant. The images of the same to cluster must be the most similar possible to ensure the quality of the results. For it we thought of practitioner our base of images by using a technique of clustering.

In what follows we give an introduction on some unsupervised methods of classification then we explain the method used.

## CLASSIFICATION

Classification is one of the many fields in Data Mining (DM), which aims to extracting information from large data volumes. Unsupervised classification or clustering is one of the most important field in Data Mining and consists in grouping together similar items into the same groups while the dissimilar ones are asserted to different groups, without any prior knowledge on the obtained groups. Alternatively, when the prior knowledge on the data is present, the technique is called supervised classification. These groups of objects are usually called clusters or classes and are used to synthesize the information contained in the initial data.

Recently, Clustering has found wide spread of applications in a large variety of fields, archaeology, natural sciences, bioinformatics, psychology and economics. Therefore, clustering combines different techniques from different disciplines: statistics, computer science, mathematics, artificial intelligence, databases, etc. For instance, the continuous growth of the world wide web (www), has led to the emergence of new applications domains, such as the web Mining, that is: the application of DM techniques on Web data.

The two common approaches in statistical clustering are partitioning clustering and hierarchical clustering: In the partitioning clustering one starts with the whole initial analyzed data set, which is split into k clusters. Usually the number k has to be specified before the analysis. It influences the result of the clustering. There are also techniques to determine the most appropriate values of k, based usually on the selection of the optimal partition for a range of k values. The partitioning methods usually produce clusters by optimizing a criterion function and sometimes due to the combinatorial number of possibilities, the algorithm is run repeatedly. Example of used criterion in this case includes the squared error criterion, the diameter, the sum of distances, etc. Among the most known partitioning methods, we remind the K\_Means (Macqueen, 1967), PAM (Fisher, 1987), CLARA (Kaufman and Rousseeuw, 1990) and CLARANS (Ng and Han, 1994).

The hierarchical clustering methods can also be divided into agglomerative, divisive and incremental. The purpose of these types of classification is to produce a tree in which the nodes represent clusters of the initial analyzed set. In particular, the initial set is the root of the tree while the leaves represent the singletons (clusters with one element). Thus, this type of structure gives an enhanced visual representation than the partitioning methods. The investigator can then select the suitable partitioning from its point of view, by making a trade off between the number of clusters and their homogeneity degree.

In the incremental hierarchical methods, an already constructed classification of objects is augmented by successively inserting new objects into the classification. The interest of these kind of methods lie on their capability to analyze very large data sets.

Among the most known methods, we remind the BIRCH (Zhang and Ramakrishan, 1998), Cure (Guha *et al.*, 1998) and Chameleon (Karypis *et al.*, 1999).

In the hierarchical divisive methods, starting from a single cluster (of all objects), successive splits of clusters are performed to obtain smaller clusters.

In the agglomerative hierarchical methods, starting from the initial elements (the singletons), the clusters are successively merged into higher level clusters, until the entire set of analyzed objects becomes a cluster. These resulting hierarchical structures contain a number of partitions which can be then easily visualized using a graphic. By far, the most known and used hierarchical agglomerative method is the Agglomerative Hierarchical Classification. This method is also called the Ascending Hierarchical Classification.

### K\_MEANS METHODS

K\_means clustering (Macqueen, 1967) is a basic and effective algorithm. It splits a set of objects into a selected

number of groups. Every object is compared to the centroid value of each group and assigned to the group that has most similar properties with the object. The centroid value is updated by taking the average of all the objects that belong to the group. This process is repeated until there is no change from the previous round compared to the current one. For K-Means clustering, K is specified by the user.

In our prototype we have implemented the k\_means method. The procedure of the implementation is as the follow:

- The choice of K (K number of the classes)
- The choice of a nuclus of the class: take both first images of the base most different and consider them as the centres both premières classes
- For the creation of the ith class, we calculate the distance between the images of the base and the nuclei of ( I-1) classes already
- To take the minimum of these distances and compare it to the threshold of dissimilarity
- If the calculated distance is higher than the threshold to create a new class, no one assigns this image to the corresponding class

### Integration of the K\_means method in prototype FindPic:

In our work, we integrated method K\_means in the system findPic system. Once the groups of images are conceived. Each group of images will be the subject of a generic quadtree.

Figure 7 shows the integration of the method K\_means in our system.

