Computers and Electrical Engineering 38 (2012) 1358–1376

Contents lists available at SciVerse ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

ICTEDCT-CBIR: Integrating curvelet transform with enhanced dominant colors extraction and texture analysis for efficient content-based image retrieval ☆

Sherin M. Youssef

Department of Computer Engineering, College of Engineering & Technology, Arab Academy for Science & Technology (AAST), Alexandria, Egypt

ARTICLE INFO

Article history:
Available online 8 June 2012

ABSTRACT

A novel Integrated Curvelet-based image retrieval scheme (ICTEDCT-CBIR) has been proposed, for the purpose of effectively retrieving more similar images from large digital image databases. The proposed model Integrates Curvelet Multiscale ridgelets with Region-based vector codebook Subband Clustering for enhanced dominant colors extraction and texture analysis. An important ingredient of the curvelet transform is to restore sparsity by reducing redundancy across scales. The discrete curvelet transform makes use of a dyadic sequence of scales, and a bank of filters with the property that the pass band filter is concentrated near the frequencies. An enhanced Region-based vector codebook Sub band Clustering (RBSC) has been proposed for effectively extract dominant colors from the color histogram of the transformed image sub-bands. An integrated matching scheme, based on most similar Highest Priority (MSHP) principle, is used to compare the query and target images. Experimental analysis has been carried out to verify the efficiency of the proposed ICTEDCT-CBIR model. Experimental results showed that the proposed approach has better retrieval performance. First, curvelets capture more accurate texture information. Second, as curvelets are tuned to different orientations, it captured more accurate directional features than wavelets. As the experimental results indicated, the proposed technique outperforms other retrieval schemes in terms of average precision with higher precision–recall crossover point values.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Modern image search engines retrieve images based on their visual contents, commonly referred to as Content Based Image Retrieval (CBIR) systems [1,2]. CBIR systems have found applications in various fields like fabric and fashion design, interior design as panoramic views, art galleries [3,4], geographical information systems, remote sensing and management of earth resources, scientific database management, medical imaging, trademark and copyright database management, the military, law enforcement and criminal investigations, picture archiving and communication systems, retailing and image search on the Internet [4]. As the bandwidth availability increases to access the internet, users will be allowed to search for and browse through video and image databases located at remote sites [5]. For that reason we need fast retrieval of images from large databases which is a new point of research addressed nowadays. CBIR is an important alternative to traditional text-based image searching and can greatly enhance the accuracy of the information being returned. It aims to develop an efficient visual-content-based technique to search, browse and retrieve relevant images from large-scale digital image collections [6]. Image retrieval is the task of searching for images from an image database.

☆ Reviews processed and recommended for publication to Editor-in-Chief by Deputy Editor Dr. Ferat Sahin.
E-mail addresses: sherin@aast.edu, sherin.youssef@gmail.com

0045-7906/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.compeleceng.2012.05.010

The CBIR process consists of calculating a feature vector that characterizes some image properties, and stored in the image feature database. The user provides a query image, and the CBIR system computes the feature vector for it, and then compares it with the particular image feature database images. The relevance comparison is done by using some distance measurement technique, and the minimum or permissible distances are the metrics for the matched or similar images. The features vector should be able enough to fully characterize image structural and spatial properties, which retrieve the similar images from the image database [2–4,7].

In a CBIR system, the images are indexed by their visual contents as the features. These include the characteristics such as colour, texture, shape. When a query image is given, the features of the query image are extracted to match the features in the feature database by a pre-established algorithm, so that a group of similar images to the query image can be returned as the retrieval images. CBIR techniques use low-level features like texture, color, and shape to represent images and retrieves images relevant to the query image from the image database. Among those low level image features, texture feature has been shown very effective and subjective.

A number of texture features have been proposed in literature [8–10], including statistic methods and spectral methods. However, most of them are not able to accurately capture the edge information which is the most important texture feature in an image. Color is one of the most reliable visual features that are also easier to implement in image retrieval systems. Color is independent of image size and orientation, because, it is robust to background complication. Color histogram is the most common technique for extracting the color features of colored images [11,12]. Color histogram tells the global distribution of colors in the images. It involves low computation cost and it is insensitive to small variations in the image structure. However, color histogram hold two major shortcomings. They are unable to fully accommodate the spatial information, and they are not unique and robust. Two dissimilar images with similar color distribution produce very similar histograms. Moreover, similar images of same point of view carrying different lighting conditions create dissimilar histograms. Many researchers suggested the use of color correlogram for avoiding inconsistencies involving the spatial information [11]. Multiresolution histograms [12,13] are also suggested to ameliorate image retrieval process. Gaussian filtering may also be used for multiresolution decomposition of an image [13].

In this paper, a texture feature based on curvelet transform is proposed. The technique makes use of curvelet transform which represents the latest research result on multiresolution analysis [14,15]. By combining the advantages of the two methods, image edge information is captured more accurately than conventional spectral methods such as wavelet and Gabor filters. Curvelet was originally proposed for image denoising and has shown promising performance. In this paper, we describe the theory and implementation of curvelet, then combine a Region-based vector codebook Subband Clustering (RBSC) for dominant color extraction with efficient curvelet-based sub-band texture extraction. Our proposed ICTEDCT-CBIR system is based on curvelet multiscale decomposition combined with a spatial band pass filtering operation to isolate different scales. Efficient color and texture feature extraction are proposed. The rest of the paper is organized as follows. A literature survey is introduced in Section 2. Section 3 presents an overview of digital curvelet transform.. In Section 4, the proposed ICTEDCT-CBIR image retrieval model will be introduced with illustration of model phases. Section 4.5 illustrates the similarity matching scheme. Experimental results and discussion are illustrated in Section 5 and, finally, conclusions are driven in Section 6.

2. Literature survey

A variety of techniques have been developed for extracting texture features. These techniques can be broadly classified into spatial methods and spectral methods. In spatial approach most techniques rely on computing values of what are known as low order statistics from query and stored images [8]. These methods compute texture features such as the degree of contrast, coarseness, directionality and regularity [8,9]; or periodicity, directionality and randomness [4]. Alternative methods of texture analysis for image retrieval include the use of Gabor filters [16,17], wavelet [18,19]. Statistic techniques suffer from insufficient number of features and sensitive to image noise. The spectral methods in literature, however, do not capture edge information accurately [20].

In BabuRao et al. [5], an image retrieval system CTDCIRS (color-texture and dominant color based image retrieval system) is provided to retrieve the images using three features called dynamic dominant color (DDC), Motif co-occurrence matrix (MCM) and difference between pixels of scan pattern (DBPSP) is proposed. Initially the image is divided into eight coarse partitions using color quantization algorithm and the eight dominant colors are obtained from eight partitions. Next the texture of the image is represented by the MCM and DBPSP. MCM is derived using a motif transformed image. MCM is similar to color co-occurrence matrix (CCM). MCM is the conventional pattern co-occurrence matrix that calculates the probability of the occurrence of same pixel color between each pixel and its adjacent ones in each image, and this probability is considered as the attribute of the image.MCM captures third order image statistics in the local neighborhood which describes the direction of textures but not the complexity of the textures. That is why the DBPSP, in that paper, is considered as one of the texture features. The three features Dominant color, MCM and DBPSP are integrated to facilitate the image retrieval system.

In Rao et al. [21], an image is uniformly divided into eight coarse partitions as a first step. After the above coarse partition, the centroid of each partition (“color Bin” in MPEG-7) is selected as its dominant color. Texture of an image is obtained by using Gray Level Co-occurrence Matrix (GLCM). Color and texture features are normalized. Shape information is captured in terms of edge images computed using **Gradient Vector Flow** fields. Invariant moments are then used to record the shape

features. The combination of the color and texture features of an image in conjunction with the shape features is provided as a feature set for image retrieval. Weighted Euclidean distance of color, texture and shape features is used in retrieving the similar images. The gray level edge images of the *R*, *G* and *B* individual planes are taken and the shape descriptors are computed.

However, when used as the image force for active contours, the gradient has the disadvantage of having a restricted capture range. Although GVF provides a good capture range, it sometimes leads to boundary delocalization. As an alternative, pressure forces have shown promising results for histological middle-ear images.

The gradient, as an external force, suffers from the drawback of having a weak capture range, and therefore active contours using gradient alone must be initialized close to the boundary. The use of GVF reduces the capture-range limitation of the gradient method and the initial contour can be located far from the boundary. However, the results of Hatamzadeh-Tabrizi et al. [22] show that the optimal blending factor for GVF depends on the contrast over the boundary, so as the contrast varies from slice to slice the blending factor must be varied. Choosing a smaller blending factor increases the number of iterations since it reduces the GVF force far from the boundary. The proper value for the blending factor was found by trial and error to range from 0.05 to 0.2. For some structures with very low boundary contrast, GVF failed and even very small values for blending factor were not successful. It is important to note that decreasing blending factor causes the effect of the Laplacian term to decrease, and when it is very small, e.g., 0.02, GVF has a behaviour very similar to that of the gradient alone. Another problem with GVF occurs for close neighboring structures, e.g., malleus and incus: even with high boundary contrast, GVF may delocalize the boundaries between them. Furthermore, some structures of interest contain small regions of different density and these regions produce false edges. GVF intensifies these false edges, and therefore the starting contour must be initialized far from these edges.

Although wavelets have been widely used in image analysis, traditional wavelets perform well only at representing point singularities, since they ignore the geometric properties of structures and do not exploit the regularity of edges [34]. Curvelet transform was proposed in order to overcome the drawbacks of conventional wavelet transform, the curvelet transform has an almost optimal sparse representation of objects with C2–singularities [15]. Combined with other methods, excellent performance of the curvelet transform has been shown in image processing [14].

