2010

Content based radiology image retrieval using a fuzzy rule based scalable composite descriptor

Chatzichristofis, Savvas A.

Noname manuscript No.
(will be inserted by the editor)

CONTENT BASED RADIOLOGY IMAGE RETRIEVAL USING A FUZZY RULE BASED SCALABLE COMPOSITE DESCRIPTOR

Savvas A. Chatzichristofis · Yiannis S. Boutalis

Received: date / Accepted: date

Abstract The rapid advances made in the field of radiology, the increased frequency in which oncological diseases appear, as well as the demand for regular medical checks, led to the creation of a large database of radiology images in every hospital or medical center. There is now an imperative need to create an effective method for the indexing and retrieval of these images. This paper proposes a new method of content based radiology medical image retrieval. The description of images relies on a Fuzzy Rule Based Compact Composite Descriptor (CCD), which includes global image features capturing both brightness and texture characteristics in a 1D Histogram. Furthermore, the proposed descriptor includes the spatial distribution of the information it describes. The most important feature of the proposed descriptor is that its size adapts according to the storage capabilities of the application that uses it. Experiments carried out on a large group of images show that even at 48 bytes per image, the proposed descriptor demonstrates a high level of accuracy in its results. To evaluate the performance of the proposed feature, the mean average precision was used.

Keywords CBMIR · Image Retrieval · Fuzzy Methods · Medical Images

1 Introduction

The last decades have witnessed significant advances in medical imaging and computerized medical image processing. The commonly used medical imaging modalities for radiological applications are: X-ray Computed Tomography (X-ray CT), Magnetic Resonance Imaging (MRI), Single Photon Emission Computed Tomography (SPECT),

Savvas A. Chatzichristofis
Department of Electrical & Computer Engineering, Democritus University of Thrace, Xanthi, Greece
E-mail: schatzic@ee.duth.gr

Yiannis S. Boutalis
Department of Electrical & Computer Engineering, Democritus University of Thrace, Xanthi, Greece,
Department of Electrical, Electronic and Communication Engineering, Chair of Automatic Control, University of Erlangen-Nuremberg, Germany
E-mail: ybout@ee.duth.gr

Positron Emission Tomography (PET) and Ultrasound [1].
Nowadays, digitized medical images are becoming more frequently used. This leads to the creation of large image databases, resulting in the need to find an efficient method for the indexing and retrieval of these images.
The process of automatic indexing and retrieval of medical images based on their content is known by the term Content-based medical image retrieval (CBMIR) [2]. Content-Based Image Retrieval (CBIR) is any technology that in principle helps to organize digital image archives by their visual content. By this definition, anything ranging from an image similarity function to a robust image annotation engine falls under the purview of CBIR [3].
CBMIR is quite different from CBIR as the retrieval similarity must consider the medical context (such as the subtle pathological changes) as well as the user individualized subjectivity [4]. Medical images are multimodal, heterogeneous and higher dimensional with temporal properties, which distinguish them from images in other domains [5]. In the CBMIR the basic objective is to provide diagnostic support to the physicians or radiologists by displaying relevant past cases, along with proven pathologies as ground truth [6].
Several medical image indexing and retrieval techniques have been proposed in the literature. Chu et al presented an image retrieval system dedicated to brain MRI which indexes images mainly on the shape of the ventricular region [7]. Korn et al proposed a system for the fast and effective retrieval of tumour shapes in mammogram X-rays [8]. Comaniciu et al described a system that aims to help physicians in the diagnosis of lymphoproliferative disorders of blood [9]. Glatard et al introduced a Texture Based Medical Image Indexing and Retrieval for Cardiac Images [10].
Most of the CBMIRs are based on visual example. The doctor/specialist uses a radiology image as input (query image), and, based on certain global or local feature vectors (FV), the system brings up similar images. These sorts of feature vectors are used to describe the content of the image and that is why they must be appropriately selected on occasion. The visual content of the images is mapped in to a new space called the feature space. The feature vectors (descriptors) that are chosen have to be discriminative and sufficient for the description of the objects [11].
This paper proposes a new descriptor that can be used for the indexing and retrieval of radiology medical images. This descriptor uses brightness and texture characteristics as well as the spatial distribution of these characteristics in one compact 1D vector. The most important characteristic of the proposed descriptor is that its size adapts according to the storage capabilities of the application that is using it. This characteristic renders the descriptor appropriate for use in large medical (or gray scale) image databases. Our earlier research papers, in which brightness and texture features were combined in one compact descriptor [12], demonstrate the effectiveness of the composite descriptors.
To extract the proposed descriptor, a two unit fuzzy system is used. To extract the brightness information, a fuzzy unit classifies the brightness values of the image's pixels into L_{Bright} clusters. The cluster centers are calculated using the Gustafson Kessel Fuzzy Classifier [13], which is described in section 2, while the process of fuzzy brightness classification is described in section 3.
The texture information embodied in the proposed descriptor comes from the Directionality histogram suggested in [14]. This feature is part of the well known Tamura texture features. The extraction process of the Fuzzy Directionality Histogram is described in section 3.

Section 4 describes the Fractal Scanning method, which is used to capture the spatial distribution of brightness and texture information. Section 5 describes the process of combining the brightness and texture features for the formation of the proposed descriptor. The experimental results are described in section 6, while the conclusions are presented in section 7.

2 Gustafson Kessel Fuzzy Classifier

One major problem of the standard fuzzy C-mean algorithm is that it produces spherical classes. For example, if the sets of points illustrated at Fig. 1(a) pass through the fuzzy C-mean algorithm for partitioning into four classes, the result will not be the optimal (Fig 1(b)). If we replace the Euclidean distance in the fuzzy C-mean algorithm with another metric, the calculation of which will include a positive symmetrical table, it will allow ellipsoid clusters to be recognized, as well as spherical ones. Such an algorithm has been designed by Gustafson and Kessel (GK) [13]. The Gustafson Kessel algorithm is used in this paper in the brightness information extraction process, which is described in section 3, as well as in the quantization process of the proposed descriptor, in the methodology analyzed in section 5.

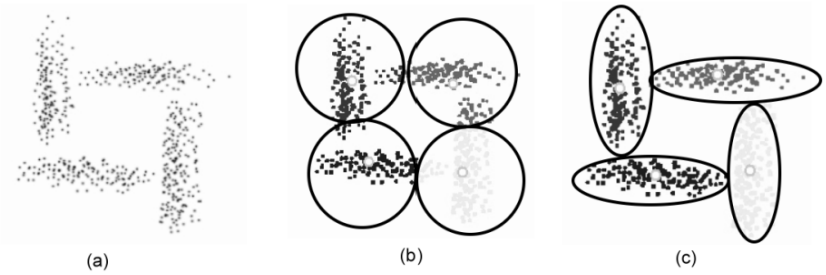


Fig. 1 (a) The points in the 2D space which must separate to four classes. (b) Four classes obtained by using the fuzzy C-mean algorithm. (c) The four classes obtained through the Gustafson - Kessel algorithm.

Let the total of prototypes $X = \{x_1, x_2, \ldots, x_n\}$ with $X_i \in R^p$, which we want to classify into L clusters. If $\{v_1, v_2, \ldots, v_L\}$ are the vectors of the cluster centers then Mahalanobis distance of every prototype x_k of the cluster (v_i, A_i) is equal to:

$$d_{ik}^2 = (x_k - v_i)^T \times A_i \times (x_k - v_i) \tag{1}$$

Where $A_i = C_i^{-1}$, with C_i the covariance matrix of the cluster i given by:

$$C_i = \sum_{k=1}^{n} u_{ik}^m (x_k - v_i) \times (x_k - v_i)^T \tag{2}$$

Where u_{ik}^m is described in equation 6. In every repetition of the GK algorithm the vectors of the center v_i of the clusters are determined as:

$$v_i = \frac{1}{\sum_{k=1}^{n} u_{ik}^m} \sum_{k=1}^{n} u_{ik}^m x_k \quad i = 1, 2, \ldots, n \quad k = 1, 2, \ldots, n \tag{3}$$

The GK algorithm is consist of the following steps:

Step 1: The number of L clusters and the largest number of repetitions are determined.

Step 2: The U^0 table of participatory functions is started, either at random or based on a particular approach. The centers of V^0 clusters and the covariance matrixes C^0 are calculated. Then, the tables A^0 of the clusters are calculated. Next, the U^0 tables are recalculated. A value is set for m. Indicator $a = 0$.

Step 3: Given table U^a the centers of V^a clusters are calculated according to equation 3.

Step 4: Given the V^a, the covariance matrixes C^a of every cluster are calculated:

$$C_i = \sum_{k=1}^{n} u_{ik}^m (x_k - v_i)(x_k - v_i)^T \quad i = 1, 2, \ldots, L \quad k = 1, 2, \ldots, n \tag{4}$$

Step 5: Given the covariance matrixes C^a, the A^a of every cluster are calculated

$$A_i = \sqrt[p]{\det(C_i)}(C_i)^{-1} \quad i = 1, 2, \ldots, L \tag{5}$$

Step 6: Given the A^a tables, the distance of every prototype x_k from the center of cluster is calculated according to the equation $d_{ik}^2 = (x_k - v_i)^T A_i (x_k - v_i)$. Next, the new participatory functions U^a of every prototype in every cluster are calculated.

$$u_{ik} = \frac{\left[\frac{1}{d_{ik}^2}\right]^{\frac{1}{m-1}}}{\sum_{j=1}^{L} \left[\frac{1}{d_{ik}^2}\right]^{\frac{1}{m-1}}} \quad i = 1, 2, \ldots, L \quad k = 1, 2, \ldots, n \tag{6}$$

Step 7: The process is completed if $\left| u_{ik}^{(a)} - u_{ik}^{(a-1)} \right| \leq e$ or if $a = Repetitions$. Otherwise make $a = a + 1$ and the process is repeated from step 3.

Steps 1 and 2 are executed only once, during the algorithm initiation, while steps 3, 4, 5 and 6 are repeatedly executed until at least one of the conditions described in step 7 is satisfied. The parameters calculated in steps 3, 4 and 5 are used in the repetition during which they are calculated while parameter U^a, calculated in step 6, is used both to check that the condition for algorithm termination is satisfied, and as data for the next repetition.

