Multimedia Tools and Applications
https://doi.org/10.1007/s11042-019-08029-7

Image Retrieval by Integrating Global Correlation of Color and Intensity Histograms with Local Texture Features

Suresh Kumar Kanaparthi [1] · U. S. N. Raju [1] · P. Shanmukhi [2] · G. Khyathi Aneesha [2] · Mohammed Ehsan Ur Rahman [3]

Received: 9 September 2018 / Revised: 8 June 2019 / Accepted: 19 July 2019
Published online: 20 August 2019

Abstract
Research on Content-Based Image Retrieval is being done to improvise existing methods. Most of the techniques that were proposed use color and texture features independently. In this paper, to get the correspondence between color and texture, we use congruence on Hue, Saturation, and Intensity by using inter-channel voting. Gray Level Co-occurrence Matrix (GLCM) on Diagonally Symmetric Pattern is computed to get texture features of an image. The similarity metrics between two images is computed using congruence and GLCM. To measure the performance; Average Precision Rate (APR), Average Recall Rate (ARR), F-measure, Average Normalized Modified Retrieval Rank (ANMRR) are calculated. In addition to these parameters, one more parameter has been proposed: Total Minimum Retrieval Epoch (TMRE) to calculate the average number of images to be traversed for each query image to get all the images of that category. To validate the performance of the proposed method, it has been applied to six image databases: Corel-1 K, Corel-5 K, Corel-10 K, VisTex, STex, and Color Brodatz. The results of most of the databases show significant improvement.

Keywords CBIR · Inter-Channel Voting · Total Minimum Retrieval Epoch · Diagonally Symmetric Pattern · Color Auto Correlogram

1 Introduction

The exponential growth of digital libraries due to mobile phones and digital cameras has resulted in enormous size of image database. Maintaining this huge database is an extremely monotonous and unmanageable task. So, an effective technique needs to be developed, which can automatically retrieve the required image from the database. Content-based image retrieval (CBIR) is one of the widely accepted solutions to accomplish this task [43]. “Content-Based”

✉ U. S. N. Raju
usnraju@nitw.ac.in

Extended author information available on the last page of the article

signifies that searching will be performed based on the analysis of genuine contents of an image. The term "content" in this context might refer to features that can be extracted from that image itself. Content-based image retrieval relies on similarity between two images, which can be measured by comparing the feature vector of corresponding images. Features of an image can be extracted from color information, local and global texture information, shape information, the orientation of the object, etc. To extract color features, methods like color histogram [41], Color Correlogram [41], color autocorrelogram [5, 16], color histogram with inter channel voting [3], etc. are being used. For extracting texture features, methods like Local Binary Pattern (LBP) [33], Uniform Local Binary Pattern (ULBP) [33], Center-Symmetric Local Binary Patterns (CS_LBP) [13], Local Extrema Pattern LEP [48], Diagonally Symmetric Pattern (DSP) [3], Local Derivative Pattern (LDP) [53], Local Tetra Pattern (LTrP) [30] are being used. For extracting shape information, Histogram of Oriented Gradients (HOG) [14], angular pattern and binary angular pattern [15], Fourier descriptor [34], Wavelet Fourier descriptor [34], Convex Hull [29], etc. are being used.

1.1 Color models

A color model is a mathematical way of describing the primary colors, whereas the color space is the total colors obtainable from the given primary colors. From the perspective of computer vision and image processing, these two terms along with color system are interchangeably used. Color space is mainly used to specify a color in a standard way easily. A color space specifies coordinate system and mathematical space within which a single point represents every single color [11]. MPEG-7 specifies four color spaces– RGB, YCbCr, HSV, HMMD [32].

In RGB color space, each color is represented as a combination of primary components of RGB color space, which are red, green, and blue. As shown in Fig. 1, this model is based on a Cartesian coordinate system. It is represented by a color cube where the primary colors: red, green and blue at three corners while the secondary colors: cyan, magenta and yellow are located at other three corners. Color black is located at the origin of the color cube whereas color white is located at the

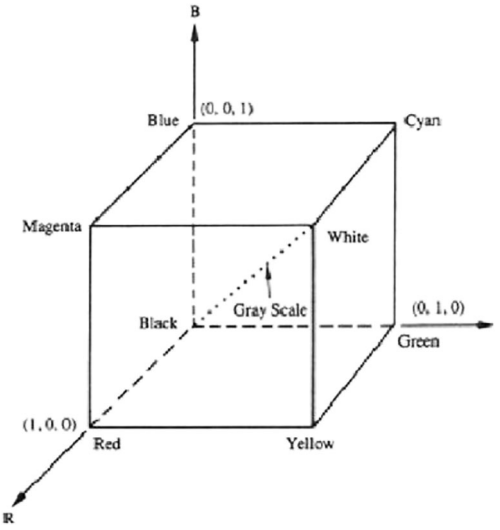

Fig. 1 RGB Color Cube

fartherest corner from the origin. In RGB color space, each color is represented by a combination of red, green and blue colors, which is located on or inside the cube.

Cyan, magenta and yellow are secondary colors of light or primary color of pigments. So, all the colors can be represented as a combination of these three colors also. This color space is known as the CMY color model, which can be represented by eq. (1):

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = 1 - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1}$$

An equal amount of these three colors should produce black color. However, in practice, combining these three colors will produce muddy-looking black. So, to produce pure black, we need to add black as fourth color. This color space is known as CMYK color space.

Among the precedent MPEG standards, YCbCr is an inheritance color space. The relationship between RGB and YCbCr color spaces are given in eq. (2) to eq. (4) [32]:

$$Y = 0.299 \times R + 0.587 \times G + 0.114 \times B \tag{2}$$

$$Cb = -0.169 \times R - 0.331 \times G + 0.500 \times B \tag{3}$$

$$Cr = 0.500 \times R - 0.419 \times G - 0.081 \times B \tag{4}$$

The above mentioned color spaces are useful for creating colors. So they are suitable for hardware implementation and not to describe a color as human interpretation. Humans describes a color objects by its hue, saturation, and brightness, Gorerge et al. [17] proposed a new color space named as HSI or HSV color space. Hue (H) component is represented by the angle from 0^0 to 360^0 which specifies the most dominant wavelength in a mixture of light. The purity of color is specified by Saturation component having a range from 0 to 1; pure yellow, blue, red, green and so on. The brightness of the color is specified by value component which also has a range from 0 to 1; The relationship between RGB and HSI color spaces are shown in eq. (5) to eq. (8) when Red is considered as the reference axis. Similarly, we can represent H, S and I values by considering Blue as well as green also as reference axis.

$$H = \begin{cases} \theta, & B \leq G \\ 360 - \theta, & B > G \end{cases} \tag{5}$$

$$\theta = cos^{-1}\left\{ \frac{\frac{1}{2}[(R-G) + (R-B)]}{\left[(R-G)^2 + (R-B)(G-B)\right]^{1/2}} \right\} \tag{6}$$

$$S = 1 - \frac{3}{(R+G+B)}[\min(R, G, B)] \tag{7}$$

$$I = \frac{1}{3}(R+G+B) \tag{8}$$

Ohm et al. proposed one new color space named HMMD (Hue-Max-Min-Diff) color space [32]. HMMD color space is more close to a perceptually uniform color space. The last three components, which are "Max", "Min" and "Diff" have transform equations between RGB and HMMD as given in eq. (9) to eq. (11).

$$Max = \max(R, G, B) \tag{9}$$

$$Min = \min(R, G, B) \tag{10}$$

$$Diff = Max - Min \tag{11}$$

Though the name of the color space (HMMD) suggests that there should be four components to represent the color space, one additional component named '*sum*' can be defined as given in eq. (12).

$$Sum = \frac{(Max + Min)}{2} \tag{12}$$

Even though this color space defines a total of five components, it can be properly represented using only three of them (H, Max, Min) or (H, Diff, Sum). Hue has the same significance as discussed in the HSI color space. Quantity of black color inside one color is specified by Max component having a range between 0 to 1. Max has the same relation with the RGB model as intensity in HIS model. Quantity of white color inside one color is specified by Min component having a range between 0 to 1. The purity of color is specified by Diff component having a range from 0 to 1. Finally, the brightness of the color is specified by Sum component, which also has the same range.