In image processing, edges are typically curved rather than straight and ridgelets [23,24] alone cannot yield efficient representations. However at sufficiently fine scales, a curved edge is almost straight, and so to capture curved edges, one ought to be able to deploy ridgelets in a localized manner, at sufficiently fine scales. Two approaches to localization of ridgelets are possible. In [25] global HSV color and GLCM texture features are used to retrieve the images. HSV color space is widely used in computer graphics, visualization in scientific computing and other fields [26–28]. In this space, hue is used to distinguish colors, saturation is the percentage of white light added to a pure color and value refers to the perceived light intensity [27,28]. The advantage of HSV color space is its ability to separate chromatic and achromatic components. Therefore we selected the HSV color space to extract the color features according to hue, saturation and value.

Ni and Leng [29] attempted an initial application of curvelet on color image retrieval, but it was not implemented properly and no meaningful result was reported. In their work, no benchmark image database is used, no retrieval accuracy was reported and there was no comparison with other techniques. From each ridgelet block, a pair of features has been extracted, where the first one represents the edge strength and the other represents the angle difference between the significant edges. This method is not effective enough in representing images with complex contents and also has some notable problems. First of all, it is not clearly mentioned which coefficients corresponds to which scale in this paper. Second, no standard database has been chosen for retrieval purpose. Third, the size of the images and the number of levels of curvelet decomposition are not mentioned. Finally, the similarity measurement technique is not described and it is not compared with any other well established spectral approach of CBIR.

3. Digital curvelet transform

Curvelets are based on multiscale ridgelets combined with a spatial bandpass filtering operation to isolate different scales [24,28,30]. The Discrete Curvelet Transform decomposites an image into several levels of curvelet coefficients, each level has a number of sub-bands locate at different directions. Given an image function $f(x,y)$, the continuous ridgelet transform is given as [27]:

$$\mathcal{R}_f(a,b,\theta) = \iint \psi_{a,b,\theta}(x,y) f(x,y)\,dx\,dy \tag{1}$$

where $a > 0$ is the scale, $b \in R$ is the translation and $\theta \in [0,2\pi]$ is the orientation. The ridgelet is defined as:

$$\psi_{a,b,\theta}(x,y) = a^{-\frac{1}{2}}\psi\left(\frac{x\cos\theta + y\sin\theta - b}{a}\right) \tag{2}$$

For a typical ridgelet, it is oriented at angle θ, and is constant along lines: $x\,cos(\theta) + y{\cdot}sin(\theta) = constant$. It can be seen that the ridgelet is linear in edge direction and is much sharper than a conventional sinusoid wavelet.

For comparison, the 2-D wavelet is given as:

$$\psi_{a_1,a_2,b_1,b_2,\theta}(x,y) = a_1^{-\frac{1}{2}} a_2^{-\frac{1}{2}} \psi\left(\frac{x-b_1}{a_1}\right)\psi\left(\frac{y-b_2}{a_2}\right) \tag{3}$$

As can be seen, the ridgelet is similar to the 2-D wavelet except that the point parameters $(b1, b2)$ are replaced by the line parameters (b, θ). In other words, the two transforms are related by:

Wavelet: $\psi_{scale,point-position}$
Ridgelet: $\psi_{scale,line-position}$

This means that ridgelet can be tuned to different orientations and different scales to create the curvelets, in the similar way to Gabor filters. Fig. 1a and b shows a single curvelet and the curvelets tuned to two scales and different number of orientations at each scale. Different from Gabor filters which only cover part of the spectrum in the frequency domain [6], curvelets have a complete cover of the spectrum in frequency domain. That means, there is no loss of information in curvelet transform in terms of capturing the frequency information from images. Fig. 1c shows the curvelet tiling and cover of the spectrum of a 512×512 images with five scales. The shaded wedge shows the frequency response of a curvelet at orientation 4 and scale 4. It can be seen, the spectrum cover by curvelets is complete.

Like ridgelets [24], curvelets occur at all scales, locations, and orientations. However, while ridgelets all have global length and variable widths, curvelets in addition to a variable width have a variable length and so a variable anisotropy. The length and width at fine scales are related by a scaling law width length and so the anisotropy increases with decreasing scale like a power law. Recent work shows that thresholding of discrete curvelet coefficients provide near-optimal-term representations of otherwise smooth objects with discontinuities along curves.

In consideration of searching speed, we use two or three levels of curvelet transform coefficients within all directions to represent spatial feature rather than using all of the coefficients.

The digital curvelet transform is taken on a 2-D Cartesian grid $f[m,n]$, $0 \leqslant m < M$, $0 \leqslant n < N$

$$C^D(a,b,\theta) = \sum_{\substack{0 \leqslant m < M \\ 0 \leqslant n < N}} f[m,n]\psi^D_{a,b,\theta}[m,n] \tag{4}$$

Based on decomposition using the multiscale ridgelet pyramid, thin "brushstrokes" can be used to reconstruct the image, with all lengths and widths available. This would allow us to trace sharp edges precisely using a few elongated elements with very narrow widths. An important ingredient of the curvelet transform is to restore sparsity by reducing redundancy across scales. In detail, one introduces interscale orthogonality by means of subband filtering. Different levels of the multiscale ridgelet pyramid are used to represent different subbands of a filter bank output. At the same time, this subband decomposition imposes a relationship between the width and length of the important frame elements so that they are anisotropic and obey width length. The discrete curvelet transform of a continuum function makes use of a dyadic sequence of scales, and a bank of filters with the property that the pass band filter is concentrated near the frequencies, e.g., In wavelet theory, one uses a decomposition into dyadic subbands. In contrast, the subbands used in the discrete curvelet transform of continuum functions have the nonstandard form.

This leads to why curvelet has better retrieval performance than wavelet. Two main aspects have to be highlighted. First, curvelets capture more accurate edge information or texture information than wavelets. Second, as curvelets are tuned to different orientations, curvelets capture more directional features and more accurate directional features than wavelets.

The mind of Discrete Curvelet Transform (DCT) is simple, it chooses a suitable sampling at the range of scales, locations and directions. The DCT thus provides a decomposition of the image f into j detail sub-bands (levels), with L_j directions on each level, and $K_{jl,1} \times K_{jl,2}$ spatial translations for each of these directions [32]. The slopes defined by the angles θ_l are equispaced. The frequency window function with compact support. Therefore, DCT decomposes the frequency space into dyadic rectangular coronae, each of which is divided into wedges, the number of wedges doubles with every second level.

Fig. 2 illustrates the curvelet decomposition of the original image into subbands followed by the spatial partitioning of each subband. The ridgelet transform is then applied to each block. The basic strategy for calculating the continuous ridgelet transform is first to compute the Radon transform [31] and second, to apply a 1-D wavelet transform to the slices. The curvelet decomposition is the sequence of the following steps: (1) *Subband Decomposition*. The object is decomposed into

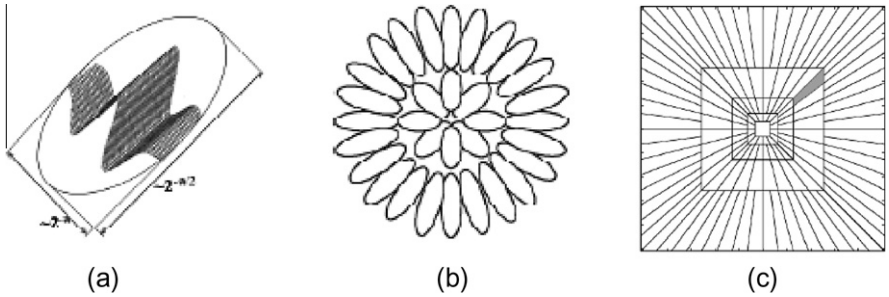

Fig. 1. (a) A single curvelet. (b) Curvelets tuned to different scales and orientations. (c) The tiling of frequency plan by curvelets.

subbands, (2) *Smooth Partitioning*. Each subband is smoothly windowed into "squares" of an appropriate scale (of side length), (3) *Renormalization*. Each resulting square is renormalized to unit scale, and (4) *Ridgelet Analysis*. Each square is analyzed via the discrete ridgelet transform. In this definition, the two dyadic subbands and are merged before applying the ridgelet transform. In developing a transform for digital n by n data which is analogous to the discrete curvelet transform of a continuous function $f(x_1,x_2)$, we replace each of the continuum concepts with the appropriate digital concept mentioned in sections above. In the digital application, leaving these subbands separate, applying spatial partitioning to each subband and applying the ridgelet transform on each subband separately led to improved visual and numerical results.

Once the curvelet coefficients have been obtained, texture features are computed for the curvelet. For each curvelet, γ texture features are obtained and ξ dominant color are obtained. If n curvelets are used for the transform, $(\gamma + \xi)\cdot n$ features are obtained. A $(\gamma + \xi)\cdot n$ dimension feature vector is used to represent each image in the database for image retrieval. This feature extraction is applied to each of the images in the database. At the end, each image in the database is represented and indexed using its curvelet feature vector. This feature extraction is applied to each of the images in the database. At the end, each image in the database is represented and indexed using its curvelet-based feature vector. Given a query, its curvelet-based texture and dominant color features are computed. The system then compares the query feature vector with all the feature vectors in the database. Finally, the images in the database are ranked according to their distance d to the query image, and the ranked list of images is returned to the user.

4. The proposed ICTEDCT-CBIR image retrieval model

The proposed ICTEDCT-CBIR model will pass through the following phases as illustrated in Fig. 3. In the following subsections, each phase will be described.

A description of each phased of the proposed model will be illustrated in the following sections, including:

- Transfer to HSV color space.
- Apply multiscale curvelet transform to get different multiscale subbands.
- The proposed Region-based Subband Clustering approach (*RBSC*). For each selected cluster i ($i = 1, \ldots, N_{DCD}$), calculate the centroid, v_i. Based on each cluster in descending order of the number of clusters, fill in the *DCD* structure of Eq. (15).
- Feature extraction using Dominant Color Extraction (*DCE*).
- Feature extraction using Gray-Level Co-occurrence Matrix (*GLCM*), contrast, entropy, energy and inverse difference.
- Similarity matching using Similar Highest Priority (*MSHP*) principle based on Bipartite graph of sub blocks of both query and target images.