3 Brightness Information

The image-acquisition system usually converts a biomedical signal or radiation carrying the information of interest to a digital image [1]. In most cases, these images are 8 bit greyscale images. The first unit of the proposed system undertakes to fuzzy classify the brightness of the image pixels into L_{Bright} preset clusters. Experiments, which are described in section 6, were carried out for $L_{Bright} = 8$ and $L_{Bright} = 16$. The number of clusters comes as a compromise between the low storage requirements of the application using the proposed descriptor, and the need for more efficient retrieval accuracy. The center of these clusters was calculated using the following method: A sample of 3000 (8 bit grayscale) radiology medical images was used. These images included 1000 X-ray images, 1000 MRI images and 1000 Ultrasound images. All the images were resized to 128×128 pixels, ignoring the initial aspect ratio. The bicubic method was used to resize the images [15].
The aim was to classify the values of pixel brightness of all the images by using the Fuzzy Classifier Gustafson-Kessel in L_{Bright} clusters. The particularly large resulting figure of samples $3000 \times 128 \times 128 \cong 10^6$, however, would lead to out of memory problems. The Gustafson-Kessel algorithm was therefore used in each image individually, classifying the pixel brightness into 20 clusters. The Gustafson Kessel parameters are selected as: Clusters $L = 20$, Repetitions=3000, $e = 0.001$ and $m = 2$. Next, the group of 3000×20 cluster values were classified into L_{Bright} clusters using a second Gustafson-Kessel classifier. In this case, the Gustafson Kessel parameters are selected as: Clusters $L = 8, 16$, Repetitions= 2000, $e = 0.001$ and $m = 2$. The centers of the resulting clusters for $L = 8$ and $L = 16$ are shown in Table 1.

Table 1 Brightness Cluster Centers.

	v(0)	v(1)	v(2)	v(3)	v(4)	v(5)	v(6)	v(7)
L=16	1.71	12.25	22.53	35.38	50.38	65.60	82.41	99.99
L=8	3.18	22.68	54.00	90.13	125.80	162.57	202.25	243.64
	v(8)	v(9)	v(10)	v(11)	v(12)	v(13)	v(14)	v(15)
L=16	116.92	134.31	153.65	173.36	193.70	214.88	234.91	251.23

The resulting cluster centers $v(i), i \in [0, L-1]$ are used to form a fuzzy system. This system will classify the pixel brightness of any image, interacted with the system into L_{Bright} classes. The fuzzy system with $L_{Bright} = 8$ is illustrated in Figure 2.
The fuzzy system output is an L_{Bright}-bin histogram. The way this system operates is described as follows: Let the image j. Every pixel of the image interacts with the system. Suppose that the brightness value of the pixel $P(x, y)$ is P_b, where P_b:

$$v(k) \leq P_b \leq v(k+1) \quad k \in [0, L_{Bright} - 2] \tag{7}$$

This value activates the membership function k by $R(A)$ and membership function $k+1$ by $R(B)$, as illustrated in Figure 2. Where A and B are the incision points of P_b with the membership function k and $k+1$ respectively. The fuzzy system output histogram alternates as follows:

$$BrightnessHisto(k) = BrightnessHisto(k) + R(A)$$

$$BrightnessHisto(k+1) = BrightnessHisto(k+1) + R(B)$$

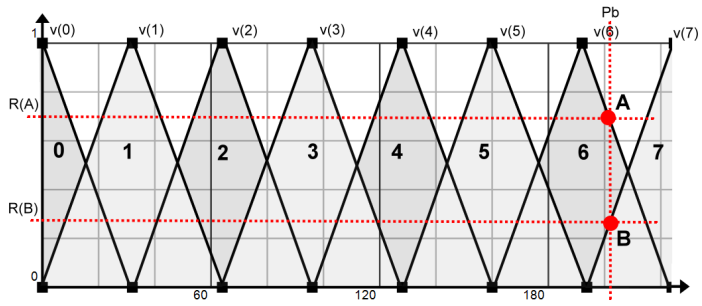


Fig. 2 Brightness Classification System.

$$R(A) + R(B) = 1$$

Where $BrightnessHisto$ is the fuzzy system output histogram.

For the example of Figure 2: $k = 6$,$R(A) = 0.7$ and $R(B) = 0.3$. The procedure is repeated for all the pixels.

4 Texture Information

Most CBIMR systems use texture features. Due to the structure of medical images, texture is a very strong feature which includes huge amounts of information. A commonly used texture feature is the Tamura Texture feature [16][17][18]. According to [17], which compares low-level features for the automatic categorization of medical images, the histograms based on Tamura's texture features yielded the best results among the features proposed for general-purpose image retrieval. The Tamura texture includes 6 features selected by psychological experiments: *Coarseness*, *contrast*, *directionality*, *line likeness*, *regularity* and *roughness* [14].
Tamura Directionality histogram is a graph of local edge probabilities against their directional angle.
In this paper, a novel fuzzy approach of the directionality histogram is proposed and used to describe the texture information to the proposed descriptor.
The extraction method of the traditional Tamura Directionality histogram utilizes the fact that gradient is a vector, so it has both magnitude and direction. In the discrete case, the magnitude $|\Delta G|$ and the local edge direction θ are approximated as follows:

$$|\Delta G| = \frac{|\Delta_H| + |\Delta_V|}{2} \tag{8}$$

$$\theta = tan^{-1}\left(\frac{\Delta_V}{\Delta_H}\right) + \frac{\pi}{2} \tag{9}$$

Where $|\Delta_H|$ and $|\Delta_V|$ are the horizontal and vertical differences computed using approximate pixel-wise derivatives measured by the Sobel edge detector in the 3×3 moving window.

The resultant θ is a real number $(0 < \theta < \pi)$ measured counter clockwise so that

the horizontal direction is zero. The desired histogram H_D can be obtained by quantizing θ in n values and counting the points with the magnitude $|\Delta G|$ over the threshold t;

$$H_D(k) = \frac{N_\theta(k)}{\sum_{i=0}^{n-1} N_\theta(i)} \qquad k = 0, 1, \ldots n-1 \tag{10}$$

Where $N_\theta(k)$ is the number of points at which $|\Delta G| \geq t$.

Thresholding by t is aimed at preventing counting of unreliable directions which cannot be regarded as edge points. The most classic approach, uses $n = 16$ and $t = 12$. Note that the shape of each histogram was not sensitive to the value of t.
The texture information extraction unit of the system that is used for the extraction of the proposed descriptor uses a fuzzy approach to extract the directionality histogram. In our approach we used $n = L_{Textures}$ quantized values of θ. The number of $L_{Texture}$ depends as much on the storage capabilities of the application using the proposed descriptor as it does on the need for more efficient retrieval accuracy. As the number of directionality areas increases, so does the calculating cost of extracting the proposed descriptor. Experiments were carried out for and $L_{Texture} = 4$ and $L_{Texture} = 16$ are described in section 6.
The $L_{Texture}$ quantized values of θ are used to form a fuzzy system, similar to the system that was described in section 2. The fuzzy system with $L_{Texture} = 16$ is illustrated in Figure 3.
For every image entered into the texture information extractor unit, an $L_{Texture}$-bin histogram that describes the directionality of the image is extracted.

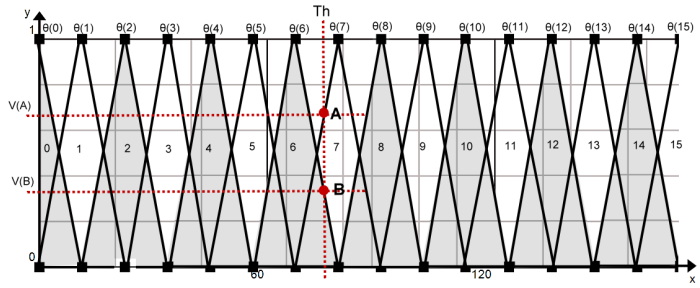


Fig. 3 Directionality Classification System.

The way this system operates is described as follows: Suppose that the θ value of a moving window (3×3 pixels) is Th, where Th :

$$\theta(k-1) \leq Th \leq \theta(k) \tag{11}$$

This value activates membership function k by $V(A)$ and the membership function $k+1$ by $V(B)$, as illustrated in Figure 3. The fuzzy system output histogram alternates as follows:

$$DirectionHisto(k) = DirectionHisto(k) + V(A)$$

$$DirectionHisto(k+1) = DirectionHisto(k+1) + V(B)$$
$$V(A) + V(B) = 1$$

Where $DirectionHisto$ is the fuzzy system output histogram.
For the example of Figure 3: $k = 7$, $V(A) = 0.7$ and $V(B) = 0.3$. The procedure is repeated for all moving windows (Entire image).

5 Fractal scanning

Fractal scanning is the term given to the image scanning technique based on a predetermined fractal curve. This technique is much more efficient for capturing spatial features in digital images than other commonly used techniques such as horizontal, vertical or diagonal pixels scan [19]. Such techniques do not maintain the adjacency of the features inside the image since their scanning direction is usually not identified with the one that features specify. Feature extraction is more substantial when we succeed to retain the neighborhood relationship among the pixels.

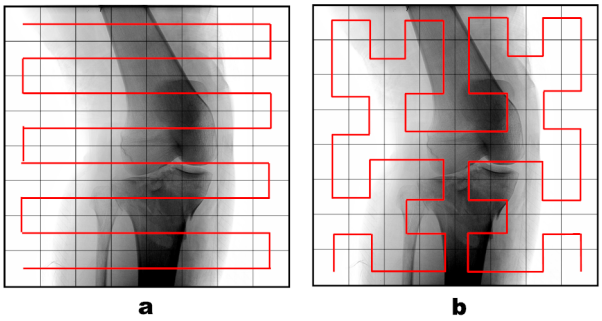


Fig. 4 Comparison between horizontal and fractal scanning.(a) Horizontal scanning, (b) Fractal Scanning

A comparison between the fractal scanning technique and the linear horizontal scanning is shown in Fig. 4. It is obvious that horizontal scanning loses the adjacency in the vertical direction and this will be also reflected on the constructed signature.