Kerke et al. proposed a new color space known as LUV color space, which is very useful for Content based Image Retrieval [19]. While L gives luminance, U and V give chromaticity values of the color image. A negative value of U indicates prominence of the red component in color image while negative value of V indicates prominence of the green component over blue. The relationship between RGB space and LUV is given by eq. (13):

$$\begin{bmatrix} L \\ U \\ V \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ -2 & 1 & 1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{13}$$

The International Commission of Illumination (CIE) came up with one new color space named **CIELAB color space** (also known as **CIE L*a*b***) in 1976 [7, 46]. In this color space, each color can be represented using three components, which are numerical values, L^* for lightness and a^* and b^* for the green-red and blue-yellow color components. With respect to human color vision, this color space is perceptually uniform. As this space itself is a three-dimensional real number space, infinitely many colors can be represented using this color space. In a real scenario, for digital representation, space can be mapped onto a three-dimensional integer space, and thus the values of L^*, a^*, and b^* are usually absolute, having a pre-defined range. As L^* *specifies brightness of a color,* $L^* = 0$ represents the darkest black, and $L^* = 100$ the brightest white. The color channels, a^* and b^*, represent true neutral gray values at $a^* = 0$ and $b^* = 0$. The green-red component is specified by a^* axis, where the negative direction specifies

the green value, and the positive direction specifies a red value. The blue-yellow component is specified by b^* axis, where the negative direction specifies the blue value, and the positive direction specifies a yellow value.

Jesus Chamorro-Martinez et al. introduced a new color space named fuzzy color space [28]. A fuzzy color C is a linguistic label whose semantics is represented as a normal fuzzy subset of colors. A fuzzy color space is a crisp set of fuzzy colors. In real applications, it is usual to work with different color terms, whose number and design depend on the application itself. In this case, the concept of fuzzy color space is useful.

1.2 Color histogram, correlogram, and autocorrelogram

The color histogram of a given image specifies the distribution of the composition of colors in the image. It is used to know different types of colors present inside an image and how many pixels are there in each type of colors. A color histogram for a given image is represented by a vector, as shown in eq. (14).

$$H = \{H[0], H[1], H[2], \ldots\ldots\ldots, H[i], \ldots . H[n]\} \tag{14}$$

Where i is the color bin in the color histogram and H[i] represents the number of pixels of color i in the image, and n is the total number of bins used in color histogram. The color histogram can be built for any color space, although the term is more often used for three-dimensional spaces like RGB or HSV.

The original idea to use the histogram for retrieval comes from Swain et al. [45], who realized that the power to identify an object using color is much larger than that of a grayscale. Given a discrete color space defined by some color axes (e.g., red, green, blue), the color histogram is obtained by discretizing the image colors and counting the number of times each discrete color occurs in the image array [42].

Strat [44] has matched cumulative histograms with a matching algorithm similar to histogram intersection to make a system that is robust to lighting changes. The color histogram of a 3-dimensional color image *img* of size M×N × ;3 can be calculated using eq. (15).

$$C\_H(i,j,k) = \sum_{m=1}^{M}\sum_{n=1}^{N} 1-\left[\left[\frac{1}{3}\times\left(\frac{|img\_1(m,n)-i|}{img\_1_{max}}\right)+\frac{1}{3}\times\left(\frac{|img\_2(m,n)-j|}{img\_2_{max}}\right)+\frac{1}{3}\times\left(\frac{|img\_3(m,n)-k|}{img\_3_{max}}\right)\right]\right] \forall i, \forall j, \forall k \in [0,255] \tag{15}$$

Where, $img\ \_\ 1(m, n)$, $img\ \_\ 2(m, n)$, $img\ \_\ 3(m, n)$ are pixel values of position (m, n) in 1st,2nd and 3rd component of image *img*. and $img\ \_\ 1_{max}$, $img\ \_\ 2_{max}$, $img\ \_\ 3_{max}$ are maximum pixel values of 1st, 2nd and 3rd components of image *img*. $C\ \_\ H$ represents the color histogram of a color image, *img*. $C\ \_\ H(i, j, k)$ is the number of occurrence where 1st component has pixel value 'i', 2nd component has pixel value 'j', and 3rd component has pixel value 'k'.

In both RGB or HIS, if the eq. (15) is followed to calculate color histogram, the number of the histogram bins will be (256 × 256 × 256) or 2^{24}. To reduce the number of bins, color quantization can be applied to each color channel, thereby reducing storage space and enhancing the processing speed [49]. So if each channel is quantized into eight levels instead of 256, the total number of histogram bins will be (8 × 8 × 8) $=2^9$. To further reduce this number Murala et al. [3] used a concept, which states that the Color distribution of an image can be represented by a histogram of each color channel in the respective color model [41]. In

the RGB color model, to get a color histogram, the image is separated into three color channels: Red (R), Green (G) and Blue (B). Then quantization is applied on each channel. Next, eq. (16) is applied to each of R, G and B channels separately; here, p is the number of bits used to represent the pixel. After that, all three histograms are concatenated to get the final color histogram. So, if each channel is quantized into eight channels the total number of histogram bins will be (8 + 8 + 8) =24.

$$H(img) = \sum_{m=1}^{M}\sum_{n=1}^{N}\left\lceil\frac{|img(m,n)-L|}{L_{max}}\right\rceil; L\in\left[0, 2^{P}-1\right] \tag{16}$$

Lin et al. [23] proposed a color-histogram based image retrieval system. They proposed three types of features, based on color, texture and color distribution, as color co-occurrence matrix (CCM), the difference between pixels of scan pattern (DBPSP) and color histogram for K-mean (CHKM) respectively. Konstantinidis et al. [20] proposed a new Image retrieval method based on fuzzy color histogram distribution.

1.2.1 Color correlogram

The color correlogram [16] was proposed for image indexing and comparison. This feature is used to characterize not only the color distribution of pixel values but also the spatial correlation of pixel pairs, which depends on distance k. As suggested in [16], we use D_8 distance. If we have two pixels $I_1(x_1, y_1)$ and $I_2(x_2, y_2)$, then D_8 distance between them is defined as given in eq. (17)

$$|I_1(x_1,y_1)-I_2(x_2,y_2)|_{D_8} = MAX\{|x_1-x_2|, |y_1-y_2|\} \tag{17}$$

The color correlogram is represented by a square matrix $\mathbb{C}^k$, of dimension N × N, where the value of each cell of matrix $\mathbb{C}(i, j)$ describes the probability of co-occurrence between a pixel having value/color 'i' and another pixel having value/color 'j' at fixed D_8 distance k. Color correlogram of an image I is given by eq. (18).

$$\mathbb{C}^k(i,j) = \frac{1}{N_i \times 8k}\sum_{m=1}^{M}\sum_{n=1}^{N} 1-\left\lceil\frac{1}{2}\left(\frac{|I(m,n)-i|}{L_{max}}\right)+\frac{1}{2}\left(\frac{|I(m+\Delta x, n+\Delta y)-j|}{L_{max}}\right)\right\rceil \forall(\Delta x,\Delta y): \Delta x,\Delta y\in\{0,1,2.....k\} \quad and\ \max(\Delta x,\Delta y)=k; \forall(i,j)\in\{0,1,.....L_{max}\} \tag{18}$$

Where N_i is histogram of color 'i', which is given by eq. (19).

$$N_i = \sum_{m=1}^{M}\sum_{n=1}^{N} 1-\left\lceil\frac{|I(m,n)-l|}{L_{max}}\right\rceil \tag{19}$$

