http://www.zju.edu.cn/jzus; E-mail: jzus@zju.edu.cn
ISSN 1009-3095 *Journal of Zhejiang University SCIENCE* V.4, No.4, P.421-425, July-Aug., 2003

An image retrieval system based on fractal dimension

YAO Min(姚敏)[1†], YI Wen-sheng(易文晟)[1], SHEN Bin(沈斌)[1], DAI Hong-hua[2]
([1] *College of Computer Science, Zhejiang University, Hangzhou 310027, China*)
([2] *School of Information Technology, Deakin University, Melbourne VIC 3125, Australia*)
[†]E-mail: myao@zju.edu.cn

Received Aug.2, 2002; revision accepted Oct.30, 2002

Abstract: This paper presents a new kind of image retrieval system which obtains the feature vectors of images by estimating their fractal dimension; and at the same time establishes a tree-structure image database. After preprocessing and feature extracting, a given image is matched with the standard images in the image database using a hierarchical method of image indexing.

Key words: Fractal dimension, Image partition, Feature extraction, Image retrieval
Document code: A **CLC number:** TP391

INTRODUCTION

With the rapid development of the technology of image collection, storage, procession and display, digital images are widely used (Zhang *et al*., 2001). However, it is difficult to manage a huge image database, so the study of the image database management system and the image inquire and retrieval system has become a hot subject (Guo *et al*., 2001). Up to now, there are two main image retrieval methods: the text-based image retrieval and the content-based image retrieval. Text-based image retrieval is based on key words. In this kind of system, the content of an image is expressed as a set of fixed attributes (Cao *et al*., 2001). It is easy to be implemented. However, the more the attributes are abstracted, the more information the users need to input. Thus the inquiring efficiency will be low. The content-based image retrieval is based on the context of an image, including color, shape, grain and the object's special relationship (Niu *et al*., 1997; Miao *et al*., 1997). This method makes full use of the image information. However, it is limited by the development of correlated subjects, such as image procession, pattern recognition and computer visualization. Furthermore, it has higher complexity.

To overcome the limitation, a new kind of image retrieval system is proposed and constructed based on the surface fractal dimension of the images, assuming that all the images, including the images stored in database and the given image used to inquire, are static and of the same size.

CONSTRUCTION OF IMAGE RETRIEVAL SYSTEM

One of the current commonly used methods for color images in computer is based on the principle of three-color primaries: red, green and blue. This method usually describes the whole characteristics by using color vector only. So it cannot properly distinguish the regions with different features. Here, a new kind of image retrieval system (FDIRS for short) is constructed based on the fractal dimension estimated. The structure of FDIRS is shown in Fig.1.

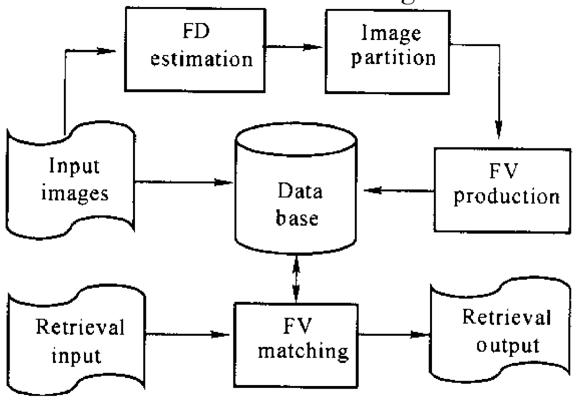

Fig.1 The structure of FDIRS

It is necessary to note that in this system model, before the fractal dimension (FD for short) is estimated, the images are divided into several types, for example, the plants, the animals, the people, the scenery, and so on. This classification can be carried out artificially or automatically. Images that belong to different kinds may be stored separately.

After computing the fractal dimension of regions for every pixel in an image, the computed fractal dimensions are used to partition the image by means of quad-tree and obtained comparatively exact segmentation and the feature vector (FV for short) is formed. The feature vectors and the corresponding partition quad-trees are stored in a feature database for mapping with the images in image database. At the same time, by clustering the feature vectors of images of different classes, the structure of a binary tree is generated. This binary tree and the quad-trees are used to match in different precision when retrieving.