5.1 Hilbert Curve Fractal scanning

To apply fractal scanning to an image, space filling curves [20] are used. The Hilbert curve which is used in this paper is essentially a fractal that, in its two-dimensional form, can be analyzed as an L-System (Lindenmayer system) [21]. Using these systems, repetitive drawing methods that use the turtle analogy rule can be described. They are created started from one axiom, such as for example a rectilinear section and one or more drawing rules. When these rules are repeated several times the result is usually a complex fractal curve. We make this clear with the following example:

Consider the symbol “F” and the rule “replace symbol F with the string F+F-F+F”. By repeating this procedure we get:

```
F
F+F-F+F
(F+F-F+F) + ( F+F-F+F) - (F+F-F+F) + (F+F-F+F)
...
```

After this, we proceed with the drawing part. This is the stage where the turtle analogy comes in. We assume that “F” says, “draw a line” (move forward), “+” says “turn left” and “-” says “turn right”. Implementing the whole process by using x iterations of the rule, line length y and 60 degrees angle we get the well-known Koch curve. We characterize the curve with the factors x and y which reflects the order and the movement step, respectively. The Hilbert space-filling curve, which we use for fractal scanning, is constructed in the same way but using different rules and symbols:

```
Symbols : ‘‘L’’, ‘‘R’’
Rules : ‘‘L -> + RF - LFL - FR + ’’,‘‘R-> -  LF  + RFR + FL - ’’
Angle : 90 degrees
```

The letters “L” and “R” are used to create the series of symbols, the final shape of which will relate to the levels of repetition chosen. The drawing will take place taking into consideration only the letters “F” (drawing the line), “+” (clockwise rotation) and “-” (anti-clockwise rotation). Figure 5 shows the first three levels of repetition of the drawing rule. A first-order Hilbert curve is a square with one open side (which is the basic element) that defines its direction. The second-order Hilbert curve replaces every square by four other in a way that depends on the direction of the first-order square [19]. The reasons we adopt the Hilbert curve are its ability to capture in its path the adjacent information of the image objects, its property to cover squared regions and that it has been successfully used in the scanning process in other image-processing applications.

5.2 Z-Grid Based scanning

Replacing the Hilbert Curve with the Z-Space Filling Curve, the Z-Grid fractal scanning [22] method results. The extraction method can be modeled as follows:
Taking a grid G of dimensions $W \times Q$, where $W \in \{0, w-1\}$ and $Q \in \{0, q-1\}$. Two marker labels are placed to the top and to the left of the grid, to be used for mapping the grid cells. The symbols L_W and L_Q are used as label descriptors. The marker labels are separated into W and Q positions respectively. Each position corresponds to a cell. Odd position values for each marker label are equal to 1, while even position values are equal to 0. The cell $G(w, q)$ of Grid G is described using a combination of the $L_Q(q)$ and $L_W(w)$.
In the simplest case, where the grid size is equal to 2×2, $G(0, 0)$ as shown in the Fig. 5 (ii)(a) is marked as $L_Q(0)L_W(0) = 00$, $G(0, 1)$ is marked as $L_Q(1)L_W(0) = 10$, $G(1, 0)$ is marked as $L_Q(0)L_W(1) = 01$ and $G(1, 1)$ is marked as $L_Q(1)L_W(1) = 11$. To draw the Z-Grid fractal curve, starting from the upper left corner of the Grid $G(0, 0)$ move to the next cell, according to the order in which the cells were marked.
In higher dimensions, the original Grid is separated into a new Grid G'. G' is surrounding G. The dimensions of G' are $W' \in \{0, \frac{w-1}{2}\}$ and $Q' \in \{0, \frac{q-1}{2}\}$. Given

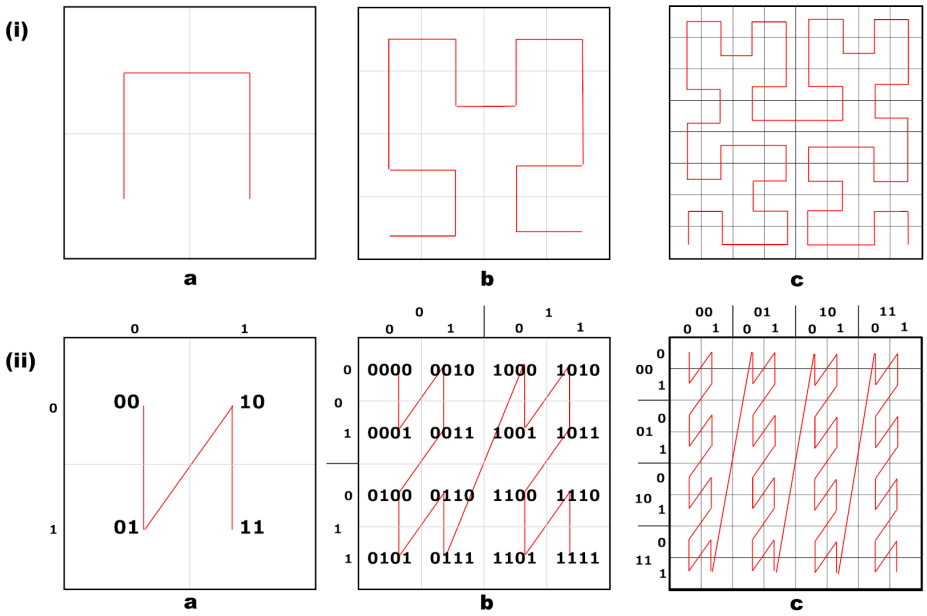


Fig. 5 Method of creating the (i) Hilbert and the (ii) Z-Grid based curve. (i)(a) First repetition of the drawing rule, (i)(b) Second repetition of the drawing rule and (i)(c) Third repetition of the drawing rule. (ii)(a) Designing of the Z-Grid Curve on a 2×2 grid, (ii)(b) Designing of the Z-Grid Curve on a 4×4 grid and (ii)(c) Designing of the Z-Grid Curve on a 8×8 grid

that G' has half the cells of G, each cell in G' could be considered to comprise one 2×2 Sub Grid. Two new marker labels are used to describe the cells in Grid G'. The symbols L'_Q and L'_W are used to describe the labels. Marker labels are respectively separated into $Q/2$ and $W/2$ positions. The procedure for label calibration is as follows: Starting from the top left corner of the Grid, $G(0,0)$, each cell is described by the binary number of the sequence in which it appears. As many bits as necessary are used to describe the binary number which describes the total number of cells resulting in each dimension. Cell $G(w',q')$ on Grid G' is described using the $L'_Q(q')$ and $L'_W(q')$. Finally, each cell $G(w,q)$ on Grid G is described by the combination: $L'_Q(q')L'_W(w')L_Q(q)L_W(w)$. This combination configures a binary number which specifies the order in which Z-Grid fractal scanning is configured. For example, $G(0,0)$ in the Fig. 5 (ii)(b) is marked as $L'_Q(0)L'_W(0)L_Q(0)L_W(0) = 0000$ while $G(1,1)$ is marked as $L'_Q(0)L'_W(0)L_Q(1)L_W(1) = 0011$. To draw the Z-Grid fractal curve in higher dimensions, starting from the upper left corner of the Grid $G(0,0)$ move to the next cell, according to the order in which the cells were marked.
For grid sized 2×2, 4×4 and 8×8, the procedure is described in Fig. 5 (ii).
The way in which the Hilbert curve and the Z-Grid fractal scanning are used in the proposed descriptor is described in section 6.

6 Descriptor Implementation

The brightness and texture features described in the previous sections are combined to produce the proposed descriptor. The production process is illustrated in Figure 6. Using the bicubic method, the image entered into the system is resized to the dimensions of 240×240 pixels, ignoring the initial aspect ratio. Our approach starts by performing brightness and edge enhancement in the *Pre-Filtering* unit. First, the auto brightness correction method proposed in [23] is applied to the image. This method is partially inspired by the HVS (Human Vision System). It particularly adopts some of the shunting characteristics of the on-center off-surround networks, in order to define the response function for a new artificial center-surround network. This network compares every pixel to its local average and assigns a new value in order to light the dark image regions, while minimally affecting the light ones. The aim of using this filter is to cover alterations in the brightness that might result from the settings of the system used to record/capture the medical image.

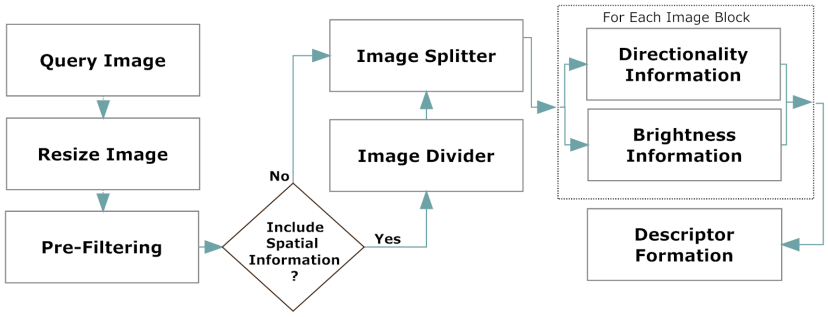


Fig. 6 Descriptor Implementation.

Next, a coordinate logic filter (CLF) -**OR**- [24] is applied to the image. This filter enhances the edges of the image and aims to help the texture information extraction unit to reach weaker texture alternations. The result of the image enhancement is illustrated in Figure 7.
In the following unit, system checks whether the user chose to integrate spatial information into the descriptor. The process of extraction of the proposed descriptor without spatial information is described in section 6.1 while the integration of spatial information is described in section 6.2.

6.1 Descriptor Implementation without Spatial Information

The improved image from the pre-filtering unit is transferred to the *Image Splitter* unit. In this unit, the image is divided into 3×3 pixels image blocks. Each image block is entered into the brightness information extractor unit (*Brightness Unit*) and the directionality information extractor unit (*Directionality Unit*). The combination of these two units forms the proposed descriptor which is constructed as follows: The descriptor's structure has $L_{Texture}$ regions determined by the Directionality Unit. Each

Fig. 7 (a) Original Image, (b) Auto Brightness Correction and (c) Application with CLF -OR- Filter. Image is taken from Auckland MRI Research Group.

Directionality Unit region contains L_{Bright} individual regions defined by the Brightness Unit. Overall, the proposed descriptor histogram contains $L_{Texture} \times L_{Bright}$ bins. Each Image Block interacts successively with the two units. Firstly, the *Directionality Unit* defines the texture type that is presented in the image block and classifies it in one or more regions $n \in [0, L_{Texture} - 1]$ with participant rate $P(n)$, for each n that participates.

$$\sum P(n) = 1$$

Meanwhile, every pixel of the image block interacts with the brightness classification unit and is classified in one or more of the $m \in [0, L_{Bright} - 1]$ preset brightness classes with participation rate $R(m)$, for each m that participates.

$$\sum R(m) = 1$$

Finally the output bin $(L_{Bright} \times n + m)$ of the proposed descriptor increases by $P(n) \times R(m)$ for each n and m. The procedure is repeated in the Brightness Unit for all the pixels.
On the completion of the process, the descriptor's histogram bin values are normalized within the interval $[0, 1]$.

$$bin(i)^{'} = \frac{bin(i)}{\sum_{j=0}^{L_{Texture} \times L_{Bright}} bin(j)} \qquad i \in [0, L_{Texture} \times L_{Bright}] \tag{12}$$

The structure of the proposed descriptor is illustrated in Figure 8.