1.2.2 Color autocorrelogram

As Color correlogram considers all the possible color pair combination, the length of the feature vector will be very large. To reduce feature vector length, color autocorrelogram is often used [5, 16]. The color autocorrelogram describes the spatial correlation between *identical colors* only. The color autocorrelogram α^k is a one dimension matrix, of size 1 × N, where the value of each cell $\alpha^k(l)$ is given as the probability of finding two pixels both

a

1	2	3	0	1
2	3	2	0	3
2	2	3	1	3
1	0	3	1	1
2	3	1	0	0

b

0.4792	0.2917	0.1667	0.0625
0.5833	0.0833	0.1667	0.1667
0.4464	0.2143	0.2143	0.1250
0.1250	0.5000	0.2917	0.0833

c

0.4792	0.0833	0.2143	0.0833

Fig. 2 (a) Original Image. (b) Result of color correlogram (c) Result of color autocorrelogram

having color '*l*' at D_8 distance *k*, comprising diagonal elements of color correlogram matrix $\mathbb{C}^k$. Color autocorrelogram of an image I is given by eq. (20). Figure 2 shows the process of calculating the correlogram and autocorrelogram for a given image.

$$\alpha^k(l) = \mathbb{C}(i,j), \forall (i,j,l) \in \{0,1,2 \ldots\ldots L_{max}\} and\ i = j = l \tag{20}$$

1.3 Texture features

Grey Level Co-occurrence Matrix (GLCM) is also the feature vector to get texture information, and it results in a matrix which gives information about the co-occurrence of pixel pairs in an image [6, 12]. Later GLCM was extended to single and multi-channel co-occurrence matrix for LUV and RGB color spaces.

The elements of GLCM may now be considered probabilities of finding the relationship *i*, *j* (or *j*, *i*) in I. The whole process is given in eq. (21).

$$GLCM_{(\Delta x,\Delta y)}(i,j) = \sum_{m=1}^{M}\sum_{n=1}^{N} 1 - \left[\frac{1}{2}\left(\frac{|I(m,n)-i|}{L_{max}}\right) + \frac{1}{2}\left(\frac{|I(m+\Delta x, n+\Delta y)-j|}{L_{max}}\right)\right] \tag{21}$$

Where (Δx, Δy) is the offset in x and y directions, respectively. When (Δx, Δy) is (1,0), which is nothing but the co-occurrence matrix for distance 1 and orientation 0^0, and the result is given in Fig. 1(b). If we want the co-occurrence matrix for a distance of 2 and orientation of 45^0, then (Δx, Δy) is (+2, −2) and the result is shown in Fig. 3(c).

Different local patterns such as LBP, ULBP, CS_LBP, LEP, and DSP are used to extract texture information from an image. The summary of these patterns is shown in Fig. 4(b-f).

a

1	2	3	0	1
2	3	2	0	3
2	2	3	1	3
1	0	3	1	1
2	3	1	0	0

b

2	2	2	0
2	0	2	1
0	3	1	2
1	2	0	0

c

1	1	0	1
1	1	1	0
1	2	0	0
0	0	0	0

Fig. 3 (a) Original Image. (b)Co-occurrence matrix when (Δx, Δy) is (1,0) (c) Co-occurrence matrix when (Δx, Δy) is(+2,-2)

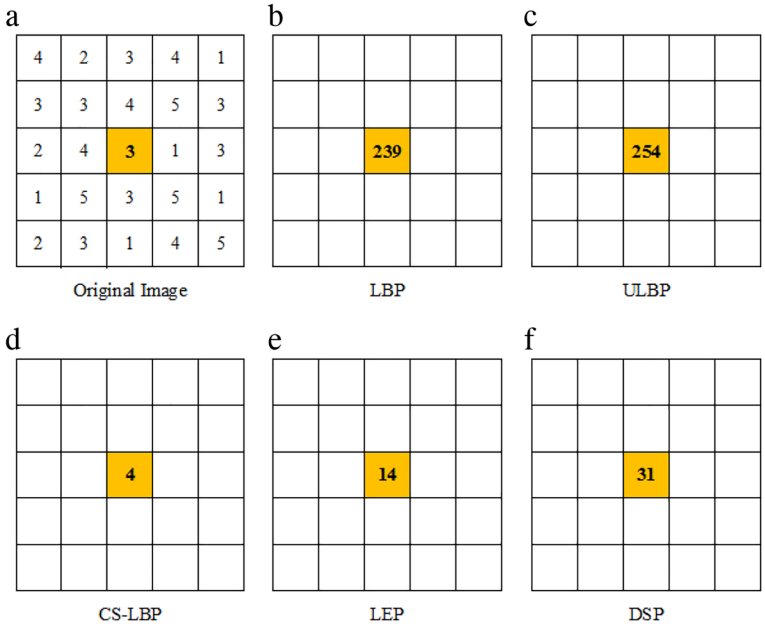

Fig. 4 (a) Original Image (b) LBP Result (c)ULBP Result (d)CS_LBP Result (e)LEP Result (f)DSP Result

1.4 Framework of CBIR

The CBIR framework proposed here follows a similar pattern as found in the traditional framework. In a general context, it includes the following stages in the sequence mentioned below.

1. Feature vector extraction, storage, and compression if required of the images in the database.
2. Feature vector extraction, storage, and compression if required of the query image.
3. Feature vector comparison.
4. Rank matrix calculation based on the output of distance calculation (similarity metrics).
5. Obtaining results using evaluation metrics such as Precision, Recall, F-Measure, Minimum Relevant Epoch, and Normalized Modified Retrieval Rank.

CBIR can be used in many real-life application like architectural and engineering design, art collections, crime prevention, geographical and remote sensing systems, intellectual property, medical diagnosis, defence, photograph archives, retail catalogs, face finding, and textile industry. Jose Ramos et al. proposed a new method, where CBIR is used as an application to interstitial lung diseases [37]. In that paper, they used CBIR as a search technology that could aid medical diagnosis by retrieving and presenting earlier reported cases that are related to the individual being diagnosed. Jiann-Jone Chen et al. proposed a new method where CBIR can be used by Dual Multiscale Morphological Reconstructions [4]. In the paper, to segment objects in one image, they use dual multiscale Graylevel morphological open and close reconstructions (SEGON) to generate a background (BG) gray-level variation mesh, which can

help to identify BG and object regions. CBIR can be used in Histopathological image classification (HIC) and content-based histopathological image retrieval (CBHIR), as proposed by Yushan Zheng et al. [54]. In general, HIC can aid pathologists in locating high-risk cancer regions from a whole slide image (WSI) by providing a cancerous probability map for the WSI. R. Ramu Naidu et al. proposed a new method to create Deterministic Compressed Sensing Matrices [31], which is very helpful to CBIR applications like text-based retrieval (such as Google search) and retrieval from medical databases. Lining Zhang et al. proposed a new method of semi-supervised biased maximum margin analysis for interactive image retrieval [25] which can be used in many CBIR applications. Zhihua Xia et al. proposed a new CBIR method, which come in useful in Privacy-Preserving in cloud computing [52]. In their paper, they proposed a privacy-preserving content-based image retrieval scheme, which allows the data owner to outsource the image database and CBIR service to the cloud, without revealing the actual content of the database to the cloud server. Local features are utilized to represent the images, and earth mover's distance (EMD) is employed to evaluate the similarity of images. Gwenole Quellec et al. proposed a new CBIR method which can be used for medical case retrieval from a committee of decision trees [36]. In their proposed framework, images are characterized by their digital content. The proposed method allows the retrieval of possibly incomplete medical cases consisting of several images together with semantic information. It relies on a committee of decision trees and decision support tools well suited to process this type of information. Herbert Daschiel et al. proposed a new CBIR concept for knowledge-driven image information mining (KIM) and both analyze and evaluate it from the perspective of human-machine communication [8]. The KIM concept enables information communication from a very large image repository to users via the Internet. In [10] Optimal Cost Region Matching Similarity Measure is proposed for region-based image retrieval. Santosh et al. [39] have proposed an efficient method for overlaid arrow detection for labeling regions of interest (ROI) in biomedical images by using fuzzy binarization. In [40], for automatic pulmonary abnormality screening in chest, X-ray analysis by Edge Map analysis method is proposed. For this analysis pyramid histograms of oriented gradients (PHOG) are used to obtain the edge map.