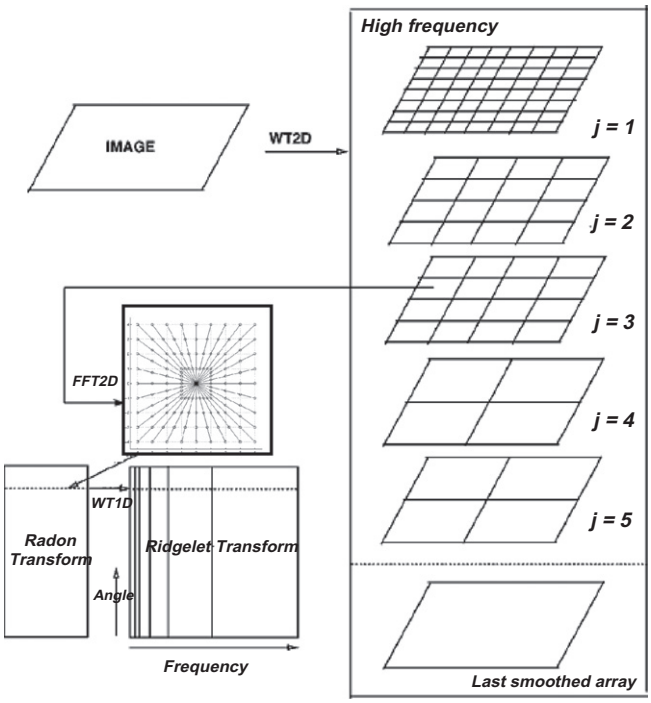

Fig. 2. Curvelet transform scheme.

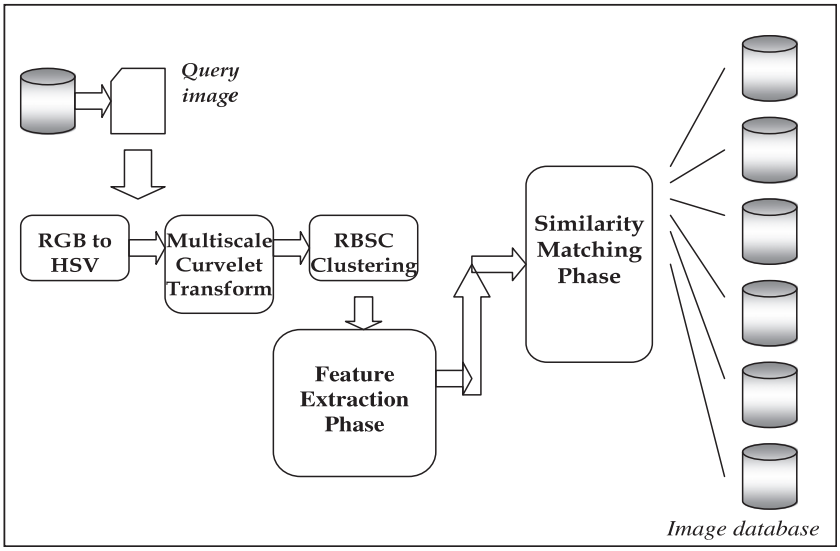

Fig. 3. The proposed ICTEDCT-CBIR model.

In the following sections, a description of each phase of the proposed ICTEDCT-CBIR model will be introduced and discussed.

4.1. Multiscale curvelet algorithm

Before applying the feature extraction phase, the query image is transformed using curvelet transform. We applied curvelet phase based on ridgelet transform [23,24] as a component step, and implemented curvelet subbands using a filter bank of à trous wavelet filters [33]. Simple thresholding of the curvelet coefficients is very competitive with "state of the art" techniques based on wavelets, including thresholding of decimated or undecimated wavelet transforms and also including tree-based Bayesian posterior mean methods. Moreover, the curvelet reconstructions exhibit higher perceptual quality than wavelet-based reconstructions, offering visually sharper images and, in particular, higher quality recovery of edges and of faint linear and curvilinear features.

- Apply the á trous algorithm with J scales;
- Set $B_1 = B_{min}$;
- For $j = 1, \ldots, J$ do
  (a) Partition the subband w_j with a block size B_j and apply the digital ridgelet transform to each block.
  (b) If j modulo 2 = 1 then $B_{j+1} = 2B_j$;
  Else $B_{j+1} = B_j$.

The side length of the localizing windows is doubled *at every other*dyadic subband, hence maintaining the fundamental property of the curvelet transform which says that elements of length about serve for the analysis and synthesis of the th subband. Note also that the coarse description of the image is not processed. Finally, the method enjoys exact reconstruction and stability, because this invertibility holds for each element of the processing chain. Fig. 4 shows a few curvelets at different scales, orientations and locations.

So a basic strategy for calculating the continuous ridgelet transform is first to compute the Radon transform and second, to apply a 1-D wavelet transform to the slices. In this section we develop a digital procedure which is inspired by this viewpoint, and is realizable on by numerical arrays. Fig. 4 illustrates digital polar grid in the frequency domain for an n by n image.

4.2. The HSV color space

4.2.1. The choice of color space

The presentation of color feature depends on color space and it is not all color spaces that are consistent with humans' perception. The color spaces used in MPEG-7 include: RGB, HSV, YCrCb, HMMD and the linear transformation matrix of *R*, *G*, and *B*. The RGB space is widely used in color histograms, where the pixels are distributed on the three axes *R*, *G* and *B*. The three primary colors Red, Green and Blue take their values between 0 and 1 or between 0 and 255. The lack of color ("black"

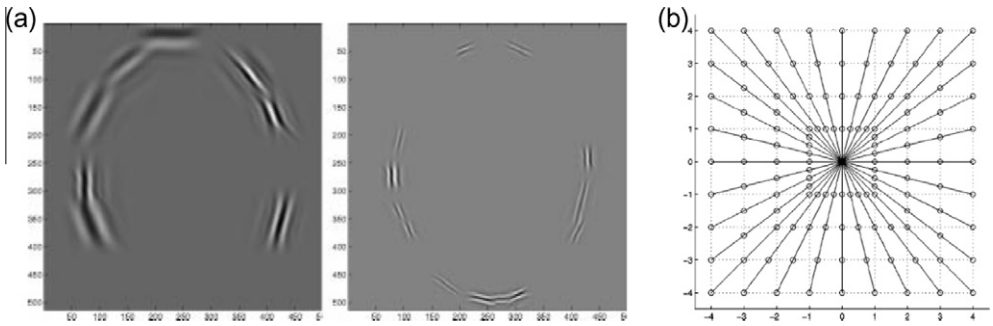

Fig. 4. The figure displays the set of radar lines joining pairs of symmetric points from the boundary of the square. (a) A view curvelets. (b) Illustration of the digital polar grid in the frequency domain for an n by n image ($n = 8$).

color) is symbolized by the triplet (0,0,0). On the other hand the point (255,255,255) corresponds to the maximum of color, i.e. the "white" color. The RGB color space is represented by a cube (as shown Fig. 5. The color histogram is appropriate as a global property and does not require knowledge of how an image is composed of different objects. This leads to retrieve similar images based on exact matching without considering smooth color transitions in natural images. Most operating systems, image processing, programs and texts treat images as collections of pixels comprised of red, green and blue values. This is convenient for display purposes, since computer monitors output color by combining different amounts of red, green and blue. However, most users do not think of color in these terms. They tend to think about color the same way they perceive it. HSV color space solves this problem by representing color in a way similar to how human think. Particularly HSV models color to be composed of Hue; represents color tone (e.g.: red, pink and blue), Saturation; represents the amount of color (e.g.: bright and pale) and Value (sometimes called Lightness and Intensity); represents the amount of light (e.g.: dark and light).

Hue (H), the color type. It ranges from 0 to 360°, with red at 0°, green at 120°, blue at 240° and so on.
Saturation (S) of the color ranges from 0% to 100%. Sometimes it called the "purity". The lower the saturation of a color, the more "grayness" is present and the more faded the color will appear.
Value (V), the Brightness (B) of the color ranges from 0% to 100%. It is a nonlinear transformation of the RGB color space.

The HSV space is represented by a cone, in which the axis of the cone is the scale progression from black to white, distance from the central axis is the saturation, and the direction is the hue (as shown in Fig. 5b).

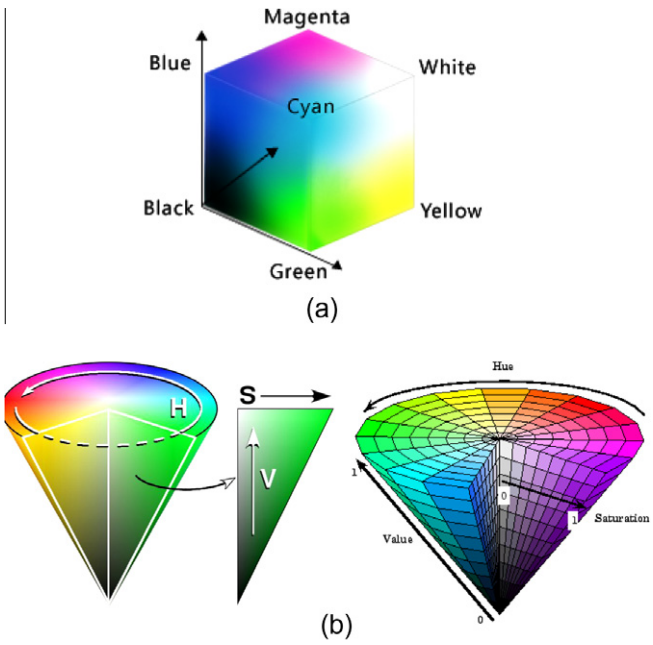

Fig. 5. (a) The RGB color space. (b) HSV color space. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The HSV color space provides an intuitive representation and approximates the way in which human perceive and manipulate color. The hue (H) represents the dominant spectral component color in its pure form, as in green, red or yellow. Adding white to the pure color changes the color: the less white, the more saturated the color is. This corresponds to the saturation (S). The value (V) corresponds to the brightness of color. The coordinate system is cylindrical, and is often represented by a subspace defined by a six-sided inverted pyramid. This color model is represented by Munsell 3D space reference frame. It can perceive the change of each color weight and has linear retractility. So it adapts to images comparability measure based on color. To convert from RGB to HSV (assuming normalized RGB values) first find the maximum and minimum values from the RGB triplet. Saturation, S, is then:

$$S = \frac{(\max - \min)}{\max} \tag{5}$$

And value, V, is:

$$V = \max \tag{6}$$

The Hue, H, is then calculated as follows. First calculate $R'G'B'$:

$$\mathbf{R}' = \frac{\max - R}{\max - \min}, \quad \mathbf{G}' = \frac{\max - G}{\max - \min}, \quad \mathbf{B}' = \frac{\max - G}{\max - \min} \tag{7}$$