Although the color information for images is included in the fractal model, it is possible to obtain the same fractal dimension from different regions with different color consistency. In order to overcome the above disadvantage, a color image is divided into three concolorous images: red, green and blue images. Then the fractal dimensions are estimated, and region segment and feature extraction are made for each concolorous image respectively. Finally, the feature vectors are combined into one and stored in the feature database. The same process is done on the queried images.

EXTRACTION OF FEATURE VECTORS

1. The estimation of fractal dimension

The shapes in nature are divided into two kinds. One kind, such as decks and basketballs, has characteristic size, which can be easily described by traditional Euclidean geometry. The other, such as the coastlines, mountains and clouds, has no characteristic size. These irregular shapes are studied by fractal geometry (Kaukoranta *et al*., 2000). Fractal sets generally have the following characteristics:

(1) Fractals possess accurate structure, i.e. they contain the integer at any small scale.

(2) Fractals are self-similar and independent of scale or scaling.

(3) Fractals, in general, are the result of a simple construction procedure or algorithm that is often recursive.

(4) Fractals are irregular and are not easily described by traditional geometric language.

Fractals could be quantified by fractal dimension, which could be calculated by the model of fractional Brownian motion (fBm for short). The fractional Brownian motion is defined as a random procession X with a real variable t (time) and a parameter a, $0 < a < 1$,

$$X: [0, \infty] \rightarrow R$$

It satisfies:

(a) With probability being 1, $X(t)$ is continuous and $X(0) = 0$.

(b) For any $t \geqslant 0$ and $h > 0$, the increments of X obeys normal distribution with mean value zero and square error h^{2a}, i.e.

$$P(X(x+h) - X(t) \leqslant x) = (2\pi)^{-1/2} h^{-a} \int_{-\infty}^{x} \exp\left(\frac{-u^2}{2h^{2a}}\right) du$$

The Hausdorff dimension or box dimension of fBm is $D = 2 - a$. The power spectral density (PSD) $P[f(x, y)]$ is proportional to $1/f^{\beta}_{(x,y)}$ with $\beta = 2a + 1$.

$$P[f(x, y)] = cf_{(x,y)}^{-(2\alpha+1)}$$

where c is a constant.

Based on the above conditions, the fractal dimensions of uniform regions can be computed for every pixel by the following steps.

(1) Select a window with size of $n \times n$ from the image (n may be 16 initially). The target pixel is included in the window. If the target pixel locates at the edge of the image, the window is selected along the opposite direction. Otherwise, the target pixel is at the center of the window.

(2) Do Fourier Transformation and obtain the Fourier PSD:

$$P_i = \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} P[f(x, y)]$$

(3) Add up the value of $f(x, y)$ for each pixel in the window

$$F_i = \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} f(x, y)$$

(4) Change the size of the window. Let $n = n - 1$, go to step (1) until $n = 4$; otherwise continue.

(5) According to the property $P[f(x, y)] = cf_{(x,y)}^{-(2\alpha+1)}$, calculate the logarithm

$$\log[P(f(x, y))] = (2a + 1)\log[f(x, y)] + c$$

(6) Calculate $\log(F_i)$ corresponding to its slope α by the means of the linear regression method. Then a can be computed by the formula $-(2a + 1) = \alpha$, and the fractal dimension D is calculated through $D = 2 - a$.

This method for estimating fractal dimension is pixel-oriented. Therefore, it escapes from the problem that the fractal dimension is computed inexactly because one window covers several regions with different features.

2. Image segmentation based on fractal dimension

After the fractal dimensions for all pixels are computed, the image may be segmented by the variance of the fractal dimensions in an image. A quad-tree structure is used here.

Before constructing the quad-tree, divide the value of the fractal dimension into eleven levels, i.e. d_1: [1, 1.05]; d_2: [1.05, 1.15]; …; d_{11}: [1.95, 2]. A quad-tree is constructed by the level that the fractal dimension of a pixel belongs to. The recursive algorithm for constructing the quad-tree is as follows.