To improve detection of pulmonary and pleural abnormalities caused by pneumonia or tuberculosis (TB) in digital chest X-rays (CXRs), Alexandros Karargyris et al. [18] used a combination of texture and shape features. Santosh et al. [38] proposed a novel method for automated CXR screening by using multiscale shape features and edge plus texture features to decide whether lung region symmetry helps detect Pulmonary Abnormalities of a HIV patient. Szilard Vajda et al. [47] provided a completely automatic frontal chest radiograph screening system that was able to detect healthy lungs and spot abnormal ones carrying different types of tuberculosis manifestations. In [2], the authors used morphological operators to describe the color feature for CBIR. They used granulometries, morphological leveling and watersheds to describe the color information in an image. Xiang-Yang Wang et al. [51] proposed a method for CBIR by integrating color and texture features. To get the color feature, they used pseudo-Zernike chromaticity distribution moments in opponent chromaticity space. In [22], Bin Li et al. proposed a method to recognize buildings based on sparse representation of spatial

texture and color features. This method used multi-scale neighborhood sensitive histograms of oriented gradient (MNSHOGs) and color auto-correlogram (CA) to extract texture and color features of building images. Li Liu et al. [26] proposed a method for texture classification in extreme scale variations using GANet where it uses Genetic Algorithm to change the filters in the hidden layers during network training, to promote the learning of more informative semantic texture patterns. For this, they used dominant texture pattern, i.e. texels. Meeras Salman Al-Shemarry et al. [1] proposed an efficient texture descriptor for the detection of License plates from vehicle images in difficult conditions such as low/high contrast, foggy, distorted, and dusty conditions. They used a multi-level extended local binary pattern for this.

In this paper, we use subtle concept of inter-channel voting, thereby establishing a correspondence between H, S, and I channels. This concept combines the notion of all features concerned with CBIR so that it becomes applicable to myriad images from different databases and aim at effective results.

2 Main contribution

2.1 Congruence on hue, saturation, and intensity

The motivation for shifting focus from segregating H, S and I channels individually into bins and concatenating their histograms to perform the operation of voting (inter-channel voting) is that we obtain the relationship among all three H, S and I channels. We have quantized the Hue channel into distinct bins and then added up the Saturation value to the corresponding bin of Hue channel and vice versa. The same approach is applied for Saturation & Intensity and Intensity & Hue channels, which results in a total of 6 permutations. Inter-channel voting operation is not commutative in nature; for example, *Effect of Hue on Saturation is different from the effect of Saturation on Hue*. Driven by this fact, we study the variation of one component with respect to other components independently. To fulfill this task, we quantize the value of one component C_1 into different bins and create the corresponding histogram by voting against the values of another component C2. Let the C1 value at the pixel location (a, b) of the image be V1. If it belongs to the kth quantized bin in C1, then $Bin_{C_1}(k)$ will be updated as given in eq. (22).

$$Bin_{C1}(k) = Bin_{C1}(k) + C_2(a,b) \tag{22}$$

where $C_2(a,b)$ is the value of component C_2 at pixel position (a,b). This process is shown in Fig. 5. The entire process of inter-channel voting among all the six permutations is performed as given in eqs. (23) to (34).

//For HUE//

$$Range_H = \frac{\max(H(i,j)) - \min(H(i,j))}{H_{level}}; \forall i,j \in H \tag{23}$$

where H_{level} is the number of quantization levels for *Hue*.

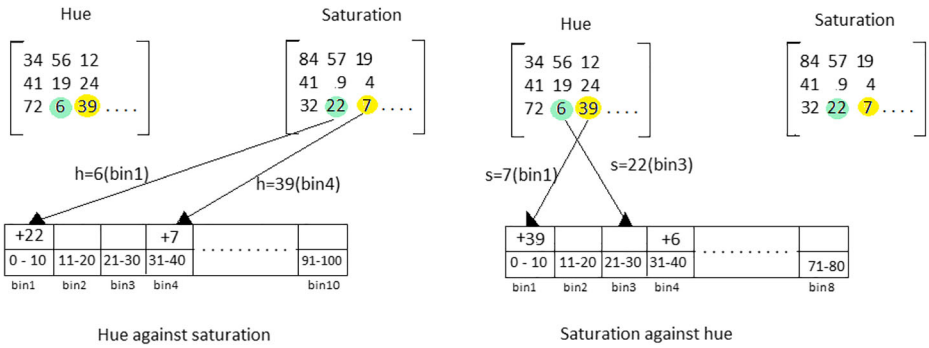

Fig. 5 (a) Inter-channel voting process between Saturation and Value components

$$H_{new}(i,j) = \begin{cases} H_{level}-1, & H(i,j) = \max(H(i,j)) \\ \left\lfloor \dfrac{H(i,j)}{Range_H} \right\rfloor, & else \end{cases} \tag{24}$$

$$Bin_{HS}\left(H_{new}\left(i,j\right)\right) = Bin_{HS}(H_{new}(i,j)) + S(i,j), \forall i,j \in H \tag{25}$$

$$Bin_{HI}\left(H_{new}\left(i,j\right)\right) = Bin_{HI}(H_{new}(i,j)) + I(i,j), \forall i,j \in H \tag{26}$$

//For SATURATION//

$$Range_S = \frac{\max(S(i,j)) - \min(S(i,j))}{S_{level}} \forall i,j \in S \tag{27}$$

where S_{level} is the number of quantization levels for *Saturation*.

$$S_{new}(i,j) = \begin{cases} S_{level}-1, & S(i,j) = \max(S(i,j)) \\ \left\lfloor \dfrac{S(i,j)}{Range_S} \right\rfloor, & else \end{cases} \tag{28}$$

$$Bin_{SH}\left(S_{new}\left(i,j\right)\right) = Bin_{SH}(S_{new}(i,j)) + H(i,j), \forall i,j \in S \tag{29}$$

$$Bin_{SI}\left(S_{new}\left(i,j\right)\right) = Bin_{SI}(S_{new}(i,j)) + I(i,j), \forall i,j \in S \tag{30}$$

//**For INTENSITY**//

$$Range_I = \frac{\max(I(i,j)) - \min(I(i,j))}{I_{level}} \forall i,j \in I \tag{31}$$

where I_{level} is the number of quantization levels for *Intensity*.

$$I_{new}(i,j) = \begin{cases} I_{level} - 1, & I(i,j) = \max(I(i,j)) \\ \left\lfloor \frac{I(i,j)}{Range_I} \right\rfloor, & else \end{cases} \tag{32}$$

$$Bin_{IH}\left(I_{new}\left(i,j\right)\right) = Bin_{IH}(I_{new}(i,j)) + H(i,j), \forall i,j \in I \tag{33}$$

$$Bin_{IS}\left(I_{new}\left(i,j\right)\right) = Bin_{IS}(I_{new}(i,j)) + S(i,j), \forall i,j \in I \tag{34}$$

We also propose to give different weights as given in eq. (35) to different features in our CBIR system to say that appropriate weight is given to dominant features of the images in the image database, which results in higher efficiency.

$$D = \sum_{i=1}^{3} D_i W_i : \forall i \; W_i \in Z^+ and \; W_i \neq W_j if \; i \neq j, \forall j \in \{1,2,3\} \tag{35}$$

D_i is the resultant distance of separate feature components {shape, color, texture} and W_i is the corresponding weightage factor given.