If saturation, S, is 0 (zero) then hue is undefined (i.e. the color has no hue therefore it is monochrome) otherwise:

$$\begin{aligned}
&\text{then, if R} = \max \text{ and G} = \min && H = 5 + B' \\
&\text{else if R} = \max \text{ and G} \neq \min && H = 1 - G' \\
&\text{else if G} = \max \text{ and B} = \min && H = R' + 1 \\
&\text{else if G} = \max \text{ and B} \neq \min && H = 3 - B' \\
&\text{else if R} = \max && H = 3 + G' \\
&\text{otherwise} && H = 5 - R'
\end{aligned} \tag{8}$$

Hue, H is then converted to degrees by multiplying by 60 giving HSV with S and V between 0 and 1 and H between 0 and 360.

4.2.2. Using the HSV color space

HSV color space is superior to other color spaces for image processing. The advantage of HSV color space is its ability to separate chromatic and achromatic components. Therefore, we selected the HSV color space to extract the color features according to hue, saturation and value. The values of H in an HSV color space are the angles of the cylinder, and the values for S are the radii. In the HSV color space, each component occupies a large range of values. If we directly use H, S and V components to represent the color feature, it requires lot of computation. So it is better to quantify the HSV color space into non-equal intervals. Based on the color model of substantial analysis, we divided color into eight parts. Saturation and intensity is divided into three parts separately in accordance with the human eyes to distinguish. In accordance with the different colors and subjective color perception quantification, quantified hue (H), saturation (S) and value (V) are showed as Eq. (1).

$$H = \begin{cases} 0 \ if \ h \in [316, 20] \\ 1 \ if \ h \in [21, 40] \\ 2 \ if \ h \in [41, 75] \\ 3 \ if \ h \in [76, 155] \\ 4 \ if \ h \in [156, 190] \\ 5 \ if \ h \in [191, 270] \\ 6 \ if \ h \in [271, 295] \\ 7 \ if \ h \in [296, 315] \end{cases} \quad S = \begin{cases} 0 \ if \ s \in [0, 0.2) \\ 1 \ if \ s \in [0.2, 0.7) \\ 2 \ if \ s \in [0.7, 1) \end{cases} \quad V = \begin{cases} 0 \ if \ v \in [0, 0.2) \\ 1 \ if \ v \in [0.2, 0.7) \\ 2 \ if \ v \in [0.7, 1) \end{cases} \tag{9}$$

The use of all matrix elements in the feature vector, will contribute to online computation significantly and this can be worse in real situations where database size is very large. Also, an increase in feature dimension essentially means an exponential growth in the number of training samples. Hence, dimension reduction has become a critical issue in feature representation and image indexing of CBIR systems. If there is no information loss due to dimension reduction then system performance would be the same in both spaces. A one-dimensional feature vector named G is constructed in accordance with the quantization levels above, using the three-dimensional feature vector for different values of H, S, V with different weights to form:

$$G = w_1 \cdot H + w_2 \cdot S + w_3 V \tag{10}$$

We studied the effect of H-only and H, S, V together on the system performance. Although, addition of S and V increases feature dimension, it provides extra information about images and thus improves retrieval. We applied different experimental analysis for setting the different values for the quantified series of $S(Q_s)$ and setting the quantified series of $V(Q_v)$. Experiments showed a higher average retrieval precision produced on setting the weights in Eq. (1) correlated to Q_s and Q_v as follows. The weighted dependency on the Hue value greater or equal triple the weighted value for (S), and setting the weighted dependency on Saturation value of (S) more than double that of (V), i.e.:

$$w_2 \geqslant 3 \cdot w_3, w_1 > 2 \cdot w_2 \tag{11}$$

$$G = Q_s \cdot Q_v \cdot H + Q_v \cdot S + V \tag{12}$$

Correspondingly, in this paper w_3 is set to 1, w_2 is set to 3 and w_3 is set to. 9

$$G = 9H + 3S + V \tag{13}$$

Our conducted experiments with the above weight settings showed that there is no information loss due to dimension reduction. In this way, three-component vector of HSV form One-dimensional vector, which quantize the whole color space for the 72 kinds of main colors. So we can handle 72 bins of one-dimensional histogram. This quantification can be effective by reducing the computational time and complexity. It will be much of the deviation of the calculation of the similarity if we do not normalize, so we must normalize the components to the same range. The process of normalization is to make the components of feature vector equal importance. Color histogram is derived by first quantizing colors in the image into 72 bins in HSV color space, and counting the number of image pixels in each bin. Therefore, this paper represents the one-dimensional vector G by constructing a cumulative histogram of the color characteristics of image after using non-interval HSV quantization for G.

4.3. Feature extraction using Dominant Color Extraction (DCE)

Dominant color descriptors (DCD) describe the salient color distributions in an image or a region of interest, and provides an effective, compact, and intuitive representation of colors presented in an image. However, DCD similarity matching does not fit human perception very well, and it will cause incorrect ranks for images with similar color distribution [43,35]. DCD contains two main components: representative colors and the percentage of each color. DCD can provide an effective, compact, and intuitive salient color representation, and describe the color distribution in an image or a region of interesting. Moreover, DCD similarity matching does not fit human perception very well, and it will cause incorrect ranks for images with similar color distribution. We will adopt a new and efficient dominant color extraction scheme to address the above problems. An enhanced algorithm to extract a DCD is proposed to provide an effective, compact, and powerful color representation for image retrieval. A DCD specifies a small number of dominant color values and the statistical properties. The structure of a DCD, F, is defined as

$$F = \{p_i, c_i, v_i, s\}, \quad i = 1, 2, \ldots, N_{DCD} \tag{14}$$

where N_{DCD} is the number of dominant colors, s is the spatial coherency value that represents the overall spatial homogeneity of the dominant colors, p_i is the percentage of pixels in the image corresponding to the ith dominant color, v_i is a vector representing the ith dominant color, and the ci is optional which is the variation of the dominant color values of the pixels around v_i. In most of applications, s is set to zero.

If H, S, and V are divided into L_H, L_S, and L_V sections, respectively, the volume of each piece is not a constant and the volume of the kth piece along S direction is $V_k = 2\Pi^2(2k-1)/L_H L_S^2 L_V$, which can be easily proved by induction. The volumes of blocks near the centre are smaller than those of outside blocks. Therefore, building histograms on the H, S, and V values is not suitable since the building blocks are not equally distributed, and a modified HSV space (MHSV) is proposed for fixing the problem. The values of X_M, Y_M, and Z_M on MHSV are calculated as follows:

$$X_M = S \cdot cos(2\Pi\ H) \tag{15 - a}$$

$$Y_M = S \cdot sin(2\Pi\ H) \tag{15 - b}$$

$$Z_M = V \tag{15-c}$$

Because $S \in [0,1]$, and cos and sin functions are continuous, the ranges for X_M and Y_M are $X_M \in [-1,1]$ and $Y_M \in [-1,1]$, respectively. A color is represented as a point in the MHSV 3-D space with its H, S, and V values as the coordinates. If X_M, Y_M, and Z_M are divided into L_X, L_Y, and L_Z sections, respectively, the volume of each piece is constant, $V = 1/L_X, L_Y, L_Z$. Because each piece has the same volume in the MHSV space, it has an equal distribution. Therefore, the new MHSV color space is more reasonable for building a histogram than the original HSV color space. The MHSV space utilizes the advantages of both the original HSV space and RGB space.

4.4. Feature extraction using Gray-Level Co-occurrence Matrix (GLCM)

GLCM [21,25] creates a matrix with the directions and distances between pixels, and then extracts meaningful statistics from the matrix as texture features. GLCM is composed of the probability value, it is defined by $P(i,j|d,\theta)$ which expresses the

probability of the couple pixels at θ direction and d interval. When θ *and d is determined,* $P(i,j|d,\theta)$ is showed by $P_{i,j}$. Distinctly GLCM is a symmetry matrix; its level is determined by the image gray-level. Elements in the matrix are computed by the equation shown below:

$$P(i,j|d,\theta) = \frac{P(i,j|d,\theta)}{\sum_i \sum_j P(i,j|d,\theta)} \tag{16}$$

GLCM expresses the texture feature according the correlation of the couple pixels gray-level at different positions. It quantificationally describes the texture feature. Four features are selected; include energy, contrast, entropy, inverse difference.