In order to reduce the storage needs of the proposed descriptor, its bin values are quantized for binary representation using a three bits/bin quantization. Given that the $L_{Bright} \times L_{Texture}$ Directionality/ Brightness bin values are concentrated within a small range (from 0 to 0.053), linear quantization cannot be applied. The descriptor's histogram bins are separated into $L_{Texture}$ quantization groups (one group for each texture type). Each group is quantized with different quantization table.
In order to calculate the quantization table of each group, the same set of images described in section 3 was used. First, the descriptor of all the images was calculated.
Let X_j be the proposed descriptor of the image j. X_j is separated into the $L_{Texture}$ regions according to the quantization groups. Let $\left[X_j^i\right]$ the set of bin values in the group

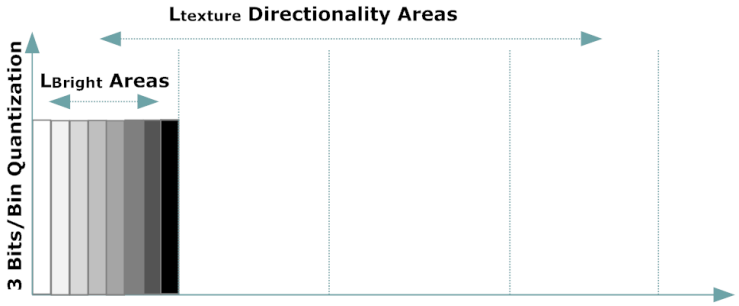


Fig. 8 Proposed Descriptor Structure.

$i \in [0, L_{Texture} - 1]$ of the image j. For each i (group), the set of bin values $\left[X_j^i\right]$ of all the j (images), constitute inputs into the fuzzy Gustafson Kessel classifier, which separates the bin values into eight regions. This technique maps the bin values into the integer area $[0, 7]$. The procedure is repeated for all the $L_{Texture}$ groups. The Gustafson Kessel parameters are selected as: Clusters $L = 8$, repetitions=2000, $e = 0.002$, and $m = 2$. This quantization method was also used in [25] and [26]. The resulting quantization tables for $(L_{Texture} = 8, L_{Bright} = 8)$, $(L_{Texture} = 8, L_{Bright} = 16)$ and $(L_{Texture} = 16, L_{Bright} = 8)$ are given in Table 2, Table 3 and Table 4 respectively. Each row represents a quantization group. The final size of the proposed descriptor is:

$$\left((L_{Texture} \times L_{Bright}) \times \frac{3}{8}\right) bytes \tag{13}$$

For each image entered into the system, the proposed descriptor is extracted. This descriptor is separated into $L_{Texture}$ regions according to the quantization groups. The value of each bin of the descriptor is assigned to one of the values $[0, 7]$ according to the minimum distance of the value from one of the eight entries in the corresponding row of the proper quantization table. For example, let the proposed descriptor histogram $bin(4) = 0.006$ for $L_{Texture} = 16$ and $L_{Bright} = 8$. This particular bin belongs to group 1 (the first Directionality/ Brightness group), the quantization table of which is shown in the first row of Table 4. The minimum distance is with the value $7.2E - 03$. Therefore the proposed descriptor histogram quantized value of $bin(4) = 011 = \mathbf{3}$.

Table 2 Quantization Table for $L_{Texture} = L_{Bright} = 8$.

000	001	010	011	100	101	110	111
2.3E-04	1.8E-03	4.0E-03	6.8E-03	1.1E-02	1.7E-02	4.0E-02	5.7E-01
2.7E-04	1.9E-03	4.0E-03	6.5E-03	9.8E-03	1.5E-02	2.7E-02	8.3E-02
2.1E-04	1.4E-03	3.1E-03	5.1E-03	7.6E-03	1.1E-02	1.8E-02	4.5E-02
2.4E-04	1.6E-03	3.5E-03	5.7E-03	8.6E-03	1.3E-02	2.1E-02	6.2E-02
2.4E-04	1.7E-03	3.9E-03	6.7E-03	1.0E-02	1.7E-02	3.9E-02	3.6E-01
2.4E-04	1.7E-03	3.9E-03	6.7E-03	1.0E-02	1.7E-02	3.9E-02	4.6E-01
2.4E-04	1.8E-03	4.0E-03	6.9E-03	1.1E-02	1.8E-02	4.0E-02	3.9E-01
2.2E-04	1.7E-03	3.8E-03	6.5E-03	1.0E-02	1.7E-02	3.6E-02	4.0E-01

Table 3 Quantization Table for $L_{Texture} = 8$ and $L_{Bright} = 16$.

000	001	010	011	100	101	110	111
8.0E-04	4.6E-03	8.8E-03	1.4E-02	2.0E-02	3.0E-02	5.5E-02	4.0E-01
8.4E-04	5.1E-03	9.9E-03	1.5E-02	2.2E-02	3.3E-02	6.5E-02	5.9E-01
7.8E-04	4.5E-03	8.5E-03	1.4E-02	2.0E-02	2.8E-02	5.5E-02	5.6E-01
8.3E-04	5.0E-03	9.5E-03	1.4E-02	2.0E-02	3.0E-02	5.4E-02	2.9E-01
7.2E-04	4.1E-03	7.8E-03	1.2E-02	1.8E-02	2.7E-02	5.1E-02	3.6E-01
6.8E-04	3.9E-03	7.7E-03	1.2E-02	1.8E-02	2.6E-02	4.8E-02	3.1E-01
6.9E-04	4.2E-03	7.8E-03	1.3E-02	1.9E-02	2.8E-02	5.0E-02	3.1E-01
8.9E-04	5.2E-03	1.0E-02	1.6E-02	2.3E-02	3.3E-02	6.2E-02	3.4E-01

Table 4 Quantization Table for $L_{Texture} = 16$ and $L_{Bright} = 8$.

000	001	010	011	100	101	110	111
3.6E-04	2.3E-03	4.5E-03	7.2E-03	1.1E-02	1.6E-02	3.1E-02	5.7E-01
3.2E-04	1.9E-03	3.7E-03	5.7E-03	8.4E-03	1.3E-02	2.4E-02	5.3E-01
3.3E-04	1.9E-03	3.5E-03	5.3E-03	7.6E-03	1.1E-02	1.6E-02	4.0E-02
3.7E-04	2.1E-03	4.1E-03	6.3E-03	8.8E-03	1.2E-02	2.0E-02	6.9E-02
3.2E-04	1.8E-03	3.4E-03	5.4E-03	7.9E-03	1.1E-02	1.7E-02	3.9E-02
3.1E-04	1.7E-03	3.3E-03	5.3E-03	7.8E-03	1.1E-02	1.8E-02	5.6E-02
3.6E-04	2.0E-03	3.9E-03	6.1E-03	8.6E-03	1.3E-02	2.1E-02	7.8E-02
3.6E-04	2.1E-03	4.0E-03	6.2E-03	9.0E-03	1.3E-02	2.4E-02	1.9E-01
3.7E-04	2.3E-03	4.4E-03	7.0E-03	1.0E-02	1.6E-02	2.9E-02	2.4E-01
4.2E-04	2.5E-03	4.9E-03	7.6E-03	1.1E-02	1.9E-02	5.3E-02	6.8E-01
3.6E-04	2.1E-03	3.9E-03	6.3E-03	9.3E-03	1.4E-02	2.7E-02	4.4E-01
3.7E-04	2.2E-03	4.3E-03	6.9E-03	1.1E-02	1.6E-02	3.5E-02	3.8E-01
3.3E-04	2.0E-03	3.8E-03	6.2E-03	9.4E-03	1.4E-02	2.8E-02	3.7E-01
2.7E-04	1.8E-03	3.6E-03	5.7E-03	8.9E-03	1.3E-02	2.6E-02	3.6E-01
3.0E-04	2.0E-03	3.9E-03	6.1E-03	8.8E-03	1.3E-02	2.6E-02	3.7E-01
3.2E-04	2.1E-03	4.2E-03	6.8E-03	1.0E-02	1.5E-02	3.1E-02	4.3E-01

6.2 Descriptor Implementation with Spatial Information

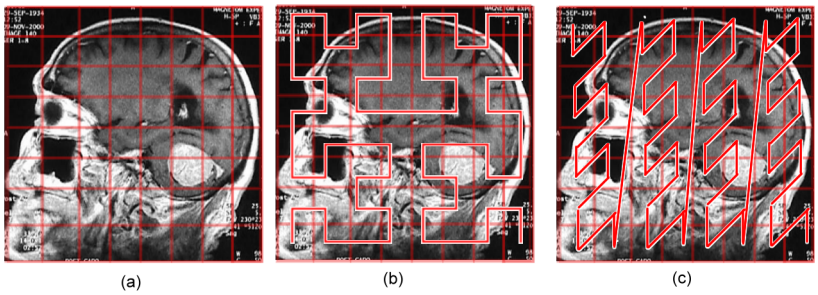
To integrate the spatial information in relation to the distribution of Brightness and Texture, the improved image from the pre-filtering unit interacts with the *Image Divider* unit.

In this unit, the image is divided into $L_{Blocks} \times L_{Blocks}$ squared sections. These sections are called *Sub Images*. Experiments were carried out for $L_{Blocks} = 4$ and $L_{Blocks} = 8$. These are described in section 7.

Each *Sub Image* is transferred to the image splitter unit. The sequence in which the Sub Images are entered into the next unit relates to the fractal scanning method, as Figure 9 illustrates.

Fig. 9(b) is a schematic illustration of how the Hilbert curve is applied for the fractal scanning of Sub Images while Fig. 9(c) is a schematic illustration of how the Z-Grid is applied for the fractal scanning of Sub Images. Fig. 9(e) numerically illustrates the order in which the Sub Images are entered into the *image splitter* in case of Hilbert fractal scanning and Fig. 9(f) numerically illustrates the order in which the Sub Images are entered into the *image splitter* in case of Z-Grid scanning. The fractal scanning of the image plays a fundamental part in the formation of the descriptor's final shape. In the image splitter unit, every Sub-Image is treated as an independent image and the proposed descriptor is extracted using the process described. The completion of the process results in $L_{Blocks} \times L_{Blocks}$ Sub-Image descriptors.