3 Proposed method

In the proposed framework, the authors have tried to get highly useful information from the images using the methods explained above. This work explores the method of quantization, inter-channel voting, DSP and GLCM. It also exploits the properties and characteristics of the channels in HSI color space that give color and texture features of the image. To reduce the size of the feature vector, as given in most of the literature [24], quantization method was used for '*n*' number of bins for reasonable information extraction and efficient memory utilization. In the proposed method, 72 were used bins for Hue, 20 bins for Saturation [3] and 32 bins for Intensity channel. Hue is a component which is dependent on dominant wavelength and independent of intensity. This intensity can be extracted from the saturation part. The above-mentioned interrelation between Hue and Saturation is explored in [3]. Interrelating Hue and Saturation values results in obtaining only the color information of the pixel, where the texture information of a pixel is neglected. So, we established a relationship between all three: Hue, Saturation, and Intensity such that both color and texture information related to a pixel is obtained. Hence, we can extract complete information from an image. We used DSP to get the

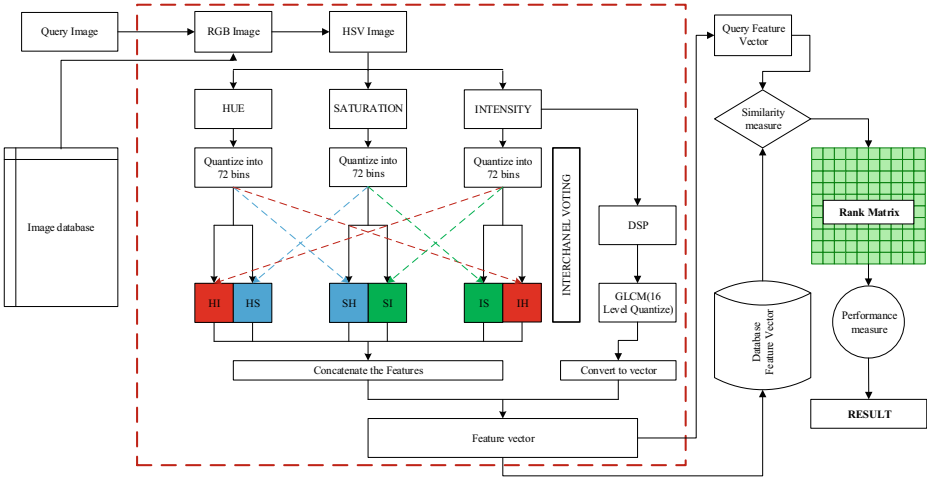

Fig. 6 Block diagram of the proposed system

mutual relationship between diagonally symmetric pixel pairs of an n × n window of an image, as explained in [3]. After quantizing DSP, we can compute histogram to get a vector, but it loses the spatial relationship of the pixels. So, GLCM is used to get the co-occurrence of two pixels at a specified distance and direction, as mentioned in Section 1.3. To measure the similarity between the query image and the images in the database, different types of distances can be used: *Manhattan, Euclidean, d_1 distance*, *Canberra*, *chi-square,* and *histogram intersection* [53]. The entire process is shown in Fig. 6, and the corresponding steps are given in algorithm-1.

Algorithm-1:

1. Take an image as input.
2. Convert RGB components of that image into HSI color space.
3. Applying congruence on Hue, Saturation, and Intensity by considering all the permutations results in six vectors.
4. Apply DSP on Intensity part of HSI color space.
5. Quantize DSP into 16 levels on which Gray Level Co-occurrence Matrix is computed.
6. Convert 16×16 GLCM into a vector form.
7. The final feature vector is the concatenation of all the vectors obtained from the above steps.
8. Use similarity measure to compare the features of the query image (input image) to all the images in the dataset.
9. Obtain the rank matrix by sorting according to the distance.

---

4 Experimental results and discussions

To evaluate the performance of the proposed method, *six* benchmark databases of color images have been used. We have compared the results obtained from the proposed method with nine of the existing techniques shown in Table 1. Different performance measures such as Precision, Recall, F-measure and ANMRR are used. A new parameter TMRE is introduced and used to compare the results. While evaluating, each query image is processed through the proposed method to get the feature vector. This feature vector is compared with the feature vector of all the images of the database by using distance. Based on the distances, a rank matrix of size $N \times N$ is

Table 1 Different methods used for evaluating the performance

	Method Name
'Color Only' as Feature Vector	Colorhist_RGB Colorhist_HSV Color Auto Correllogrm
'Texture Only' as Feature Vector	LBP ULBP CS_LBP
'Color+Texture' as Feature Vector	Colorhist_HSV + CS_LBP Colorhist_HSV+ULBP LECoP
'Color and Texture Integrated' Feature Vector	Proposed Method

formed, where N is the total number of images in the database, shown in Fig. 7. The value of each cell of the rank matrix is given as Rank(k,i) which is a k^{th} similar image with respect to the i^{th} query image.

4.1 Precision & Recall

Precision is described as the ratio of total number of relevant images retrieved to the number of images retrieved for a query image. The recall is described as the ratio of total number of relevant images retrieved for a query image to the total number of images in that category. Let N_{ic} be the total number of images in category C_i. Let a total of n images be retrieved from a database for a query image i from that category, so that precision and recall are calculated by using eq. (36) to eq. (38). Here, in eq. 36 and eq. 37, $f3$ results in 1 if the rank of the image in the database belongs to the same category of query image; otherwise, it results in 0. The average precision and average recall for the j^{th} category are calculated using eq. (39) and eq. (40). If the total number of categories is N_c, then Average Precision Rate (APR) and Average Recall Rate (ARR) for n images are calculated using eqs. (41) and (42).

$$P(i,n) = \frac{1}{n}\sum_{k=1}^{n} f_3(Rank(k,i)) \tag{36}$$

		Query Images considered from the Database								
		Class-1				...	...	...	**Class-10**	
		Image1	**Image2**	...	**Image100**	...	...	...	**Image999**	**Image1000**
Ranks given to images in the Database	1	Image1								
	2	Image 21								
	3	Image 804								
		...								
	1000									

Fig. 7 Rank Matrix Representation for 1000 images database

$$R(i,n) = \frac{1}{N_{ic}} \sum_{k=1}^{n} f_3(Rank(k,i)) \tag{37}$$

$$f_3(Rank(k,i)) = \begin{cases} 1, & rank(k,i) \in C_i \\ 0, & else \end{cases} \tag{38}$$

$$P_{avg}(j,n) = \frac{1}{N_{ic}} \sum_{i=1}^{N_{ic}} P(i,n) \tag{39}$$

$$R_{avg}(j,n) = \frac{1}{N_{ic}} \sum_{i=1}^{N_{ic}} R(i,n) \tag{40}$$

$$APR(n) = \frac{1}{N_c} \sum_{j=1}^{N_c} P_{avg}(j,n) \tag{41}$$

$$ARR(n) = \frac{1}{N_c} \sum_{j=1}^{N_c} R_{avg}(j,n) \tag{42}$$

4.2 F-measure

Precision and Recall are good performance measures. To understand the relationship between these two measures, we calculate the third performance measure *F-measure* given in eq. (43).

$$F\text{-}measure(n) = \frac{(2 \times APR(n) \times ARR(n))}{(APR(n) + ARR(n))} \tag{43}$$

4.3 Average Normalized Modified Retrieval Rank (ANMRR)

To measure the retrieval accuracy, we used ANMRR measure, which can be calculated using eq. (44). The average retrieval rank (AVGRR) is calculated by using eq. (45). Now, normalized modified retrieval rank (NMRR) for query image i, can be calculated using eq. (46). Finally, the average NMRR for the entire database can be calculated as given in eq. (47).

$$RR(k,i) = \begin{cases} f_5(k,i) & , \ if\ f_5(k,i) \le f_6(i) \\ 1.25 \times f_6(i), & else \end{cases} \tag{44}$$

$\forall k \in C(i)$, where $C(i)$ is the set of all the images of the category C_i,

$$f_5(k,i) = x, where\ Rank(x,i) = k \text{ (and)}$$
$$f_6(i) = \min\left(4 \times N_{ic}, 2 \times \left\{\max\left(N_{ic}\right), \forall i\right\}\right).$$

$$AVGRR(i) = \frac{1}{N_{ic}} \sum_{k=1}^{N_{ic}} RR(k,i) \tag{45}$$

$$NMRR(i) = \frac{AVG(i) - 0.5[1 + N_{ic}]}{1.25 f_6(i) - 0.5[1 + N_{ic}]} \tag{46}$$

$$ANMRR = \sum_{i=1}^{DB} NMRR(i) \tag{47}$$

where $DB = N_c \times N_{ic}$, total number of images in database.