Step 1. Generate a new node p. The initial node represents the whole image.

Step 2. Set node p to its fractal dimension and block code.

Step 3. If all pixels in the block represented by node p are at the same fractal dimension level, the node p is as signed a corresponding location code and area $S(p)$, then finish.

Step 4. Otherwise, divide the block into four equal sub-blocks. For each sub-block, repeat Step 1 to Step 4.

The position encoding rules of quad-tree are as follows.

(1) Let the size of the image be $2^N \times 2^N$, then the code of any block is N bits.

(2) For four sub-blocks having the same parent, set them in left to right and top to bottom order, encoding as 0, 1, 2, 3 from high bit to low bit.

(3) If a block has no sub-block and its location code has k bits, let the left $N - k$ bits be 4.

The advantage of this encoding method is that the information on blocks can be easily obtained. If a leaf node code has k's equal to 4, the corresponding block's area is $2^k \times 2^k$. What is more, if its position code is arranged in increasing order, the result is exactly the same as the result of depth-first searching. By adding some node information, this kind of position code table can be considered as a kind of high dimension feature vector.

3. Feature vectors production

To begin with, calculating the area of each fractal dimension level, searching the quad-tree and accumulating the areas of all leaf nodes at the same fractal dimension level.

$$S(d_i) = \sum_{j=0}^{n} S(P_{d_i})$$

Where $d_i = \{d_1, d_2, \cdots, d_{11}\}$ is the set of the fractal dimension levels.

Then, in an image, three largest areas are selected, and combined with their fractal dimension levels. These three pairs of areas and fractal dimension levels comprise the feature vectors of a concolorous image. For example, the feature vector of a red image is

$$(S_{R1}, d_{R1}, S_{R2}, d_{R2}, S_{R3}, d_{R3})$$

The feature vector of a color image is the combination of the feature vectors of its three concolorous images, i.e.

$$(S_{R1}, d_{R1}, S_{R2}, d_{R2}, S_{R3}, d_{R3}, S_{G1}, d_{G1}, S_{G2}, d_{G2}, S_{G3}, d_{G4}, S_{B1}, d_{B1}, S_{B2}, d_{B2}, S_{B3}, d_{B3})$$

This is the very feature vector of a given image.

FEATURE VECTOR CLUSTERING

Clustering is often considered as a set of techniques that

1. are all dependent on some basic choices.
2. cover different problems, generally formulable, in terms of criterion to be optimized.

Here the clustering criterion is the minimum distance criterion. The following clustering algorithm is proposed for the feature vectors.

(1) Let there be N pattern samples, each of which comprises a class; that is, there are N classes: $G_1^0, G_2^0, \cdots, G_n^0$. Compute the distances

between any two classes using the Euclidean distance formula

$$D(P_q, P_t) = \{\sum_{R,G,B}\sum_{i=0}^{3}[(P_q(S_i) - P_t(S_i))^2 + (P_q(d_i) - P_t(d_i))^2]\}^{1/2}$$

where P_q and P_t are two images, and S_i, d_i are the ith weight in their feature vectors. Thus a distance matrix D^0 of $N \times N$ dimensions is obtained.

(2) If a distance matrix D^n is computed at last step, find its minimum element. Supposing that this element is the distance between class G_i^n and class G_j^n, put them into one class G_{ij}^{n+1}. The new class G_{ij}^{n+1} and other unchanged classes comprise a new classification: G_1^{n+1}, G_2^{n+1}, $\cdots$.

(3) If there is only one class, the end. Otherwise, continue.

(4) Compute the distances between any two classes in the new classification, and construct a new distance matrix D^{n+1}. Go to step 2.

In this way, the feature vectors are clustered step by step. The number of classes varies from N to 1. At last, a binary tree storage structure is formed.

The advantage of this algorithm is that the information of the original database is fully used. When some images are added or deleted, the only task is to adjust some nodes in the binary tree, while most nodes stay unchanged.

INDEX OF THE SYSTEM

A hierarchical structure is adopted as the indexing structure of the FDIRS. At different hierarchies, different index methods are used.

At the highest level, the traditional index method of keywords is used to find which feature class the target image belongs to.