– *Contrast*: Contrast is the main diagonal near the moment of inertia, which measure the value of the matrix is distributed and images of local changes in number, reflecting the image clarity and texture of shadow depth. Contrast is large means texture is deeper. $P(x,y)$ is the gray-level value at the Coordinates (x,y). $(x–y)$ Is the differences between pairs of gray levels x and y.

$$Contrast(I) = \sum_x \sum_y (x-y)^2 P(x,y) \tag{17}$$

– *Entropy:* Entropy measures image texture randomness, when the space co-occurrence matrix for all values is equal, it achieved the minimum value; on the other hand, if the value of co-occurrence matrix is very uneven, its value is greater.

$$Entropy(S) = -\sum_x \sum_y P(x,y) \bullet Log(x,y) \tag{18}$$

– *Inverse difference moment:* It measures local changes in image texture number. Its value in large is illustrated that image texture between the different regions of the lack of change and partial very evenly. Where x, y are the coordinates of the image pixel at location (x,y), and $p(x,y)$ is the gray-level value at the Coordinate (x,y).

$$InverseDifference(H) = \sum_x \sum_y \frac{P(x,y)}{1+(x-y)^2} \tag{19}$$

4.5. Region-based vector codebook subband clustering (RBSC)

A clustering algorithm is needed to extract the dominant colors from the color histogram of the multiscale image. Based on a vector codebook generation algorithm, Region-based subband clustering (RBSC) has been applied. Choosing RBSC as the clustering algorithm for the proposed of dominant color descriptors (DCD) is due to that it is an unsupervised clustering and due to its relatively low complexity. Suppose that a histogram is $\{Hist(x), x = 1, 2, \ldots, n\}$. The histogram value of the point i is x_i and that of point j is x_j. One adjustable parameter θ is the same as the one in RBSC. The RBSC algorithm for generating directed trees is an extended version to the graph-theoretic clustering.

Shown below how to apply the RBSC algorithm for a 1-D histogram is explained as below. The steps of RBSC algorithm for generating directed trees are an enhanced version of graph-Theoretic clustering [36], as follows:

(1) Compute the distance d_{ij} between *xi*and x_j, and count the number of x_j as N_i (called the density of x_i) if d_{ij} is less than θ.

$$d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\| \tag{20 - a}$$
$$\Phi_i = \{j | d_{ij} \leqslant \theta\} \tag{20 - b}$$
$$N_i = |\Phi_i| \tag{20-c}$$

(2) Compute the index g_{ij} for $\mathbf{x}_i$ with another $\mathbf{x}_j$ by

$$g_{ij} = (N_j - N_i)/d_{ij} \tag{20-d}$$

(3) Obtain the maximum g_i at $\mathbf{x}_i$, $g_i = \max_j gij$

(4) (a) If $N_i = 1$ (no data round $\mathbf{x}_i$), $\mathbf{x}_i$ is a root of directed trees. (b) If $g_i < 0$, $\mathbf{x}_i$ is a root of directed trees. (c) If $g_i > 0$, $\mathbf{x}_j$ is the parent node of $\mathbf{x}_i$.

5. Similarity matching

An integrated image matching procedure similar to the one used in [37,38] is proposed. With the decomposition of the image, the number of sub-blocks remains same for all the images. In [6], a sub-block from query image is allowed to be matched to any subblock in the target image. However a sub-block may participate in the matching process only once. A bipartite graph of sub-blocks for the query image and the target image is built as shown in Fig. 6. The labeled edges of

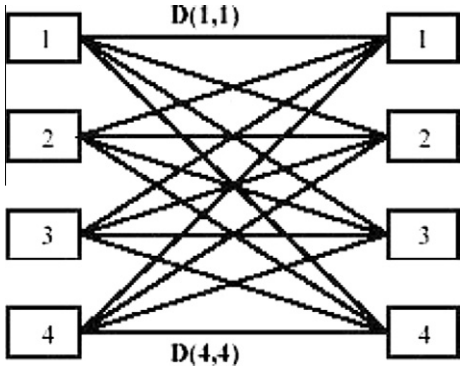

Fig. 6. Bipartite graph showing 4 sub-blocks of target and query images.

the bipartite graph indicate the distances between sub-blocks. A minimum cost matching is done for this graph. Since, this process involves too many comparisons; the method has to be implemented efficiently.

A key advantage of bipartite matching is its inherent scalability. In this paper, we present a way to model similarity matching via bipartite graphs. We model the problem of matching pairs of sub-bands locations of query and target images as a minimum weight matching problem in bipartite graphs. The bipartite matching essentially establishes one-to-one correspondences between the cells in different images. A key advantage of using bipartite matching is the inherent scalability, which arises from its polynomial time-complexity.

An algorithm is used for finding the minimum cost matching based on most Similar Highest Priority (MSHP) principle using the adjacency matrix of the bipartite graph [6]. Here in, the distance matrix is computed as an adjacency matrix. The minimum distance dij of this matrix is found between sub-blocks i of query and j of target. The distance is recorded and the row corresponding to sub-block I and column corresponding to sub-block j, are blocked. This will prevent sub-block i of query image and sub-block j of target image from further participating in the matching process. The distances, between i and other sub-blocks of target image and, the distances between j and other sub-blocks of query image, are ignored (because every sub-block is allowed to participate in the matching process only once). This process is repeated till every sub-block finds a matching.

The process is demonstrated in Fig. 6 using an example for four sub-blocks. The complexity of the matching procedure is reduced from $O(n^2)$ to $O(n)$, where n is the number of features involved. The integrated minimum cost match distance between images is now defined as $D_{qt} = \Sigma\Sigma d_{ij}$, where $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n$. d_{ij} is the best-match distance between sub-block i of query image q and sub-block j of target image t and D_{qt} is the distance between images q and t.

6. Experimental results

Experiments have been carried out to validate the efficiency of the proposed model. The experiments were carried out on a Core i3, 2.4 GHz processor with 4GB RAM using MATLAB 7.0. Comparisons with other methods have been conducted. The proposed ICTEDCT-CBIR model is compared with the CTDCIRS method [5], Jhanwar et al. [41], Hung and Dai's [40] and Rao et al. [21] methods (as will be illustrated in Table 1 and Figs. 8, 9a and b, respectively). Moreover, experimental results have also been compared with Linde-Buzo-Gray (LBG), Kekre's Proportionate Error (KPE) and Kekre's Fast Codebook Generation (KFCG) algorithms [39] (as will be illustrated in Table 2 and Figs. 11a and b). Furthermore, experiments have been conducted to compare the performance of our proposed model with the two best spectral texture features in literature, i.e., Gabor filter and wavelet filters (as will be illustrated in Fig. 12).

To compare the performance of our proposed method with the BabuRao et al. [5], experiments are conducted to explore the performance of the system on Wang's image set [42] downloaded from http://wang.ist.psu.edu/docs/related/. Image Set consists of 1000 Corel images with ground truth. These images are grouped into 10 categories with each containing 100 images of the size 256 × 384. The images in the same cluster are considered as similar images. The images are subdivided into 10 classes such that it is almost sure that a user wants to find the other images from a class if the query is from one of these 10 classes. This is a major advantage of this database because due to the given classification it is possible to evaluate retrieval results. This database was used extensively to test the different features because the size of the database and the availability of class information allows for performance evaluation [43].

We consider the curvelet-based color and texture features not only to be able to express more image information, but also to describe image from the different aspects for more detailed information in order to obtain better search results. The performance of a retrieval system can be measured in terms of its recall (or sensitivity) and precision (or specificity). Recall measures the ability of the system to retrieve all models that are relevant, while precision measures the ability of the system to retrieve only models that are relevant. They are defined as:

Table 1
Comparison among the average precision of the proposed ICTEDCT-CBIR method, the CTDCIRS method, BabuRao et al. [5]), Jhanwar et al. [41], Hung and Dai's [40] and Rao et al. [21], applied on Wang data set.

Class	Average precision				
	Jhanwar et al. [41]	Hung and Dai's [40]	CTDCIRS method [5]	Rao et al. [21]	The proposed ICTEDCT-CBIR
Africa	0.4525	0.4240	0.562	0.42	0.635
Beaches	0.3975	0.4455	0.536	0.39	0.642
Building	0.3735	0.4105	0.610	0.43	0.698
Bus	0.741	0.8515	0.893	0.65	0.915
Dinosaur	0.9145	0.5865	0.984	0.97	0.992
Elephant	0.304	0.4255	0.578	0.63	0.781
Flower	0.8515	0.8975	0.899	0.9	0.948
Horses	0.568	0.589	0.78	0.65	0.952
Mountain	0.2925	0.268	0.512	0.46	0.738
Food	0.3695	0.4265	0.684	0.52	0.806
Average	0.52645	0.5324	0.7048	0.602	0.8107

Table 2
Comparison of average precision obtained by different retrieval techniques.

Class	Average precision			
	HSV color	GLCM texture	HSV Color + GLCM texture	The proposed ICTEDCT-CBIR
Africa	0.26	0.21	0.25	0.35
Beaches	0.27	0.35	0.21	0.42
Building	0.38	0.5	0.24	0.35
Bus	0.45	0.22	0.51	0.32
Dinosaur	0.26	0.29	0.6	0.93
Elephant	0.3	0.24	0.26	0.48
Flower	0.65	0.73	0.81	0.90
Horses	0.19	0.25	0.28	0.52
Mountain	0.15	0.18	0.2	0.38
Food	0.24	0.29	0.25	0.36
Average	0.315	0.326	0.361	0.51

$$precision = \frac{Number\ of\ relevant\ images\ retrieved}{Total\ number\ of\ images\ retrieval} \tag{21 - a}$$

$$Recall = \frac{Number\ of\ relevant\ images\ retrieved}{Total\ number\ of\ relevant\ images} \tag{21-b}$$

The proposed curvelet-based image retrieval process is applied to each image in the database and index each of the images using the curvelet-based feature vector. Various test scenarios are applied, three levels analysis, four levels analysis and five levels analysis with four levels analysis, 50(=1+16+32+1) subbands of curvelet coefficients are computed, while with with 5 levels analysis, 42 (=1 + 8 + 16 + 16 + 1) subbands of curvelet coefficients are computed. However, curvelet at angle θ produces the same coefficients as curvelet at angle $(\theta + \pi)$. Therefore, half of the subbands at scale 2 and 3 are discarded due to this symmetry. As the result, for example for a four levels analysis, 26 (=1 + 8 + 16 + 1) subbands are preserved.

Fig. 7 shows some of these images. The images in the same row in Fig. 7 belong to the same cluster. This experiment used each image in each class as a query image. The experiment was carried out with the number L of retrieved images set as 20 to compute the precision P of each query image and finally obtain the average precision $P/100$ (100 images of a class).