4.4 Total Minimum Retrieval Epoch (TMRE)

In addition to the above mentioned four performance measures, one new performance measure is introduced for overall performance evaluation of the entire database, which is known as the minimum retrieval epoch ratio. It is defined as an average of ratios between the average number of images to be traversed to get all the relevant images for each query image of a category and total number of images of that category. To get TMRE for the entire dataset, initially, minimum retrieval epoch MRE(i) for each query image can be calculated by using eq. (48). Average MRE for each category $AMRE(C_i)$ can be calculated using eq. (49). Then AMRE value is divided by the total number of images in corresponding category to make performance measure a more generalize for any database. AMRE Ratio for one category $AMRER(C_i)$is calculated by using eq. (50). Finally, AMRER for the entire dataset i.e. TMRE is calculated using eq. (51).

$$MRE(i) = \max(k), \forall k \exists Rank(k, i) \in C_i \tag{48}$$

$$AMRE(C_i) = \frac{1}{N_{ic}} \sum_{i=1}^{N_{ic}} MRE(i) \tag{49}$$

$$AMRER(C_i) = \frac{AMRE(C_i)}{N_{ic}} \tag{50}$$

$$TMRE = \frac{1}{N_c} \sum_{C_i=1}^{N_c} AMRER(C_i) \tag{51}$$

4.5 Database-1 (Corel-1 K)

Corel-1 K database [50], which is the first database to evaluate the performance of the proposed method with other existing methods, comprises 10 categories of 100 images each resulting in a total of 1000 images in the database. This database includes images of Africans (1–100), Beaches (101–200), Buildings (201–300), Buses (301–400), Dinosaurs (401–500), Elephants (501–600), Flowers (601–700), Horses (701–800), Mountains (801–900) and Food (901–1000). Size of images in this database is either 384 × 256 or 256 × 384. Some sample images from this database are shown in Fig. 8, in which three images per category are shown. Precision, Recall, F-measure, ANMRR and TMRE are obtained according to Eqs. (36), (37), (38), (42) and (46) respectively.

In experiments, a different number of images for each query image (10,20,...100) is retrieved separately, and its performance is evaluated. Figure 9 shows the APR of the proposed method as well as all ten other methods mentioned, by retrieving different

Fig. 8 Corel-1 K Samples (three images per category)

number of images. Similarly, Fig. 10 shows ARR. Precision vs. Recall graph for this database is shown in Fig. 11. The f-measure graph is shown in Fig. 12. Finally, category wise precision and recall graphs are shown in Fig. 13 and Fig. 14 respectively. The APR, ARR, F-measure, ANMRR, and TMRE values for the Corel-1 K database are shown in Table 2. Here we can see both precision and recall increased around 4%,F-measure is increased by 3% and ANMRR improved by 0.5.

4.6 Database-2 (Corel-5 K)

Corel-5 K image database [27] is the next database which consider for the purpose of experiment. It consists of 5000 images of 50 categories consisting of 100 images each. It contains images of animals, e.g., bear, fox, lion, tiger, etc., human, natural scenes, buildings, paintings, fruits, cars, etc. The size of images in this database is

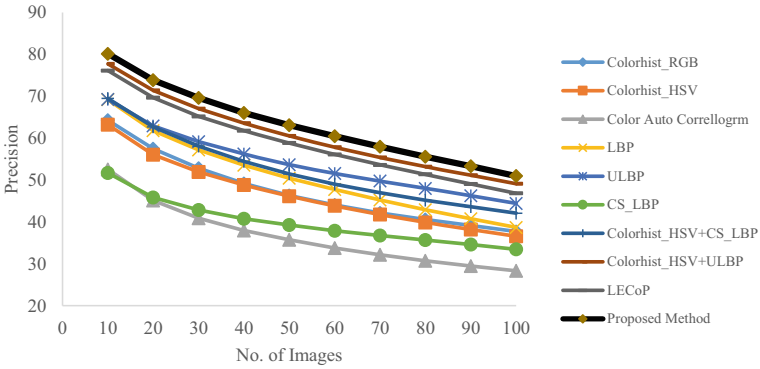

Fig. 9 Precision Graph with the different number of retrieved images for Corel-1 K

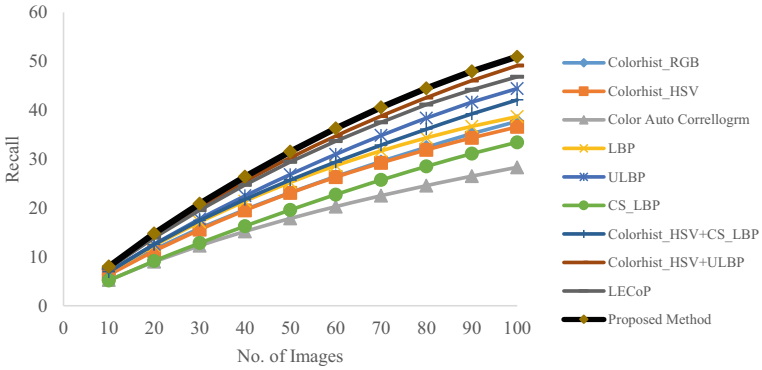

Fig. 10 Recall Graph with the different number of retrieved images for Corel-1 K

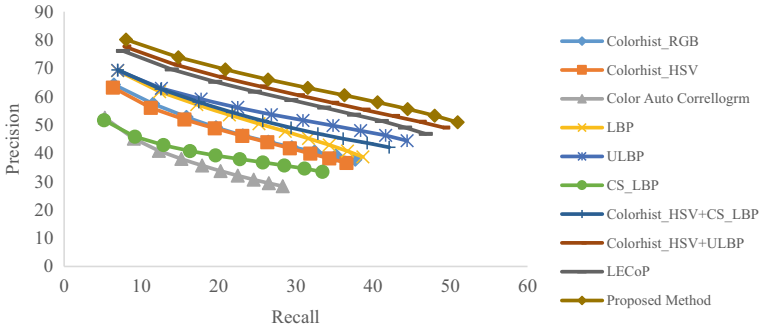

Fig. 11 Precision vs. Recall Graph with the different number of retrieved images for Corel-1 K

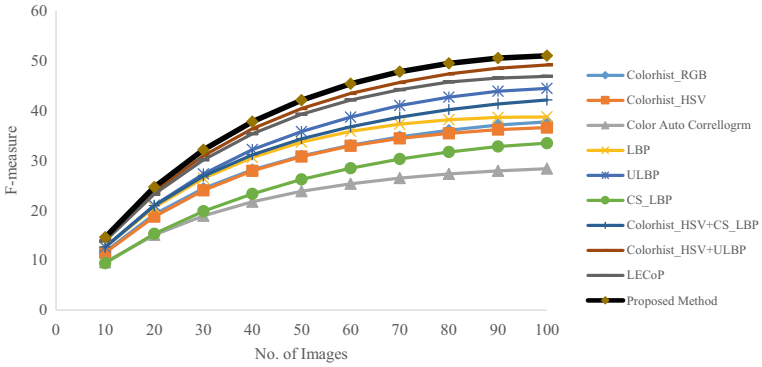

Fig. 12 F-measure graph with the different number of retrieved images for Corel-1 K

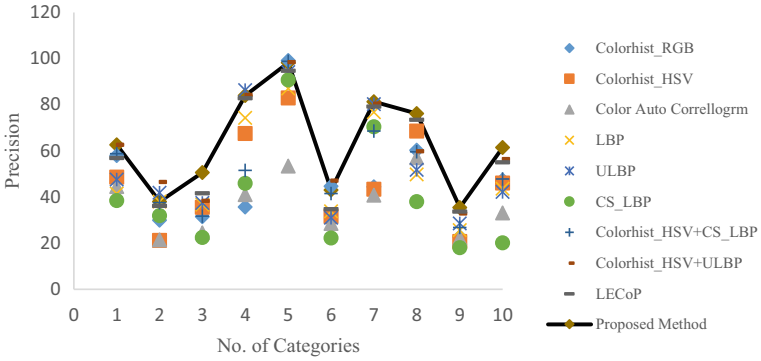

Fig. 13 Category wise precision graph for Corel-1 K

either 187 × 126, 126 × 187, 128 × 192, 192 × 128, 126 × 188 or 188 × 126. In Fig. 15, one image from each category of this database is shown. Precision, Recall, F-measure, ANMRR and TMRE are obtained. In this experiment, different images (10,20,...100) for each query image are retrieved separately, and the performance is evaluated. Figure 16 shows the APR of the proposed method as well as ten othe methods mentioned, by retrieving a different number of images. Similarly, Fig. 17 shows ARR for this database. The APR, ARR, F-measure, ANMRR, and TMRE values for Corel-5 K database are shown in Table 3. Here we can see that precision is increased by 4%, both recall and F-measure by 3%, ANMRR is improved by 0.04 and TMRE is improved by 1.