The resulting $L_{Blocks} \times L_{Blocks}$ vectors are combined for the formation of the final descriptor shape. The Sub-Images descriptors are placed in succession, one next to the



1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

(d)

22	23	26	27	38	39	42	43
21	24	25	28	37	40	41	44
20	19	30	29	36	35	46	45
17	18	31	32	33	34	47	48
16	13	12	11	54	53	52	49
15	14	9	10	55	56	51	50
2	3	8	7	58	57	62	63
1	4	5	6	59	60	61	64

(e)

1	3	17	19	33	35	49	51
2	4	18	20	34	36	50	52
5	7	21	23	37	39	53	55
6	8	22	24	38	40	54	56
9	11	25	27	41	43	57	59
10	12	26	28	42	44	58	60
13	15	29	31	45	47	61	63
14	16	30	32	46	48	62	64

(f)

Fig. 9 Linear and Fractal Scanning. (a) Original image. (b) The image is divided into 64 ($L_{Blocks} = 8$) *Sub Images* using Hilbert Fractal. For $L_{Blocks} = 8$ the third repetition of the drawing rule is used. (c) The image is divided into 64 ($L_{Blocks} = 8$) *Sub Images* using Z-Grid. (d) Numerical representation of the Sub Images using Linear Scanning.(e) Numerical representation of the Sub Images using Hilbert Fractal and (f) Numerical representation of the Sub Images using Z-Grid.

other according to the order determined by the fractal scanning method. The final length of the descriptor is:

$$L^2_{Blocks} \times (L_{Bright} \times L_{Texture}) \tag{14}$$

If spatial distribution with $L_{Blocks} = 8$ is integrated in the proposed descriptor, the quantization tables are given in Table 6 and Table 7. If the $L_{Blocks} = 4$, the quantization tables is given in Table 5.
Each Sub-Image descriptor is quantized using a different set of quantization values. The same quantization table is used for both the Hilbert fractal and the Z-Grid. The first column in the tables corresponds to the Sub-Image resulting from Linear Scanning (Fig. 9(d)). Depending on the method of fractal scanning implemented, the position of a given Sub-Image is defined in relation to the position of the Sub-Image in the case of Linear Scanning, and the corresponding row in the table is selected. For example: if the Hilbert fractal scanning method with $L_{Blocks} = 8$ is applied to the image, the quantization of the histogram produced by the 22^{nd} Sub-Image is achieved using the 1^{st} line on Table 6. If the Z-Grid is used, the quantization of the histogram of the 22^{nd} Sub-Image would be achieved with the 27^{th} row of Table 6. The same quantization process is followed when using $L_{Blocks} = 4$. In this case, Table 5 is used.
The final size of the descriptor is:

$$\left(L^2_{Blocks} \times (L_{Bright} \times L_{Texture}) \times \frac{3}{8}\right) bytes \tag{15}$$

Table 5 Quantization Table for $L_{Block} = 4, L_{Texture} = 16$ and $L_{Bright} = 8$.

Line	000	001	010	011	100	101	110	111
1	5.4E-06	1.2E-03	4.5E-03	9.4E-03	1.7E-02	2.7E-02	5.2E-02	4.1E-01
2	3.4E-06	9.1E-04	3.4E-03	7.2E-03	1.2E-02	2.2E-02	3.1E-02	6.8E-02
3	3.8E-06	8.6E-04	3.2E-03	6.8E-03	1.2E-02	1.9E-02	2.9E-02	6.0E-02
4	4.7E-06	1.5E-03	4.7E-03	9.8E-03	1.7E-02	2.8E-02	5.0E-02	1.3E-01
5	1.5E-06	8.2E-04	3.5E-03	7.4E-03	1.3E-02	2.1E-02	3.3E-02	7.5E-02
6	5.0E-07	1.2E-03	4.6E-03	9.8E-03	1.7E-02	2.9E-02	5.1E-02	1.3E-01
7	4.6E-06	1.2E-03	4.5E-03	9.7E-03	1.7E-02	2.2E-02	5.2E-02	4.4E-01
8	5.7E-06	1.5E-03	4.2E-03	9.0E-03	1.6E-02	2.8E-02	5.2E-02	4.9E-01
9	4.8E-06	1.2E-03	4.5E-03	9.6E-03	2.2E-02	2.8E-02	5.3E-02	3.2E-01
10	5.3E-06	1.2E-03	4.3E-03	9.6E-03	1.9E-02	3.3E-02	5.3E-02	1.9E-01
11	3.0E-06	1.1E-03	4.3E-03	9.1E-03	1.6E-02	2.5E-02	4.1E-02	1.1E-01
12	2.7E-06	1.2E-03	4.7E-03	8.0E-03	1.8E-02	3.0E-02	5.4E-02	3.3E-01
13	3.4E-06	1.7E-03	4.2E-03	9.0E-03	1.6E-02	2.8E-02	5.5E-02	1.8E-01
14	2.9E-06	8.3E-04	3.5E-03	7.5E-03	1.3E-02	2.1E-02	3.3E-02	7.4E-02
15	1.8E-06	9.6E-04	3.8E-03	6.1E-03	1.4E-02	2.2E-02	3.5E-02	8.6E-02
16	2.8E-06	1.1E-03	4.4E-03	9.5E-03	1.7E-02	3.0E-02	5.7E-02	4.1E-01

Table 6 Quantization Table for $L_{Block} = 8, L_{Texture} = 16$ and $L_{Bright} = 8$ Part 1.

Line	000	001	010	011	100	101	110	111
1	5.5E-06	1.2E-03	4.5E-03	9.6E-03	1.7E-02	2.8E-02	5.1E-02	4.2E-01
2	3.6E-06	8.8E-04	3.4E-03	7.1E-03	1.2E-02	2.0E-02	3.2E-02	7.6E-02
3	3.6E-06	8.3E-04	3.2E-03	6.8E-03	1.2E-02	1.9E-02	2.9E-02	5.5E-02
4	3.6E-06	8.2E-04	3.1E-03	6.6E-03	1.1E-02	1.8E-02	2.8E-02	5.2E-02
5	5.4E-06	1.1E-03	4.4E-03	9.5E-03	1.7E-02	2.9E-02	5.4E-02	4.3E-01
6	5.0E-06	1.1E-03	4.4E-03	9.6E-03	1.8E-02	2.9E-02	5.3E-02	4.0E-01
7	5.0E-06	1.1E-03	4.2E-03	9.0E-03	1.6E-02	2.7E-02	5.2E-02	4.6E-01
8	4.5E-06	1.2E-03	4.5E-03	9.6E-03	1.7E-02	2.8E-02	5.3E-02	4.8E-01
9	5.2E-06	1.2E-03	4.2E-03	9.0E-03	1.6E-02	2.8E-02	5.2E-02	4.9E-01
10	3.5E-06	8.6E-04	3.3E-03	6.8E-03	1.2E-02	1.9E-02	2.9E-02	6.1E-02
11	3.8E-06	8.6E-04	3.2E-03	6.8E-03	1.2E-02	1.9E-02	2.9E-02	6.0E-02
12	4.0E-06	8.5E-04	3.2E-03	6.8E-03	1.2E-02	1.8E-02	2.8E-02	5.3E-02
13	5.4E-06	1.1E-03	4.3E-03	9.3E-03	1.7E-02	2.8E-02	5.1E-02	3.9E-01
14	5.2E-06	1.1E-03	4.4E-03	9.5E-03	1.7E-02	2.8E-02	5.1E-02	3.5E-01
15	5.7E-06	1.2E-03	4.3E-03	9.0E-03	1.6E-02	2.7E-02	5.1E-02	4.0E-01
16	4.7E-06	1.2E-03	4.4E-03	9.4E-03	1.7E-02	2.8E-02	5.2E-02	4.2E-01
17	4.5E-06	1.1E-03	4.2E-03	8.9E-03	1.6E-02	2.7E-02	5.2E-02	4.6E-01
18	4.4E-06	1.2E-03	4.4E-03	9.7E-03	1.7E-02	2.9E-02	5.4E-02	4.5E-01
19	4.2E-06	8.5E-04	3.2E-03	6.9E-03	1.2E-02	1.8E-02	2.8E-02	5.5E-02
20	4.0E-06	7.8E-04	3.0E-03	6.6E-03	1.2E-02	1.8E-02	2.8E-02	5.3E-02
21	5.1E-06	1.2E-03	4.5E-03	9.6E-03	1.7E-02	2.9E-02	5.3E-02	4.9E-01
22	5.0E-06	1.2E-03	4.5E-03	9.7E-03	1.7E-02	2.9E-02	5.2E-02	4.1E-01
23	3.8E-06	9.1E-04	3.4E-03	7.0E-03	1.2E-02	1.9E-02	3.1E-02	8.1E-02
24	5.9E-06	1.2E-03	4.4E-03	9.2E-03	1.6E-02	2.7E-02	4.8E-02	2.9E-01
25	4.7E-06	1.2E-03	4.5E-03	9.7E-03	1.7E-02	2.8E-02	5.2E-02	4.4E-01
26	4.7E-06	1.1E-03	4.4E-03	9.6E-03	1.7E-02	2.9E-02	5.3E-02	4.0E-01
27	3.5E-06	8.4E-04	3.3E-03	7.0E-03	1.2E-02	1.9E-02	2.9E-02	6.1E-02
28	3.4E-06	9.1E-04	3.4E-03	7.2E-03	1.2E-02	2.0E-02	3.1E-02	6.8E-02
29	6.6E-06	1.2E-03	4.6E-03	9.8E-03	1.7E-02	2.8E-02	4.7E-02	2.7E-01
30	6.4E-06	1.3E-03	4.8E-03	1.0E-02	1.8E-02	2.9E-02	5.1E-02	3.8E-01
31	3.5E-06	9.0E-04	3.4E-03	7.1E-03	1.2E-02	1.9E-02	3.1E-02	8.0E-02
32	3.2E-06	8.8E-04	3.4E-03	6.9E-03	1.2E-02	1.9E-02	3.0E-02	6.9E-02

Having described the extraction method of the proposed descriptor, it is now possible to justify the reason for resizing each image to the dimensions of 240×240 ignoring the initial aspect ratio. To begin with, the resizing procedure is imperative for the acceleration of the descriptor extraction process. The reason for selecting these dimensions are as follows:

By applying spatial information to the descriptor, the image is divided into 8×8 or 4×4 Sub-Images. Therefore, the objective is that each side of the image is to be divisible by the L_{Blocks} without leaving a remainder. The size of each Sub-Image produced must

Table 7 Quantization Table for $L_{Block} = 8, L_{Texture} = 16$ and $L_{Bright} = 8$ Part 2.