Comparisons have been conducted between the experimental results of the proposed ICTEDCT-CBIR method and that of the CTDCIRS method (BabuRao et al. [5]) and the other two methods in literature [40,41] (are shown in Table 1). The precision and recall measurements are used to describe the performance of image retrieval. The precision (P) and recall (R) are defined as follows:

$$P(k) = n_k/L \text{ and } R(k) = n_k/N \tag{22}$$

where L is the number of retrieved images; n_k is the number of relevant images in the retrieved images and N is the number of all relevant images in the database.

Fig. 8 illustrates a Comparison of average precision obtained by proposed system with other retrieval systems.

It is obvious from the above experimental results, reported in Table 1 and Fig. 8, that the proposed ICTEDCT-CBIR method has achieved a better average precision of various images than the other four methods.

A comparison has been conducted to compare the performance of the proposed ICTEDCT–CBIR model with that of Rao et al. [21] in terms of the average precision versus number of retrieved images (as illustrated in Fig. 9) and the average recall versus number of retrieved images (as illustrated in Fig. 9b).

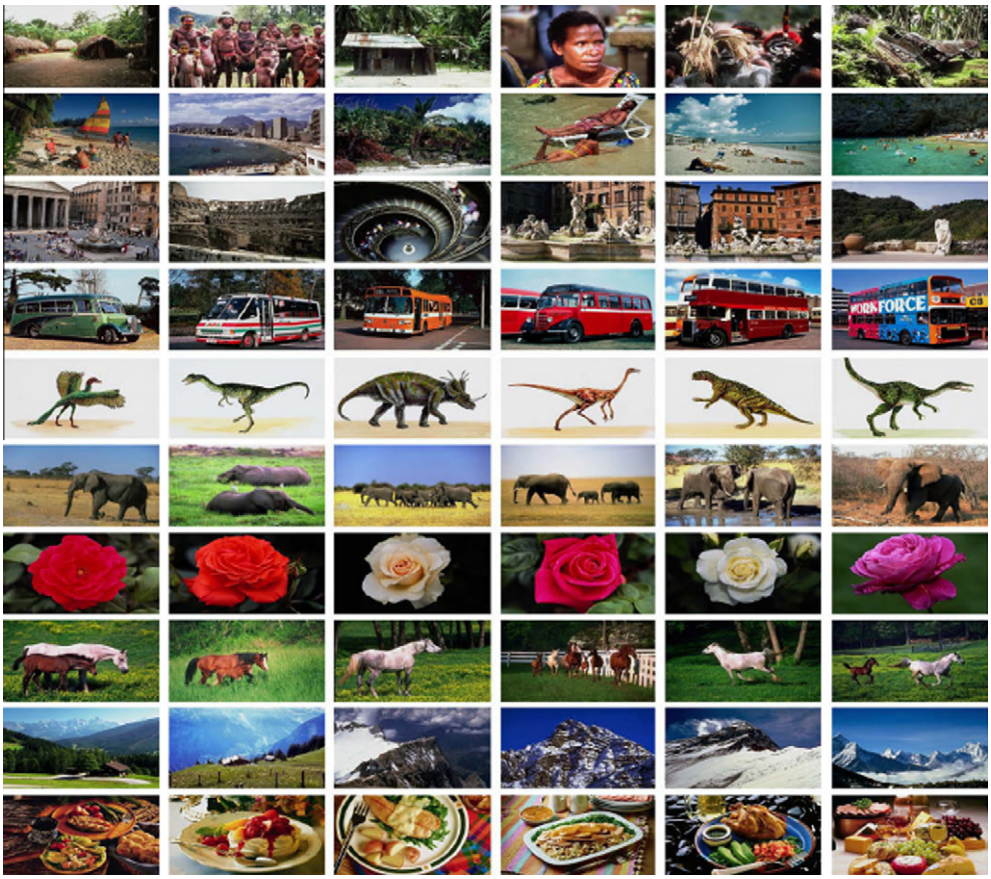

Fig. 7. Samples of the experimental Wang data set [42].

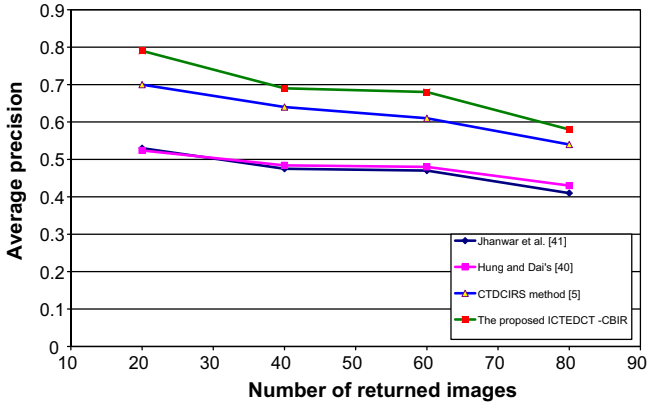

Fig. 8. The average precision versus number of returned images, for the proposed ICTEDCT-CBIR method, the CTDCIRS method, BabuRao et al. [5], Jhanwar et al. [41], Hung and Dai's [40] and Rao et al. [21], applied on Wang data set.

As observed from the experimental results illustrated in Fig. 9a and b, the average precision and recall values of the proposed ICTEDCT–CBIR model outperforms that of Rao et al. [21], where the precision remains higher for all number of returned images with average precision of 0.7. Moreover, the recall values remain higher for all returned images with average recall value of 0.57.

Moreover, experiments have also been carried out on the Columbia Object Image Library (COIL) database [44] that consists of 7200 images from 100 different categories. A sample of COIL database is illustrated in Fig. 10. To test the proposed

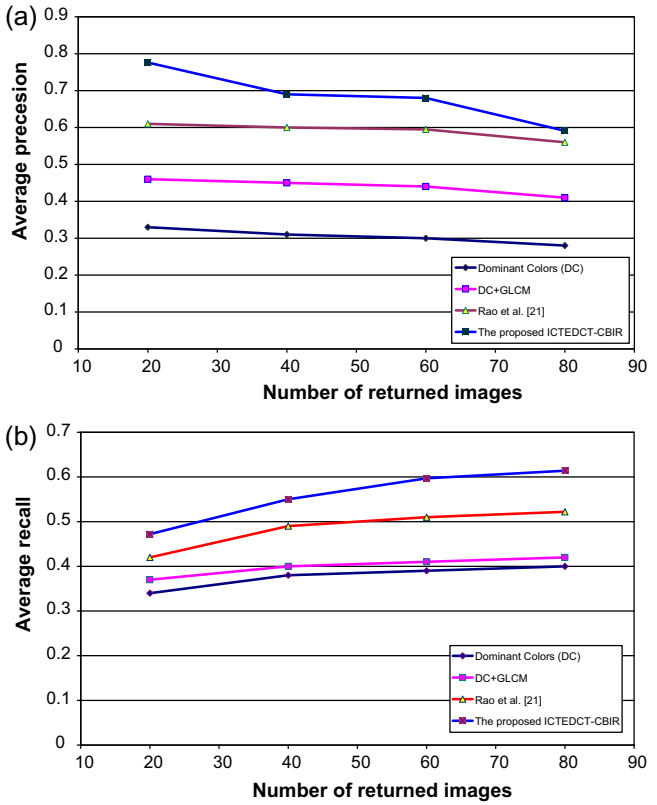

Fig. 9. (a) The average precision versus number of returned images. (b) The average recall versus number of returned images.

ICTEDCT-CBIR model, five query images are selected randomly from each class. To check the performance of proposed technique we have used precision and recall. The quantitative measure is given below

$$p(i) = \frac{1}{100} \sum_{1 \leqslant j \leqslant 1000, r(i,j) \leqslant 100, ID(j)=ID(i)} 1 \tag{23-a}$$

where $P(i)$ is precision of query image i, $ID(i)$ and $ID(j)$ are category ID of image i and j respectively, which are in the range of 1–10. The $r(i,j)$ is the rank of image j. This value is percentile of images belonging to the category of image i, in the first 100 retrieved images. The average precision p_t for category t ($1 \leqslant t \leqslant 10$) is given by

$$p(i) = \frac{1}{100} \sum_{1 \leqslant i \leqslant 1000, ID(i)=t} p(i) \tag{23-b}$$

The comparison of proposed ICTEDCT-CBIR system with other retrieval systems is also presented in the Table 2. Experimental results have been compared with Linde-Buzo-Gray (LBG), Kekre's Proportionate Error (KPE) and Kekre's Fast Codebook Generation (KFCG) algorithms [39]. These retrieval systems are based on HSV color, GLCM texture and combined HSV color and GLCM texture. Our proposed curevelet-based retrieval system is better than these systems for all categories of the database.

Fig. 11 gives the average precision and average recall values plotted against number of retrieved images for the proposed ICTEDCT-CBIR model, compared with KFCG-CBIR, KPE-CBIR, and LGB-CBIR. Here the higher precision and recall values are given by the proposed ICTEDCT-CBIR model proving the best performance. The distinction in performance of KPE-CBIR and LBG-CBIR is difficult here, which are zoomed across their crossover points of precision and recall curves in Fig. 11b.

The crossover point values of precision and recall curves for the proposed ICTEDCT-CBIR image retrieval model, LBG-CBIR, KPE-CBIR and KFCG-CBIR (with codebook size 128 × s12) are shown in Fig. 11b. The figure gives average precision and recall plotted against number of retrieved images with the precision–recall crossover point values respectively as 0.57, 0.46, 0.43, and 0.39, for the proposed ICTEDCT-CBIR, KFCG-CBIR, KPE-CBIR, and LGB-CBIR models, respectively. Here the best performance with higher precision and recall values is given by the proposed ICTEDCT-CBIR image retrieval model followed by KFCG-CBIR, then the LBG-CBIR and KPE-CBIR models. These graphs clearly show that the higher precision and recall values yield in to higher crossover point value. So the performance of image retrieval techniques can also be compared using the heights of precision and recall crossover points.