At the feature class level, the indexing method is changed into content-based method. Search starts from the root of the binary tree, and go down to the node with the nearest distance to the target image. The similarity criterion and distance function are the same as those used in constructing the binary tree. The match is rough granularity at low-dimension end.

When matching a node, which includes images comprising about one-third the total number of images in the database, the indexing method is altered again. The accurate match is carried out with high-dimension. The quad-trees are generated when segmenting is used during matching of the querying image with the images in the database. Since the result of depth-first searching for quad-trees is the same as the result of arranging the location code by increasing order, we can compare the target image with the location code table of image regions, and try to fine the number of regions in querying images I_q which are at the same level as the corresponding regions in an image I_t in the database. This number is the foundation of the similarity of two images. The matching result is displayed according to the degrees of similarity from large to small. The formula of similarity measurement is

$$D(I_q, I_t) = \sum_{R,G,B}\sum_{i}(w_i * A_i)$$

where w_i is the weight, and A_i is the characteristic function determining whether the fractal dimension of the ith region in I_q and I_t is equal or not.

$$A_i = \begin{cases} 1 & d(I_q(P_i)) = d(I_t(P_i)) \\ 0 & \text{other} \end{cases}$$

By this hierarchical index method, images can be retrieved quickly, reliably and precisely. At high level, it is easy to classify images using abstract concepts. Therefore, the keyword index method does work. Because at lower level, it is difficult to match images by abstract concepts, content-based index is adopted and a binary tree is used. However, the distance between two classes becomes smaller and smaller when searching down in a binary-tree, so the accuracy of the similarity decision decrease, even result in wrong conclusions. Thus the index method has to be changed for accurate matching by using quad-trees.

CONCLUSIONS

In this paper, the fractal concept is used to represent images' feature, and an image retrieval system is designed and constructed based on fractal analysis of images.

(1) By computing the fractal dimensions for every pixel in images, the problem of inaccurate computation could be avoided when the window is selected across different regions. On the other hand, it provides the basis for image segmentation and accurate matching.

(2) By means of the diversity of fractal dimension with different regions' features, precise region segmentation and feature vectors can be obtained.

(3) The hierarchical index structure designed in this paper enables rough to meticulous indexing. At the same time, the position information of regions included in high dimension vectors is used when matching, so that it favors increase of not only the accuracy of querying but also the efficiency.

It seems that this image retrieval system introduced new ideals, which would enhance both theory and the methodology of image retrieval to a new advanced stage. We expect this new system will have a good perspective in real world applications.

References

Cao, K., Feng, Y. and Wang, Y., 2001. Efficient color image retrieval using color and spatial feature. Image Compression and Encryption Technologies. *Proceedings of SPIE*, **4551**: 177 – 182.

Guo, L., Yang, J. and Sun, X., 2001 Trademark Image Retrieval. *In*: Tian *et al*. Eds Image Compression and Encryption Technologies. *Proceedings of SPIE*, **4551**: 160 – 164.

Kaukoranta, T., Franti, P. and Nevalainen, O., 2000. A fast exact GLA based on code vector activity detection. *IEEE Transaction Image Processing*, **9**: 1337 – 1342.

Miao, Y., Sun, Z. Q., Zhu, W. B. and Ma, X. J., 1997. Content Based Object Oriented Video Index and Query. Proceedings of the 1997 IEEE International Conference on Intelligent Processing Systems. International Academic Publishers/World Publishing Corporation, Beijing, p. 1705 – 1709.

Niu, Y. P., Ozsu, M. T. and Li, X. B., 1997. 2D-h Trees: An Index Scheme for Content-Based Retrieval of Imges in multimedia Systems. Proceedings of the 1997 IEEE International Conference on Intelligent Processing Systems. International Academic Publishers/World Publishing Corporation, Beijing, p. 1710 – 1714.

Zhang, L., Ke, X. and Shu, H., 2001. Image Retrieval by Using Sub-piece Accumulative Histogram. *In*: Tian *et al*. Eds, Image Compression and Encryption Technologies. *Proceedings of SPIE*, **4551**: 149 – 153.