Line	000	001	010	011	100	101	110	111
33	3.7E-06	1.1E-03	4.3E-03	9.4E-03	1.7E-02	2.9E-02	5.5E-02	4.7E-01
34	6.0E-06	1.2E-03	4.5E-03	9.4E-03	1.7E-02	2.8E-02	5.2E-02	4.2E-01
35	5.6E-06	1.2E-03	4.4E-03	9.3E-03	1.7E-02	2.7E-02	5.0E-02	3.6E-01
36	5.5E-06	1.2E-03	4.5E-03	9.5E-03	1.7E-02	2.8E-02	5.0E-02	3.3E-01
37	2.1E-06	1.1E-03	4.3E-03	9.1E-03	1.6E-02	2.5E-02	4.1E-02	1.1E-01
38	1.8E-06	9.6E-04	3.8E-03	8.1E-03	1.4E-02	2.2E-02	3.5E-02	8.6E-02
39	2.3E-06	9.0E-04	3.5E-03	7.5E-03	1.3E-02	2.1E-02	3.3E-02	7.4E-02
40	2.8E-06	8.5E-04	3.2E-03	6.7E-03	1.2E-02	1.8E-02	2.9E-02	6.8E-02
41	3.4E-06	1.1E-03	4.1E-03	9.0E-03	1.6E-02	2.8E-02	5.3E-02	4.8E-01
42	5.1E-06	1.2E-03	4.5E-03	9.6E-03	1.7E-02	2.9E-02	5.4E-02	5.2E-01
43	5.0E-06	1.3E-03	4.7E-03	9.8E-03	1.7E-02	2.8E-02	5.0E-02	3.4E-01
44	4.8E-06	1.2E-03	4.6E-03	9.8E-03	1.7E-02	2.8E-02	5.0E-02	3.7E-01
45	2.8E-06	1.2E-03	4.9E-03	1.1E-02	1.9E-02	3.1E-02	5.6E-02	3.3E-01
46	2.7E-06	1.2E-03	5.0E-03	1.1E-02	1.9E-02	3.2E-02	5.9E-02	3.9E-01
47	2.6E-06	8.4E-04	3.2E-03	6.8E-03	1.2E-02	1.9E-02	3.0E-02	6.2E-02
48	2.6E-06	8.4E-04	3.3E-03	6.8E-03	1.2E-02	1.8E-02	2.9E-02	6.2E-02
49	2.9E-06	1.2E-03	4.7E-03	1.0E-02	1.8E-02	2.9E-02	5.2E-02	2.3E-01
50	2.0E-06	9.3E-04	3.8E-03	8.0E-03	1.4E-02	2.2E-02	3.7E-02	9.8E-02
51	3.5E-06	9.4E-04	3.5E-03	7.2E-03	1.2E-02	2.0E-02	3.1E-02	7.6E-02
52	3.3E-06	9.5E-04	3.5E-03	7.3E-03	1.3E-02	2.0E-02	3.2E-02	7.2E-02
53	2.7E-06	1.2E-03	4.5E-03	9.6E-03	1.7E-02	2.9E-02	5.4E-02	3.2E-01
54	2.9E-06	1.2E-03	4.7E-03	1.0E-02	1.8E-02	3.0E-02	5.4E-02	3.3E-01
55	3.7E-06	1.2E-03	4.4E-03	9.3E-03	1.7E-02	2.9E-02	5.7E-02	4.4E-01
56	3.1E-06	1.1E-03	4.2E-03	9.0E-03	1.6E-02	2.8E-02	5.5E-02	3.8E-01
57	4.7E-06	1.2E-03	4.6E-03	9.8E-03	1.7E-02	2.9E-02	5.1E-02	2.9E-01
58	2.1E-06	9.8E-04	3.9E-03	8.2E-03	1.4E-02	2.2E-02	3.6E-02	8.6E-02
59	1.9E-06	8.7E-04	3.5E-03	7.4E-03	1.3E-02	2.1E-02	3.3E-02	7.4E-02
60	2.9E-06	9.9E-04	3.8E-03	7.8E-03	1.3E-02	2.1E-02	3.2E-02	6.7E-02
61	3.3E-06	1.2E-03	4.6E-03	9.8E-03	1.8E-02	3.0E-02	5.7E-02	4.1E-01
62	3.4E-06	1.2E-03	4.4E-03	9.5E-03	1.7E-02	2.9E-02	5.6E-02	4.2E-01
63	3.1E-06	1.1E-03	4.2E-03	9.1E-03	1.6E-02	2.8E-02	5.4E-02	3.7E-01
64	2.8E-06	1.1E-03	4.4E-03	9.5E-03	1.7E-02	3.0E-02	5.7E-02	4.0E-01

be a multiple of 3 so that during the extraction of the non-overlapped image blocks (whose dimensions are 3×3) all pixels in the block can be included. An extension of this is also the reason why it was decided to ignore the initial aspect ratio. In many images used in the experiments described in section 7, one dimension is exceedingly small compared to the other (e.g. 512×91). If resizing is attempted while maintaining the initial aspect ratio, then it is very likely that Sub-Images with dimensions that are not multiples of 3 would emerge, and many pixels, and therefore useful information from the image, would be lost.

7 Experiments

The proposed method has been implemented in the image retrieval system img(Rummager)[1] [27] and the on line application img(Anaktisi)[2][28]. Img(Rummager) and img(Anaktisi) are developed by the authors of this paper in the Automatic Control Systems & Robotics Laboratory[3] at the Democritus University of Thrace-Greece. Both applications are implemented in C#.
To evaluate the performance of the proposed descriptor experiments are performed on

[1] http://www.img-rummager.com
[2] http://www.anaktisi.net
[3] ACSL: http://www.ee.duth.gr/acsl

two medical image databases: The IRMA 2007 database and the IRMA 2005[4] database. To evaluate CBIR, several performance evaluation measures have been proposed [29] based on the precision P and the recall R. In our experiments, the Mean Average Precision is employed to evaluate the performance of the proposed descriptor. The MAP was selected to compare the proposed descriptor retrieval results to the results presented by other low level descriptors in the IRMA 2005 database, which are available in the bibliography.

$$Precision = P = \frac{Number\ of\ relevant\ images\ retrieved}{Total\ number\ of\ images\ retrieved} \tag{16}$$

$$Recall = R = \frac{Number\ of\ relevant\ images\ retrieved}{Total\ number\ of\ relevant\ images} \tag{17}$$

The average precision AP for a single query q is the mean over the precision scores after each retrieved relevant item:

$$AP(q) = \frac{1}{N_R} \sum_{n=1}^{N_R} P_Q(R_n) \tag{18}$$

where R_n is the recall after the nth relevant image was retrieved. N_R is the total number of relevant documents for the query. The mean average precision MAP is the mean of the average precision scores over all queries:

$$MAP = \frac{1}{|Q|} \sum_{q \in Q} AP(q) \tag{19}$$

where Q is the set of queries q.
An advantage of the mean average precision is that it contains both precision and recall oriented aspects and is sensitive to the entire ranking.
The searching procedure is described as follows: For all benchmarking database images used in performance measurement experiments, the proposed descriptor is extracted and saved in an **XML** file. For each query image there is a given ground truth. The ground truth is defined by an image set from the database exhibiting visual similarity with the query image. Further details for the type of image forming the ground truth are described in section 7.1. The descriptor is extracted from the query image and its distance from the descriptor of each image included in the XML file is measured, based on a selected measurement. The $P_Q(R_n)$ is calculated for each n image included in the ground truth. Each resulting P_Q is a function of the total number of images retrieved in order to retrieve image n. Then the $AP(q)$ is calculated. The MAP is calculated for each benchmarking image database, being the average of the $AP(q)$s of all database queries.
To compare the ratio, retrieval efficiency with descriptor size, experiments were carried out using 7 approaches, as described in Table 8.

[4] IRMA is courtesy of TM Deserno, Dept. of Medical Informatics, RWTH Aachen

Table 8 The 7 approaches used in the experiments

Approach	L_{Blocks}	$L_{Texture}$	L_{Bright}	Fractal Scanning Method	Size in bytes
E1	1	8	8	None	24
E2	1	8	16	None	48
E3	1	16	8	None	48
E4	4	16	8	Hilbert	768
E5	4	16	8	Z-Grid	768
E6	8	16	8	Hilbert	3072
E7	8	16	8	Z-Grid	3072

7.1 Experiments on IRMA 2007 Medical Image Database

The IRMA 2007 database consists of 12000 fully classified medical radiographs taken randomly from medical routine at the RWTH Aachen University Hospital. 10000 of these were released together with their classification as training data; another 1000 were also published with their classification as validation data to allow for tuning classifiers in a standardized manner. In addition, the database includes 1000 images which comprise the query images.
Each of the 10000 images is described with the IRMA code. The IRMA code is a method for describing medical images and includes information relating to the following:

- Technical code (T) describes the imaging modality
- Directional code (D) models body orientations
- Anatomical code (A) refers to the body region examined
- Biological code (B) describes the biological system examined.

The 1000 query images are also described by the IRMA CODE. Each query image has all the images from the set of the 10000 with the same IRMA CODE as ground truth. In total, 116 different IRMA codes occur in the database.
The IRMA 2007 database was used in the ImageCLEF 2007 image retrieval evaluation for the automatic annotation task.

First, experiments were carried out to identify which method of measuring the distance works better with the proposed descriptor. Using the 3rd approach ($L_{Blocks} = 1$, $L_{Texture} = 16$, and $L_{Bright} = 8$) three similarity metric techniques ware tested in 100 queries.