Fig. 10. Sample of Coil database

The higher precision and recall values are given by the proposed image retrieval model proving the best performance, then the KFCG-CBIR reported the 2nd best, while the distinction in performance of KPE-CBIR and LBG-CBIR are difficult which are zoomed and shown in Fig. 11b). Table 2 shows a comparison of average precision obtained by different retrieval techniques.

Fig. 11c illustrates a comparison chart of a precision versus recall values. The experimental curves compare the results of the proposed model, concerning the HSV with invariant DCD, CCM and GLCM features, compared with other feature models.

6.1. Comparison with the performance of other filters Gabor filter feature and wavelet feature

Moreover, experiments have been conducted to compare the performance of our proposed model with the two best spectral texture features in literature, i.e., Gabor filter feature and wavelet feature. For Gabor filter, we use the best combinations of parameters found by Chen et al. in [16], that is, six scales and four orientations with filter size of 13×13 pixels. The Gabor filter with this configuration generates 48 texture features for each input image, which is comparable with the 52 texture features of the four levels curvelet transform.

Fig. 12 shows a comparison between the proposed ICTEDCT-CBIR method and both L. Chen et al. Gabor Filter based retrieval method [16] and Bhagavathy et al. [19] Wavelet-based retrieval method on the same image database. The average precision values are illustrated versus average recall values. However, in all situations, curvelet feature gives good results and outperforms Gabor filter feature significantly. As can be seen from Fig. 12, the retrieval performance of curvelet feature is significantly higher than that of Gabor filter retrieval and wavelet-based retrieval. The curvelet transform with four levels decomposition has even better retrieval performance than at three levels decomposition, however, at the cost of slightly higher feature dimension. Wavelet produces an average retrieval precision of 68.40%, while the proposed ICTEDCT-CBIR method gives an average retrieval precision of 87.50% with three scale decomposition and an average retrieval precision of 89.8% with four scale decomposition.

The reasons why cuvelet feature has significantly higher retrieval performance than Gabor filter feature are due to curvelet has several advantages over Gabor filter. First, due to the half-height truncation of the filters to remove the overlap in spectral omain, many holes are created in the spectrum, that is, the spectrum is not completely covered by Gabor filters [16]. As the result, much of the frequency information in an image is lost due to the incomplete spectrum cover. Secondly, curvelet transform uses ridgelets as the filter bank, while Gabor uses wavelet as the filter bank. Compared to [18,19], Ridglets capture edge information or texture information more accurately than wavelets. Moreover, curvelet transform is a true wavelet transform, while Gabor filter is not because it does not scale image during the filtering. Due to the scaling of image size

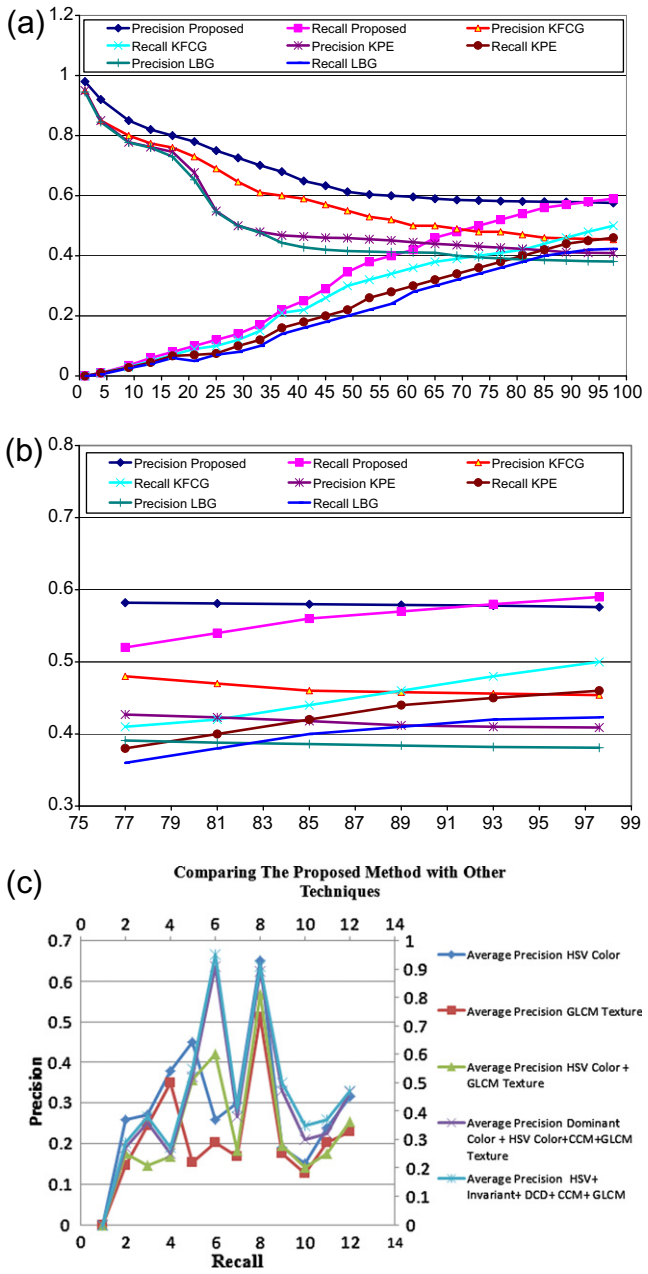

Fig. 11. (a) Average precision and recall plotted against number of retrieved images for the proposed ICTEDCT-CBIR model, compared with KFCG-CBIR, KPE-CBIR, and LGB-CBIR. (b) Crossover point values of average precision and average recall plotted against number of retrieved images for the proposed ICTEDCT-CBIR model, KFCG-CBIR, KPE-CBIR, and LBG-CBIR of test set Images (with Codebook size 128×12). (c) Features comparison chart with the proposed method.

at different levels of transform to adapt the image to the curvelets, curvelet can tolerate more scale distortions in images. This can be seen in all the examples in Fig. 12.

The reasons why curvelet has better retrieval performance than wavelet are in two aspects. First, as mentioned earlier, curvelets capture more accurate edge information or texture information than wavelets. Second, as curvelets are tuned to different orientations, curvelets capture more directional features and more accurate directional features than wavelets.

6.2. Complexity

In our feature extraction step, in consideration of computational complexity, not all levels of curvelet coefficients are used. Only level 2 and level 5 sub-bands coefficients are selected for feature extraction in a 6 levels decomposition of images

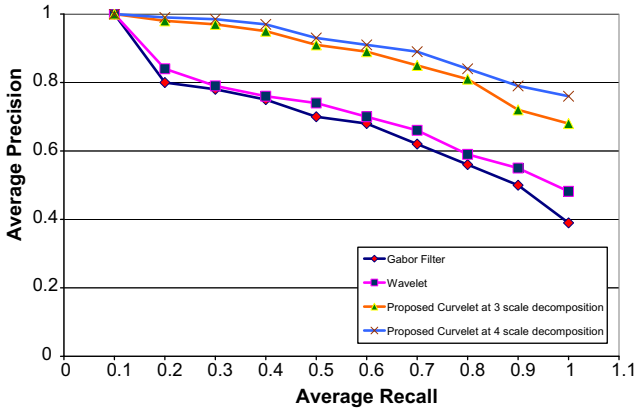

Fig. 12. Comparison among the average retrieval precision of 1000 queries versus recall using the proposed curvelet-based retrieval at scale three decomposition and scale four decomposition, Bhagavathy et al. [19], Wavelet-based retrieval, and Chen et al. using Gabor Filter [16].

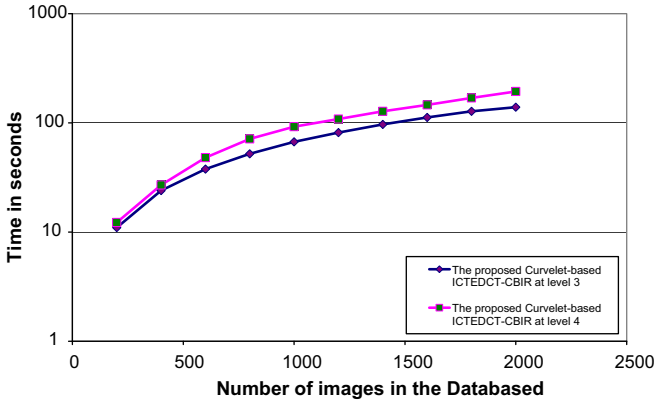

Fig. 13. The average retrieval time (in seconds) for the proposed curvelet-based ICTEDCT-CBIR method versus vaious data set sizes.

with size 512×512. The reason for this selection is level 2 coefficients are in 'finer' details of an image, they can be used as the texture description of the image, and coefficients in level 5 on the contrary, they are in a 'coarser' level, they are fittable for description of the edges in an image. The selection is appropriate since level 2 describe the image in a coarser view and the level 5 coefficients are detail descriptions. Once the curvelet coefficients have been obtained, texture features are computed for the curvelet. For each curvelet, γ texture features are obtained and ξ dominant color are obtained. If *n*curvelets are used for the transform, $(\gamma + \xi)$ features are obtained. This feature extraction is applied to each of the images in the database. At the end, each image in the database is represented and indexed using its curvelet feature vector.

This feature extraction is applied to each of the images in the database. Given that K^2 is the number of pixels in each image, the curvelet process is in order of $O(K^2 \log(K))$. Given a query, its curvelet-based texture and dominant color features are computed. These curvelet-based texture and dominant color features are computed to each image in the database as well. At the end, each image in the database is represented and indexed using its curvelet-based feature vector. The system then compares the query feature vector with all the feature vectors in the database. Corresponding, if number of images in the database is N, then the time complexity of the retrieval process is in order of $O(N.K^2 \log(K) + (\gamma + \xi).N)$. Fig. 13 illustrates the Retrieval time in seconds for different Data set sizes. The retrieval time is greatly improved if parallel processing in applied.