**Validation of k\_means:** We validated our prototype Algo\_k\_means conceived in 2008 in our laboratory using different bases of images (satellite, medical, animals, landscapes, etc.) and of sizes various (60, 80 100,120) images while varying the number of class K. We give an

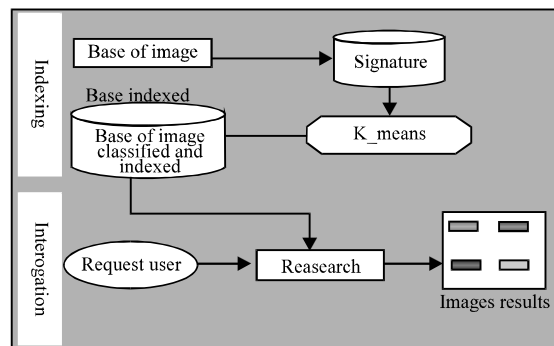


Fig. 7: Integration of the K\_means method in retriviel system FindPic





Fig. 8: Example of base of images

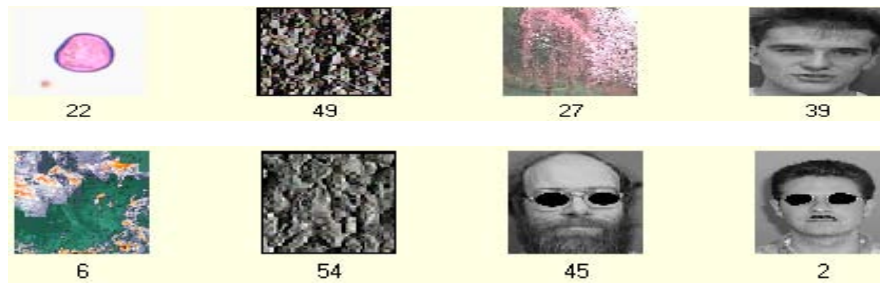


Fig. 9: The nucleuses of the classes

Table 4: Comparison of execution time

Base of images	Size (Mo)	No. of images	No. of class (k)	Time of classification(sec)
Base 1	17,2	120	2	146
			3	181
			5	189
			9	202
			10	207
			Base 2	9,4
5	56			
6	59			
7	61			
10	68			
Base 3	9,6	60		
			3	5,125
			4	5,375
			5	6,312
			6	6,94
			7	6,218
			8	7,469
			9	6,75
			10	7,46

Table 5: Sizes of each class

Size bases bmp (Mo)	Size bases QTG (Mo)	K: No. of class	Sum (Mo)
9,6	2,93	2	2,76
		3	2,73
		4	2,73
		5	2,7
		6	2,47
		7	2,42
		8	2,28

The nucleuses of the classes are shown in Fig. 9

Figure 10a-h shows the different class obtained. The images of the classes are:

Table 4 gives the execution time of various bases of images with different number of class k

We notice that when the images base becomes larger, the classification time increases.

We give in Table 5 the result of the global size of the image base after classification. We notice that the structure of the generic tree is protected

After some validation, we notice that the higher the number of classes, the less the over all size of their classes.

example (Fig. 8 of validation of base of images which contains 60 images with K = 8

The results of the classification with the number of classes equal to eight (K = 8) are the following ones:

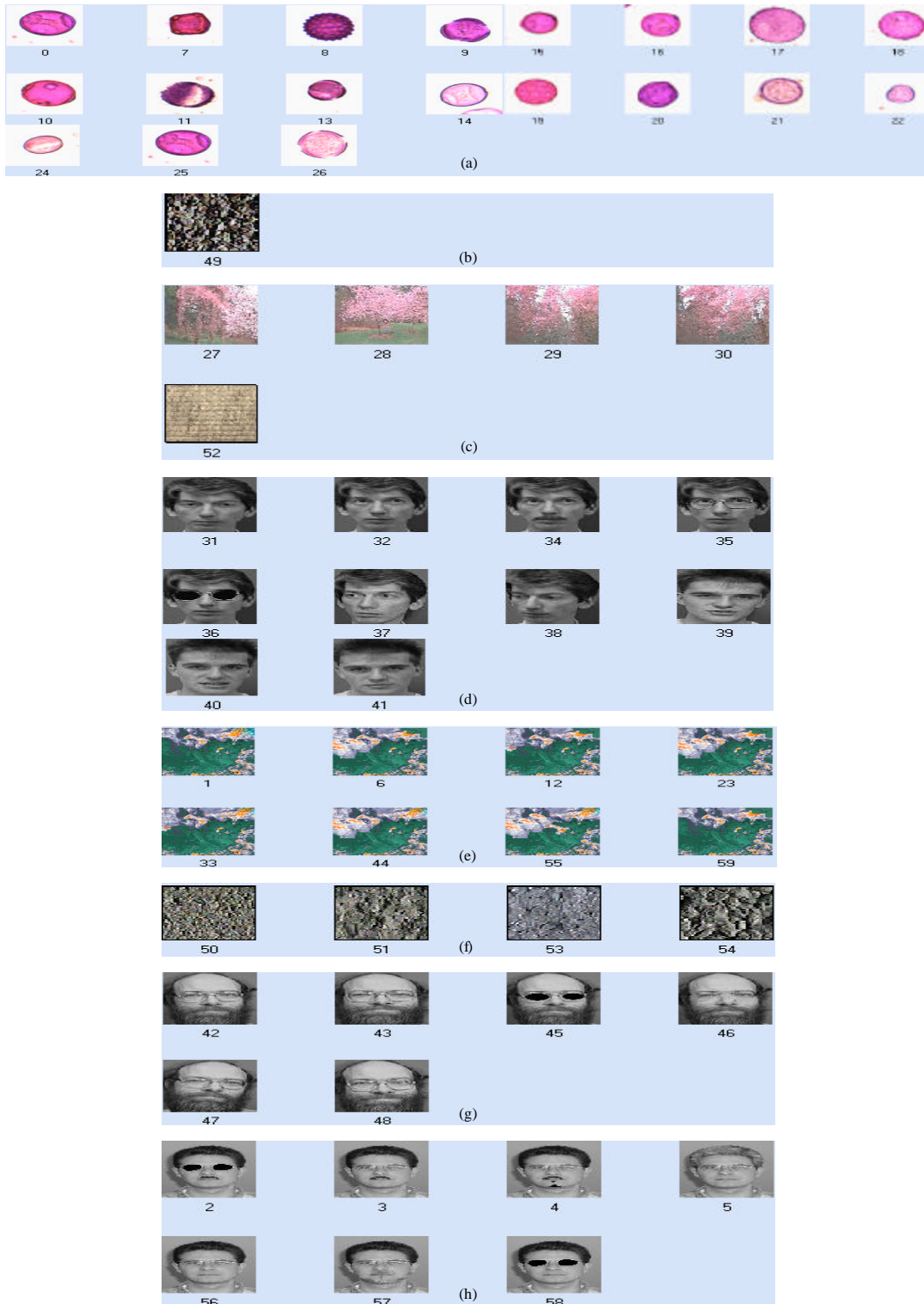


Fig. 10: Classes obtained after application of  $k$ -means ( $k = 8$ ). (a) classe 1, (b) classe 2, (c) classe 3, (d) classe 4, (e) classe 5, (f) classe 6, (g) classe 7 and (h) classe 8

## CONCLUSIONS

The classification and content-based retrieval in bases of image are from now a very active field of research.

We presented our system for indexing and research for images in which we partitioned a generalist base of image in similar groups by using the method of K\_means.

Our objective of is adapted this techniques of clustering to a particular data structure which is the generic quadtree. For that we associated with each image a quadtree and we stored each cluster in generic quadtree. This last minimizes the size of storage. The experiments carried out show that the organization it clusters, allows to preserve the structure of generic quadtree, but also to reduce the search time.

## REFERENCES

- Fisher, D.H., 1987. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2: 139-172.
- Guha, S., R. Rastogi and K. Shim, 1998. CURE: An efficient clustering algorithm for large databases. *SIGMOD Rec.*, 27: 73-84.
- Haralick, R.M., 1979. Statistical and structural approaches to texture. *Proc. IEEE.*, 67: 786-804.
- Huang, P.W. and C.H. Lee, 2003. An efficient method of organizing bit-string signatures for searching symbolic images. *Inform. Technol. J.*, 2: 159-172.
- Jain, A.K. and M. Tuceryan, 1992. Texture Analysis. In: *The Handbook of Pattern Recognition and Computer Vision*, Chapter 1.1, Chen, C.H., L.F. Pan and P.S.P. Wang (Eds.). World Scientific, Singapore.
- Karypis, G., E.H. Han and V. Kuner, 1999. CHAMELEON: A hierarchical clustering algorithm using dynamic modeling. *Computer*, 32: 68-75.
- Kaufman, L. and P.J. Rousseeuw, 1990. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley and Sons, New York.
- Landre, J., 2005. Multiresolution analysis for content-based image indexing and retrieval in images data bases. Doctoral Thesis, University of Burgundy.
- Macqueen, J.B., 1967. Some methods for classification and analysis of multivariate observations. *Proc. Berkeley Symp. Math. Statist. Prob.*, 1: 281-297.
- Manouvrier, M., 2000. Objets de grande taille dans les bases de donne es. Doctorat Thesis, Universite de Paris.
- Ng, R.T. and J. Han, 1994. Efficient and effective clustering methods for spatial data mining. *Proceedings of the 20th Conference on Very Large Data Bases*, Sept. 12-15, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA., pp: 144-155.
- Oria, V., G. Jomier, M. Manouvrier and M. Rukoz, 2004. Indexing multi-level global research and partial images by content. *Proceedings of the 20th Day Advanced Databases*, Oct. 19-22, Montpellier, France, pp: 177-196.
- Samet, H., 1990. *The Design and Analysis of Spatial Data Structures*. Addition Wesley, New York.
- Zhang, T. and R. Ramakrishan, 1998. BRICH: An efficient data clustering method for very large data bases. *Proceedings of the International Conference on Very Large Data Bases*, Aug. 24-27, ACM, New York, pp: 392-403.