The distance $D(a, b)$ of two image descriptors X_a and X_b was calculated using 3 different approaches.
First, the distance was calculated using the Euclidean metric, so that the results could be compared with the implementation of the Tamura features in the LIRe Demo retrieval system [30] (which uses the Euclidean distance to compare the Tamura directionality histogram of the images).
Then, the Jensen-Shannon divergence [31] was attempted. This similarity matching method was used in order to compare the Tamura Features on [32] and [18] and computed using the following formulas:

$$\tilde{X}(i) = \frac{X_a(i) + X_b(i)}{2} \tag{20}$$

$$D(a,b) = \sum_i X_a(i) \log \frac{X_a(i)}{\tilde{X}(i)} + \sum_i X_b(i) \log \frac{X_b(i)}{\tilde{X}(i)} \tag{21}$$

Finally, the similarity between the images was calculated using the non binary Tanimoto Coefficient [33], which was used in order to compare the compact composite descriptors of [25] and [26] and is given by:

$$D(a,b) = \frac{X_a^T X_b}{X_a^T X_a + X_b^T X_b + X_a^T X_b} \tag{22}$$

where X^T is the transpose vector of the descriptor X.

Table 9 shows th MAP results for the proposed descriptor using the 3 similarity metric techniques. As the results show, the non-binary Tanimoto coefficient presented the best results.

Table 9 MAP results using several similarity metric techniques

Approach	Similarity Metric	Number Of Queries	Mean Average Precision
E3	Euclidean	100	35.45
E3	Jensen-Shannon	100	34.79
E3	Tanimoto Coefficient	100	**37.14**

Therefore, by using the Tanimoto coefficient, all the approaches were tested on all the queries.
When spatial information is integrated into the proposed descriptor, the similarity matching is differentiated, in order to take into account the information about the sequence with which the histograms of the Sub-Images forms the final histogram. Depending on the fractal scanning technique selected during descriptor extraction, the histograms of the Sub-Images are placed in a different order. The proposed descriptor is defined as $H(i)$, which contains the information extracted from k Sub-Images ($k = L^2_{Blocks}$). This histogram can be analysed as follows:

$$\mathrm{H}\left(\mathrm{i}\right) = \mathrm{H}\left(\mathrm{h}_1\left(\mathrm{j}\right), \mathrm{h}_2\left(\mathrm{j}\right), \mathrm{h}_3\left(\mathrm{j}\right) \ldots \mathrm{h}_k\left(\mathrm{j}\right)\right) \tag{23}$$

$$i \in [0, k \times L_{Textute} \times L_{\mathrm{Brightness}}], j \in [0, L_{Texture} \times L_{\mathrm{Brightness}}]$$

Where $h_k(j)$ are the histograms resulting from each Sub-Image S_j. It is noted that every such histogram is the proposed descriptor without spatial distribution for $L_{Texture} = n$ and $L_{Brightness} = m$.

Neighborhood of the Sub-Image S_j is defined as the neighborhood with $r = 1$

$$S^M_{(j_0)} = (j) : |j - j_0| <= r \tag{24}$$

In total, L_{Blocks} neighborhoods are defined for each $H(i)$. The distance between 2 images is defined as the sum of the distances between $S(j)$ neighborhoods according to Tanimoto Coefficient, from one image in relation to the other.

$$D(a,b) = \sum_{j=0}^{k=L^2_{Blocks}} D_{Tanimoto}(S_a(j), S_b(j)) \quad (25)$$

Experimental results have shown that in the case of distance measurement of the Proposed Descriptor with Spatial Distribution Information, the results are slightly better if the Tanimoto Coefficient is replaced with the Ordinal distance [34]. The reason for this is that the histogram of the proposed descriptor displays a number of zero values. However, the calculation cost necessary for conversion of the histogram to a signature, and more specifically to extended signatures [34], is quite high, and slows the retrieval process down considerably.

The proposed descriptor retrieval results are compared with the corresponding results of the following descriptors: Tamura Directionality Histogram, Gray value Histogram and MPEG-7: Edge Histogram (EHD)[35][36]. Implementations of these low level features are available in [30]
Given that the proposed descriptor is a combination of Tamura directionality histogram and the Gray Value Histogram, it would be useful to compare the obtained retrieval results with the results that would come from other combination processes of these characteristics.
One way of combining the retrieval results that come from different low level features is the Borda Count. Initially, the Borda count was proposed as a single-winner election method in which voters rank candidates in order of preference. In the field of retrieval, it was originally suggested by [37]. The way in which the Borda count was applied in order to combine the results of Tamura directionality histogram and the Gray Value Histogram is described as follows:
Let the query Q consisting of q images. The search is performed on a database of images that includes N images. The results that come from the Tamura directionality histogram are classified according to the distance D, that each image presents from the image Q. Each image l, depending on the position shown in the results, is scored as follows:

$$Rank(l)' = [N - (N - location\ of\ the\ image\ l\ after\ the\ classification)]^{-1} \quad (26)$$

The same procedure is followed and at the results that come from the Gray Value Histogram. The results are classified and each image is scored with $Rank(l)''$.
Finally, for each l image the $Rank(l) = Rank(l)' + Rank(l)''$ is calculated and a final classification of the results according to the Rank of each image is being made. The evaluation process of the results through the MAP, after this classification, is the same as the evaluation process, that has been described.
Observation of the results can easily lead to the following conclusions: The results of the compact versions of the proposed descriptor are directly related to the size of the descriptor. Approaches E2 and E3 produced decidedly better results than approach E1. In addition, when comparing the results of E2 and E3 to each other, we observe that their results are similar, thus strengthening the conclusion that the system's performance is related to the number of bins of the descriptor.
It is taken for granted that the integration of information about the spatial distribution further enhances the system's performance. However, it is observed that in this case

Table 10 MAP results on IRMA 2007 Medical Image Database

Descriptor	MAP	Descriptor	MAP
Proposed Descriptor E7	**36.81**	Proposed Descriptor E3	20.4
Proposed Descriptor E6	36.78	Tamura Directionality Histo.	15.1
Proposed Descriptor E5	35.1	Borda Count	14.7
Proposed Descriptor E4	35.0	Gray Value Histogram	12.4
MPEG-7: Edge Histogram	28.1	Proposed Descriptor E1	10.3
Proposed Descriptor E2	20.9		

the descriptor size does not correlate with the retrieval accuracy. The size difference in approaches E4 and E5 with E6 and E7 is not reflected in the results. It could be concluded that result improvement is achieved through integration of spatial information, and it approaches maximum performance either when using a small number L_{Blocks}, or a large one.

7.2 Experiments on IRMA 2005 Medical Image Database

The IRMA 2005 database consists of 10000 annotated radiographs. The images are separated into 9000 training images and 1000 test images. The images are subdivided into 57 classes. For CBMIR, the relevances are defined by the classes, given a query image from a certain class, all database images from the same class are considered relevant. [38]
The IRMA 2005 database was used in the ImageCLEF 2005 image retrieval evaluation for the automatic annotation task.

Table 11 MAP results on IRMA 2005 Medical Image Database

Descriptor	MAP	Descriptor	MAP
Proposed Descriptor E7	**44.28**	Gray Value Histogram	26.1
Proposed Descriptor E6	44.26	Gabor histogram	25.2
Proposed Descriptor E5	42.8	inv. feature histogram (mon.)	24.4
Proposed Descriptor E4	42.7	inv. feature histogram (relational)	24.1
32x32 image	40.9	LF patches signature	23.0
Xx32 image	35.0	Borda Count	23.0
LF SIFT histogram	32.7	Tamura Directionality Histo.	21.5
LF patches histogram	31.4	LF SIFT global search	20.9
Proposed Descriptor E3	28.3	LF patches global	17.6
Proposed Descriptor E2	28.3	global texture feature	16.4
Proposed Descriptor E1	28.0	LF SIFT signature	10.9
Gabor vector	27.7	MPEG-7: Edge Histogram	10.9

Table 11 shows the MAP results in this database. The values of the remaining descriptors are taken from [38].
Observation of the experiment results strengthens the conclusions that resulted from experiments on the IRMA2007 database. The sole exception is perhaps the high performance that approach E1 produced, which is however considered unsuitable due to the inconsistency of its results. Finally, it is worth noting that use of different fractal scanning methods leads to equally good results, with the Z-Grid method displaying slightly better results. Another one conclusion that can come, is related to the information described in the proposed descriptor. Considering the results of both experiments

and comparing them with the results that come from the Borda Count between the Tamura directionality histogram and the Gray Value Histogram, we perceive that the proposed descriptor (without the E1 approach) shows much better results than the Borda Count. Hence, the combination of Tamura Directionality Histogram and the Gray Value Histogram through the proposed descriptor is better than the combination made by the Borda Count. In addition, the difference between the sizes of the XML files using the 2 approaches, is notable. The size of the file, which carries the information that uses the Borda Count to describe the number of 9000 images for IRMA2005, is 8.47 MB, while for the E3 approach the file size is not exceeding 1.3 MB.
Having studied both experiments, we have reached the conclusion that the approach displaying the best correlation between size and result is approach E5.
The AP results for all the queries are available on line[5].

8 Conclusions and Future Works

This paper proposed a new method combining Brightness and Texture information in one scalable descriptor. Furthermore, the proposed descriptor includes the spatial distribution of the information it describes. The most important feature of the proposed descriptor is that its size adapts according to the storage capabilities of the application that is using it. This characteristic render the descriptor appropriate for use in large medical (or greyscale) image databases.
The experimental results showed that the proposed descriptor can be used for the retrieval of medical images more successfully than other state of art low level features.
The proposed method is an extension of the state of art Tamura Texture to which, in the Directionality histogram, brightness and spatial distribution information is added.
The proposed method can be used as part of a broader retrieval system that uses more characteristics, replacing the Tamura Texture Directionality Histogram extraction unit.
The proposed descriptor's capability of achieving reliable retrieval results, even in its compact versions, allows the possibility of an additional element that can be integrated into the header of DICOM files. DICOM is an all-encompassing data transfer, storage and display protocol built and designed to cover all functional aspects in digital imaging [6]. The combination of visual characteristics described by the proposed descriptor with the textual characteristics already included in the DICOM header could create a hybrid retrieval system with very good results. Corresponding systems that have been developed [39] [40], [41] use non compact descriptors which are extracted from the images in advance and saved in appropriate databases. The difference with the system which could comprise future work lies in the fact that the proposed descriptor would be extracted during image production and integrated into the file header, without particularly affecting the image's size. Integration of the proposed descriptor into DICOM files would contribute to the development of a CAD (computer-aided diagnosis) system. Nowadays, doctors can search for similar cases based on the textual information included in DICOM files. With the proposed method, they would also be able to use the visual information. For example, for a patient exhibiting an unexpected image on a chest MRI scan, the doctor would use the terms "Chest" and "MRI" as textual information and would supply the CAD with the patient's radiology images. The system will return similar cases entered in the database as well as the diagnosis/treatment

[5] http://www.ee.duth.gr/acsl/results

followed in said cases. It should be noted that by no means does the system offer a diagnosis, but rather a reference to similar cases.
Additionally, by using the proposed descriptor, it is also possible to design a quick retrieval system for medical images. A significant factor influencing the speed of a retrieval system is the distance computational time. The proposed descriptor can be used to design a multi-level database which would include different sizes of the descriptor. During the first retrieval phase, a compact version of the descriptor would be used, which may not achieve the best retrieval results, yet it would locate and reject those images least relevant to the query image and exclude them from the next phase of the process. In the next stage, a less compact version of the descriptor would be used, concluding in the final stage where the largest descriptor version would be used, but applied to a very small number of images.
Finally, an element which could improve the performance of the proposed descriptor is the application of a detection technique for the minimum boundary rectangle to the images, and the extraction of features only from the pixels enclosed within it.
Proposed descriptor, img(Rummager) and img(Anaktisi) are programmed in C# and are available as open source projects under the GNU General Public License (GPL).