Experimental results showed that the proposed method yielded higher average precision and average recall with reduced feature vector dimension than other techniques. In addition, the proposed method almost always showed performance gain of average retrieval time over the other methods.

7. Conclusions

In this paper, a novel Integrated Curvelet-based image retrieval scheme (ICTEDCT-CBIR) has been introduced, using Enhanced Dominant Color features and Texture analysis. The first contribution of the paper is the efficient integration of Curv-

elet Multiscale ridgelets of spatial band pass filtering with Region-based vector codebook Subband Clustering (RBSC) for dominant colors extraction and texture analysis. The second contribution is the design of an enhanced Region-based vector codebook Subband Clustering (RBSC) for efficient dominant colours extraction. Higher retrieval performance has been illustrated, where more accurate texture and directional features have been captured. Results showed that the proposed model is very promising for image retrieval, and significantly outperforms other models in literature. The higher precision and recall values given by the proposed scheme proving the best performance. Our experimental results demonstrated that the proposed method has higher average precision and retrieval performance, compared with other Image retrieval models, applied on Wang's images set and Columbia Object Image Library (COIL). Retrieval performance of curvelet feature is significantly higher than that of Gabor-based filter retrieval and wavelet-based retrieval. In addition, the proposed method almost showed performance gain of average retrieval time over the other methods.

References

[1] Kekre HB, Sudeep DT, Athawale A, Shah A, Verlekar P, Shirke S. Walsh transform over row mean and column mean using image fragmentation and energy compaction for image retrieval. Int J Comput Sci Eng (IJCSE) 2010;2S(1). <www.enggjournals.com/ijcse>.
[2] Min Rui, Cheng HD. Effective image retrieval using dominant color descriptor and fuzzy support vector machine. Pattern Recog 2009;42(1):147–57.
[3] Kekre HB, Thepade SD, Parkar A. A comparison of Kekre's fast search and exhaustive search for various grid sizes used for colouring a greyscale image. In: International conference on signal acquisition and processing, 2010. ICSAP'10, 2010. p. 53–7.
[4] Kekre DrHB, Thepade Sudeep D, Athawale Archana, Shah Anant, Verlekar Prathamesh, Shirke Suraj. energy compaction and image splitting for image retrieval using kekre transform over row and column feature vectors. IJCSNS Int J Comput Sci Network Secur 2010;10(1):289–98.
[5] BabuRao M, Rao BP, Govardhan A. CTDCIRS: content based image retrieval system based on dominant color and texture features. Int J Comput Appl (IJCSs) 2011;18(6):0975–8887.
[6] Kavitha C, Rao BP, Govardhan A. Image retrieval based on color and texture features of the image sub-blocks. Int J Eng Sci Technol (IJEST) 2011;3(2):1060–8.
[7] Zhang Jun, Ye Lei. Series feature aggregation for content-based image retrieval. J Comput Electr Eng 2010;36(4):691–701.
[8] Tamura H, Mori S, Yamawaki T. Texture features corresponding to visual perception. IEEE Trans Syst, Man Cybern 1978;8(6):460–73.
[9] Manjunath BS et al. Color and texture descriptors. IEEE Trans CSVT 2001;11(6):703–15.
[10] Liu F, Picard RW. Periodicity, directionality, and randomness: wold features for image modeling and retrieval. IEEE Trans Pattern Anal Mach Intell 1996;18(7):722–33.
[11] Huang Jing, Kumar S Ravi, Mitra Mandar, Zhu Wei-Jing, Zabi Ramin. Image indexing using color correlograms. In: Computer vision and pattern recognition, 1997. Proceedings, 1997 IEEE computer society conference on 17-19 June, 1997.
[12] Engel J. The multiresolution histogram. Metrika 1997;46:41–57.
[13] Hadidemetriou E, Grossberg MD, Nayar SK. Multiresolution histograms and their use for recognition. IEEE Trans Pattern Anal Mach Intell 2004;26(7):831–47.
[14] Starck J, Candès EJ, Donoho DL. The curvelet transform for image denoising. IEEE Trans Image Process 2002;11(6):670–84.
[15] Candes E et al. Fast discrete curvelet transforms. Multiscale Model Simul 2006;5(3):861–99.
[16] Chen L, Lu G, Zhang DS. Effects of different gabor filter parameters on image retrieval by texture. In: Proceedings of IEEE 10th international conference on multimedia modeling; 2004. p. 273–8.
[17] Manjunath BS, Ma WY. Texture features for browsing and retrieval of large image data. IEEE Trans Pattern Anal Mach Intell 1996;18(8):837–42.
[18] Suematsu N, et al. Region-based image retrieval using wavelet transform. In: Proceedings of the 15th international conference on vision interface; May, 2002.
[19] Bhagavathy S, Chhabra K. A wavelet-based image retrieval system. Technical report—ECE278A, Santa Barbara: Vision Research Laboratory, University of California; 2007.
[20] Palm C, Keysers D, Lehmann T, Spitzer K. Gabor filtering of complex hue/saturation images for color texture classification. In: Proceedings of the international conference on computer vision. (vol. 2), Atlantic City (NJ): Pattern Recognition, and Image Processing; 2000. p. 45–9.
[21] Babu Rao M, Prabhakara Rao B, Govardhan A. Content based image retrieval system based on dominant color, texture and shape. Int J Eng Sci Technol (IJEST) 2011;3(4):2887–96.
[22] Hatamzadeh-Tabrizi J, Funnell WRJ. Comparison of gradient, gradient vector flow and pressure force for image segmentation using active contours. In: Proc 27th ann conf can med biol eng soc (CD-ROM); 2002.
[23] Candès Emmanuel J. Monoscale ridgelets for the representation of images with edges. Tech. rep., Stanford (CA): Dept. Statist., Stanford University; 1999.
[24] Candès EJ, Donoho DL. Curvelets; 1999. <http://www-stat.stanford.edu/~donoho/Reports/1999/curvelets.pdf>.
[25] Kong F. Image retrieval using both color and texture features. In: International conference on machine learning and cybernetics, baoding, vol. 4; 2009. p. 2228–32.
[26] Mailing S, Huan L. An image retrieval technology based on HSV color space. Comput Knowl Technol 2007;3:200–1.
[27] Smith JR, Chang FS. Tools and techniques for color image retrieval. In: International conference on storage and retrieval for image and video databases (SPIE), (SPIE)'1996; 1996. p. 426–37.
[28] Ji-quan Ma. Content-based image retrieval with HSV color space and texture features. In: International conference on web information systems and mining (WISM 2009); 2009. p. 61–3.
[29] Ni L, Leng HC. Curvelet transform and its application in image retrieval. In: 3rd International symposium on multispectral image processing and pattern recognition, Proceedings of SPIE, vol. 5286; 2003.
[30] Donoho DL, Duncan MR. Digital curvelet transform: strategy, implementation and experiments. Proc SPIE 2000;4056:12–29.
[31] Carsten Høilund. The radon transform. Technical report, Aalborg University, VGIS; 07gr721, November 12, 2007.
[32] Geback T, Koumoutsakos P. Edge detection in microscopy images using curvelets. BMC Bioinform 2009;10(75):1–14.
[33] Shensa MJ. Discrete wavelet transforms: wedding the a trous and mallat algorithms. IEEE Trans Signal Process 1992;40:464–82.
[34] Turgay Çelik, Tardi Tjahjadi. Multiscale texture classification and retrieval based on magnitude and phase features of complex wavelet subbands. Comput Electr Eng 2011;37(5):729–43.
[35] Yang N, Chang W, Kuo C, Hsing Li T. A fast MPEG-7 dominant color extraction with new similarity measure for image retrieval. J Vis Commun Image Represent 2008;19(2):92–105.
[36] Koontz WLG, Narendra PM, Fukunaga K. A graph-theoretic approach to nonparametric cluster analysis. IEEE Trans Comput 1976;C-25:936–44.
[37] Li J, Wang JZ, Wiederhold G. IRM: integrated region matching for image retrieval. In: Proc. of the 8th ACM int. conf. on multimedia; 2000. p. 147–56.
[38] Hiremath PS, Pujari Jagadeesh. Content based image retrieval based on color, texture and shape features using image and its complement. Int J Comput Sci Secur 2007;1(4):25–35.
[39] Kekre HB, Sarode TK, Thepade SD. Image retrieval using color-texture features from DCT on VQ codevectors obtained by Kekre's fast codebook generation. ICGST-Int J Graphics, Vis Image Process (GVIP) 2009;9(5):1–8. <http://www.icgst.com/gvip/Volume9/Issue5/P1150921752.html>.
[40] Huang PW, Dai SK. Image retrieval by texture similarity. Pattern Recogn 2003;36(3):665–79.

[41] Jhanwar N, Chaudhuri S, Seetharaman G, Zavidovique B. Content based image retrieval using motif co-occurrence matrix. Image Vis Comput 2004;22:1211–20.
[42] http://wang.ist.psu.edu/docs/related/Image.orig, September 2008. <http://wang.ist.psu.edu/>.
[43] Deselaers T, Ney H, Seidl T, Keysers D. Features for Image Retrieval, Dessertation in content-based image retrieval, Aachen; 2003.
[44] Nene S, Nayar S, Murase H. Columbia object image library (COIL-100). Technical report, CUCS-006-96, Feb-1996. <http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php>.

Sherin M. Youssef received her PhD degree from University of Nottingham (UK, 2004) in Intelligent distributed swarm intelligence and optimization systems. She received her Master degree (MSc) from University of Alexandria (Egypt, 1995) in Pattern recognition and machine intelligence. She is currently an associate professor and a senior lecturer in the department of computer engineering, college of engineering, Arab Academy for science and Technology (AAST). **Her research interests** lie in the areas of Artificial intelligence, Intelligent agent based systems, digital signal processing, image processing, Video surveillance systems, Image retrieval systems, network security, Multimedia systems, and biomedical engineering.