4.7 Database-3 (Corel-10 K)

Corel-10 K image database [27] is the other database for experiment, and it has 10000 images of 100 categories, and 100 images per category. Out of these 10000, the first 5000 images are the same as that from Corel-5 K. This database contains a total of 53 different sized images:128 × 192, 192 × 128, etc. As this database is a superset of Database-2, one image from each additional 50 categories is shown in

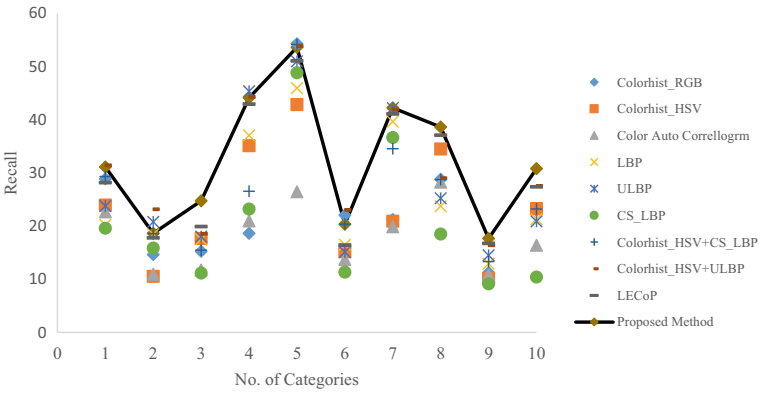

Fig. 14 Category wise Recall graph for Corel-1 K

Table 2 Performance measures for Core-1 K

		Corel 1 K				
		APR	ARR	F-Measure	ANMRR	TMRE
1	Colorhist_RGB	64.28	37.71	29.275	0.5334	8.08
2	Colorhist_HSV	63.21	36.56	28.819	0.5335	8.73
3	Color Auto Correllogrm	52.54	28.32	22.423	0.6278	9.46
4	LBP	69.18	38.69	31.228	0.5208	8.04
5	ULBP	69.26	44.42	33.929	0.4574	7.50
6	CS_LBP	51.69	33.44	25.045	0.5746	7.82
7	Colorhist_HSV + CS_LBP	69.47	42.09	32.468	0.4817	7.63
8	Colorhist_HSV + ULBP	77.73	49.11	37.944	0.4077	7.24
9	LECoP	76.14	46.82	36.691	0.4301	7.59
10	Proposed Method	80.15	50.94	39.502	0.3880	7.35

Fig. 18. Precision, Recall, F-measure, ANMRR and TMRE are obtained. In this experiment, different images(10,20,...100) for each query image are retrieved separately, and performance is evaluated. Figure 19 shows the APR of the proposed method as well as other ten methods mentioned, by retrieving a different number of images. Similarly, Fig. 20 shows ARR for this database. The APR, ARR, F-measure, ANMRR, and TMRE values for the Corel-10 K database are shown in Table 4. Here again we can see precision is increased around 5%, recall is increased by around 3%, F-measure is by 2%, ANMRR is improved by 0.02 and TMRE is improved by 2. On evaluating the results of all three natural image database: Corel-1 K, Corel-5 K, and Corel-10 K, our proposed method is gives far better results than existing methods.

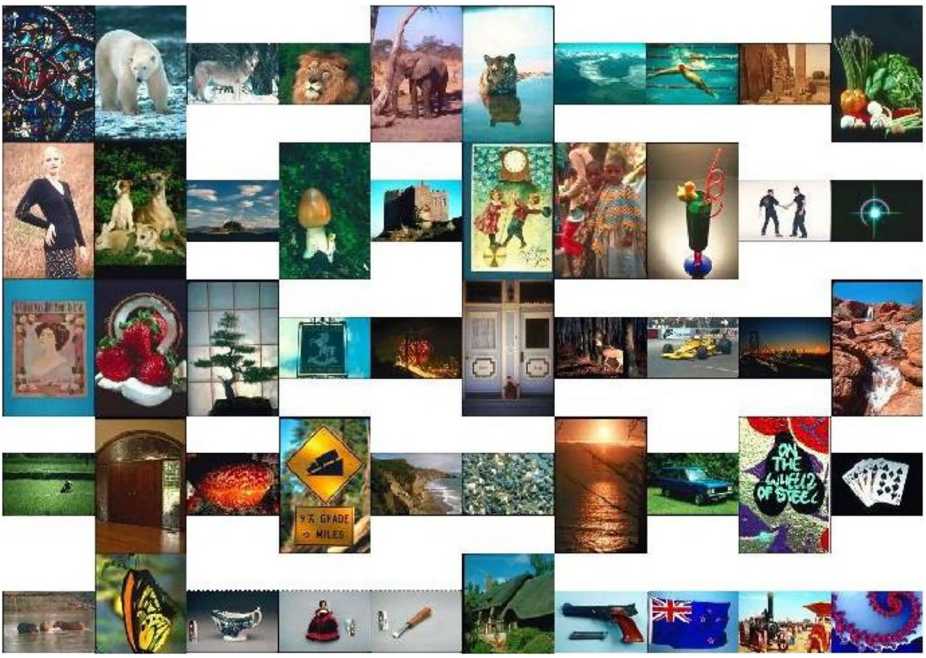

Fig. 15 Corel-5 K Samples (one image per category)

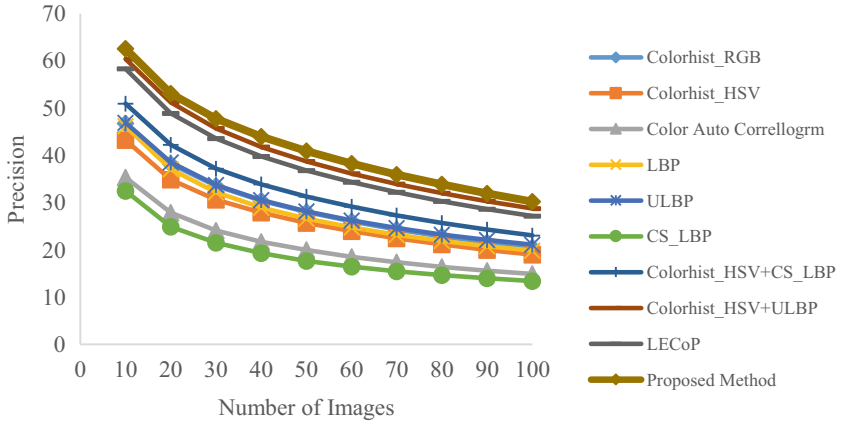

Fig. 16 Precision Graph with the different number of retrieved images for Corel-5 K