References

1. Atam Dhawan, Medical Image Analysis, Wiley-IEEE Press, (2003)
2. Paul Miki Willy, Karl-Heinz K., Content-based Medical Image Retrieval (CBMIR): An Intelligent Retrieval System for Handling Multiple Organs of Interest, 17th IEEE Symposium on Computer-Based Medical Systems, 113 (2004)
3. R. Datta, D. Joshi, J. Li, and J. Z. Wang, Image retrieval: Ideas, influences, and trends of the new age, ACM Computing Surveys, 40(2):1-60 (2008)
4. Jian Yaoa, Zhongfei (Mark) Zhanga, Sameer Antanib, Rodney Longb and George Thomab, Automatic medical image annotation and retrieval, Neurocomputing 71 (10-12):2012-2022, (2008)
5. Tagare, H.D. and Jaffe, C.C. and Duncan, J., Medical Image Databases A Content-based Retrieval Approach,Journal of the American Medical Informatics Association 4(3):184-198 (1997)
6. Pianykh, O.S., Digital Imaging and Communications in Medicine (DICOM): A Practical Introduction and Survival Guide, Springer (2008)
7. W. Chu, C. Hsu, A. Cardenas, and R. Taira, Knowledge-based image retrieval with spatial and temporal constructs, IEEE Trans. on Knowledge and Data Engineering, 10(6):872-888 (1998)
8. F. Korn, N. Sidiropoulos, C. Faloustos, E. Siegel, and Z. Protopapas, Fast and effective retrieval of medical tumor shapes, IEEE Tran. on Knowledge and Data Engineering, 10(6):889-904 (1998)
9. D. Comaniciu, P. Meer, D. Foran, and E. Medl, Bimodal system for interactive indexing and retrieval of pathology images, Workshop on Applications of Computer Vision, 76-81, Princeton, NJ (1998)
10. Glatard, T., Montagnat, J., and Magnin, I., Texture based medical image indexing and retrieval: application to cardiac imaging, 6th ACM SIGMM international Workshop on Multimedia information Retrieval MIR '04. ACM, New York, NY, 135-142 (2004)
11. S. A. Chatzichristofis and Y. S. Boutalis, A Hybrid Scheme for Fast and Accurate Image Retrieval Based on Color Descriptors, IASTED International Conference on Artificial Intelligence and Soft Computing (ASC 2007), Spain, (2007)
12. S. A. Chatzichristofis and Y. S. Boutalis, Content Based Medical Image Indexing and Retrieval Using a Fuzzy Compact Composite Descriptor, The Sixth IASTED International Conference on Signal Processing, Pattern Recognition and Applications SPPRA 2009:1-6 (2009)
13. Gustafson, E.E., Kessel, W.C., Fuzzy Clustering with a Fuzzy Covariance Matrix. 18th IEEE Conference on Decision and Control (IEEE CDC, San Diego, CA). Piscataway, USA 761-766 (1979)

14. H. Tamura, S. Mori, and T. Yamawaki, Textural features corresponding to visual perception, IEEE Transactions on Systems, Man, and Cybernetics, 8(6):460-472, (1978)
15. R. Keys, Cubic convolution interpolation for digital image processing, IEEE Transactions on Signal Processing, Acoustics, Speech, and Signal Processing 29:1153 (1981)
16. Deselaers T., Muller H., Deserno T.M., Automatic Medical Image Annotation in ImageCLEF 2007, Pattern Recognition Letters, vol. 29, no. 15, pp. 1988-1995 (2008)
17. Guld, M.O., Keysers, D., Deselaers, T., Leisten, M., Schubert, H., Ney, H., Lehmann, T.M., Comparison of global features for categorization of medical images. Proceedings SPIE 5371 211-222 (2004)
18. Mark O Guld, Christian Thies, Benedikt Fischer, Thomas M. Lehmann, Content-Based Retrieval of Medical Images by Combining Global Features, Lecture Notes in Computer Science, Accessing Multilingual Information Repositories (2006)
19. N. Nikolaou, N. Papamarkos, Color image retrieval using a fractal signature extraction technique, Engineering Applications of Artificial Intelligence 15 81-96 (2002)
20. Sagan, H., Space Filling Curves, Springer-Verlag, New York (1994)
21. Przemyslaw Prusinkiewicz, James Hanan, Lindenmayer Systems, Fractals, and Plants, Springer-Verlag Berlin and Heidelberg GmbH & Co. K, (1989)
22. Poullot, Sebastien and Buisson, Olivier and Crucianu, Michel, Z-grid-based probabilistic retrieval for scaling up content-based copy detection, CIVR '07: Proceedings of the 6th ACM international conference on Image and video retrieval: pp 348-355 (2007)
23. Vonikakis, V., Andreadis, I., & Gasteratos, A., Fast centre-surround contrast modification, IET Image processing 2(1), 19-34 (2008)
24. B. Mertzios, K. Tsirikolias, Logic Filters: Theory and Applications, Nonlinear Image Processing, Chapter 11, S. Mitra and G. Sicuranza editors, Academic Press, ISBN:0125004516 (2004)
25. S. A. Chatzichristofis and Y. S. Boutalis, CEDD: Color and Edge Directivity Descriptor - A Compact Descriptor for Image Indexing and Retrieval, 6th International Conference in advanced research on Computer Vision Systems ICVS 2008, 312-322, (2008)
26. S. A. Chatzichristofis and Y. S. Boutalis, FCTH: Fuzzy Color and Texture Histogram - A low Level Feature for Accurate Image Retrieval, 9th International Workshop on Image Analysis for Multimedia Interactive Services, 191-196, (2008)
27. S. A. Chatzichristofis, Y. S. Boutalis and Mathias Lux, Img(Rummager): An Interactive Content Based Image Retrieval System, 2nd International Workshop on Similarity Search and Applications (SISAP), To Appear (2009)
28. K Zagoris, S. A. Chatzichristofis, Nikolas Papamarkos and Y. S. Boutalis, Img(Anaktisi): A Web Content Based Image Retrieval System, 2nd International Workshop on Similarity Search and Applications (SISAP), To Appear (2009)
29. Muller H, Muller W, Squire D M, Marchand-Maillet S, Pun T. Performance Evaluation in Content-Based Image Retrieval: Overview and Proposals. Pattern Recognition Letters (Special Issue on Image and Video Indexing), 22(5):593601 (2001)
30. Mathias Lux & Savvas A. Chatzichristofis, LIRe: Lucene Image Retrieval - An Extensible Java CBIR Library, ACM MM 2008, Vancouver Canada, 1085-1087 (2008)
31. Puzicha, J., Rubner, Y., Tomasi, C., Buhmann, J., Empirical evaluation of dissimilarity measures for color and texture, International Conference on Computer Vision, 2 1165-1173 (1999)
32. Thomas M. Lehmann, Mark O. Guld, Daniel Keysers, Thomas Deselaers,Henning Schubert, Berthold Wein, and Klaus Spitzer, Similarity of Medical Images Computed from Global Feature Vectors for Content-Based Retrieval, M .Gh. Negoita et al. (Eds.): KES 2004, LNAI 3214, pp. 989-995, Springer-Verlag Berlin Heidelberg (2004)
33. Chi, Z., Yan, H. and Pham, T., Fuzzy Algorithms: With Applications to image processing and pattern recognition, Advance in fuzzy systems, Applications and theory, Volume 10, World Scientific, (1996)
34. Serratosa, F. and Sanfeliu, A., Signatures versus histograms: Definitions, distances and algorithms, Pattern recognition (39), 921-934 (2006)
35. B. S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan, Akio Yamada, Color and Texture Descriptors, IEEE Trans. on Circuits and Systems for Video Technology 11, 6 703-715 (2001)
36. Chee Sun Won, Dong Kwon Park, Soo-Jun Park, Efficient Use of MPEG-7 - Edge Histogram Descriptor, ETRI Journal, 24 (2002)
37. S. Jeong, K. Kim, B. Chun, J. Lee, Y.J. Bae, An effective method for combining multiple features of image retrieval,IEEE Region 10 Conference: TENCON99,pp. 982985 (1999)

38. Thomas Deselaers, Daniel Keysers and Hermann Ney, Features for Image Retrieval: An Experimental Comparison, Information Retrieval 11, 2, 77-107 (2007)
39. Luz Jr, A. and Abdala, D.D. and Wangenheim, A.V. and Comunello, E., Analyzing DICOM and non-DICOM Features in Content-Based Medical Image Retrieval: A Multi-Layer Approach, Proceedings of the 19th IEEE Symposium on Computer-Based Medical Systems, pp93-98,IEEE Computer Society Washington, DC, USA (2006)
40. Caicedo, J.C. and Gonzalez, F.A. and Triana, E. and Romero, E., Design of a medical image database with content-based retrieval capabilities, Lecture Notes in Computer Science 4872, Springer (2007)
41. Deselaers, T. and Weyand, T. and Keysers, D. and Macherey, W. and Ney, H., FIRE in ImageCLEF 2005: Combining content-based image retrieval with textual information retrieval, Lecture Notes in Computer Science 4022, Springer (2006)