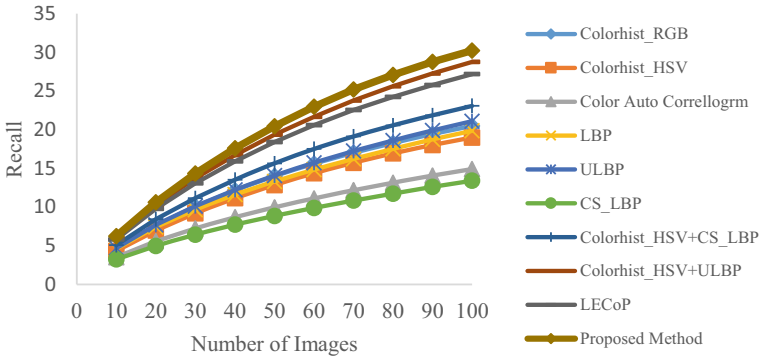

Fig. 17 Recall Graph with a different number of retrieved images for Corel-5 K

Table 3 Performance measures for Core-5 K

		Corel 5 K				
		APR	ARR	F-Measure	ANMRR	TMRE
1	Colorhist_RGB	46.81	20.49	17.356	0.7414	42.23
2	Colorhist_HSV	43.26	19.00	15.996	0.7585	42.91
3	Color Auto Correllogrm	35.33	14.91	12.509	0.8065	44.67
4	LBP	46.19	19.85	16.667	0.7446	41.62
5	ULBP	46.94	21.10	17.584	0.7276	38.88
6	CS_LBP	32.50	13.40	11.178	0.8241	42.58
7	Colorhist_HSV + CS_LBP	50.97	23.07	19.411	0.7089	40.51
8	Colorhist_HSV + ULBP	60.43	28.76	23.989	0.6410	37.66
9	LECoP	58.34	27.18	22.782	0.6603	37.58
10	Proposed Method	**62.60**	**30.23**	**25.250**	**0.6269**	**36.56**

Fig. 18 Corel-10 K Samples

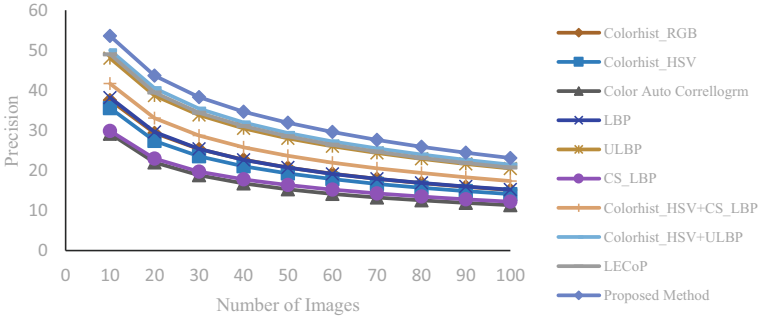

Fig. 19 Precision Graph with the different number of retrieved images for Corel-10 K

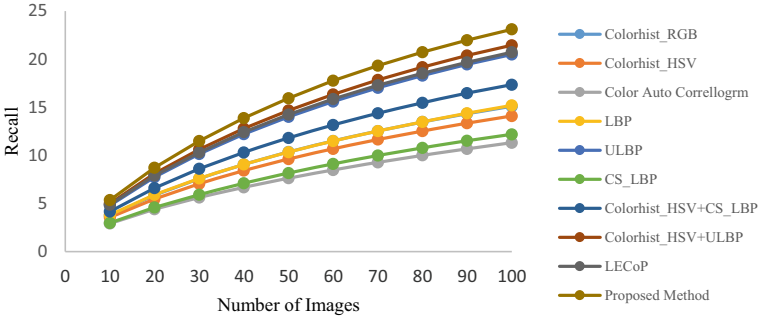

Fig. 20 Recall Graph with the different number of retrieved images for Corel-10 K

Table 4 Performance measures for Core-10 K

		Corel 10 K				
		APR	ARR	F-Measure	ANMRR	TMRE
1	Colorhist_RGB	37.51	15.11	12.936	0.8101	85.50
2	Colorhist_HSV	35.57	14.10	12.045	0.8204	84.92
3	Color Auto Correllogrm	29.18	11.32	9.628	0.8530	86.74
4	LBP	38.17	15.20	12.969	0.8057	83.20
5	ULBP	48.01	20.49	17.419	0.7445	77.95
6	CS_LBP	29.88	12.19	10.271	0.8445	86.45
7	Colorhist_HSV + CS_LBP	41.66	17.35	14.760	0.7823	81.77
8	Colorhist_HSV + ULBP	49.99	21.45	18.244	0.7329	77.16
9	LECoP	48.99	20.72	17.685	0.7398	75.65
10	Proposed Method	**53.57**	**23.10**	**19.728**	**0.7121**	**73.57**

4.8 Database-4 (VisTex)

To evaluate the performance of the proposed method, the other three image databases considered are colored texture databases. Database 4 is gathered from MIT VisTex database [35]. It consists of a total of 484 texture images with different dimensions. Among these texture image, forty texture images of dimension 512 × 512 are considered for experimentation. These forty texture images are divided into sixteen 128 × 128 nonoverlapping subimages; hence we obtain a total of 640 texture images. The forty texture images are shown in Fig. 21. Precision, Recall, F-measure, ANMRR and TMRE are obtained. In this experiment, different sets of images (4,8,12,16) for

Fig. 21 Forty VisTex texture images considered

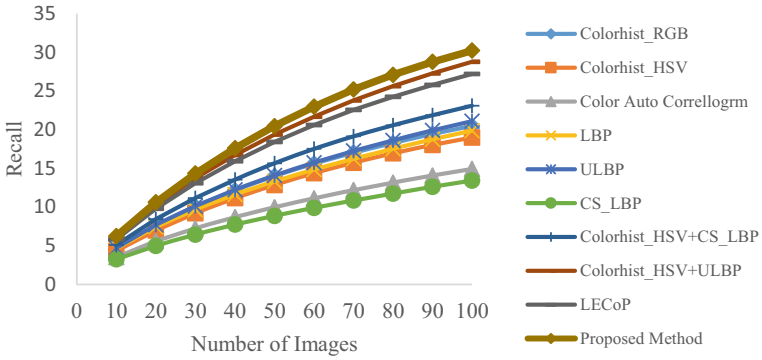

Fig. 22 Precision Graph with the different number of retrieved images for VisTex database

each query image are retrieved separately, and performance is evaluated. Figure 22 shows the APR of the proposed method as well as other ten methods mentioned, by retrieving a different sets of images. Similarly, Fig. 23 shows ARR for this database. Precision vs. Recall graph for this database is shown in Fig. 24 The f-measure graph is shown in Fig. 25. Finally, category wise precision and recall graphs are shown in Fig. 26 and Fig. 27 respectively. The APR, ARR, F-measure, ANMRR, and TMRE values for the VisTex database are shown in Table 5. Here, precision is increased by 0.4% and F-measure is by 0.1% .

4.9 Database-5 (STex)

For the performance analysis of the proposed method, one more color-texture image database is used, which is the Salzburg Texture Image Database (STex) [21]. This database consists of a total of 476 texture images, each one of size 512 × 512. Here again, each of the 476 texture images is grouped into 16 nonoverlapping images of size 128 × 128. Hence, it creates a total of 7616 texture images. Some of the textures images from this database are shown in Fig. 28. Precision, Recall, F-measure, ANMRR and TMRE are obtained for this image database. In this experiment, different sets of images (4,8,12,16) for each query image are retrieved separately, and performance is evaluated. Figure 29 shows the APR of the proposed method as well as ten

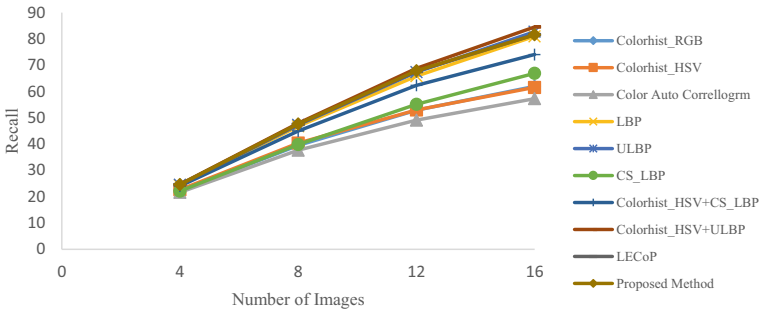

Fig. 23 Recall Graph with the different number of retrieved images for VisTex database

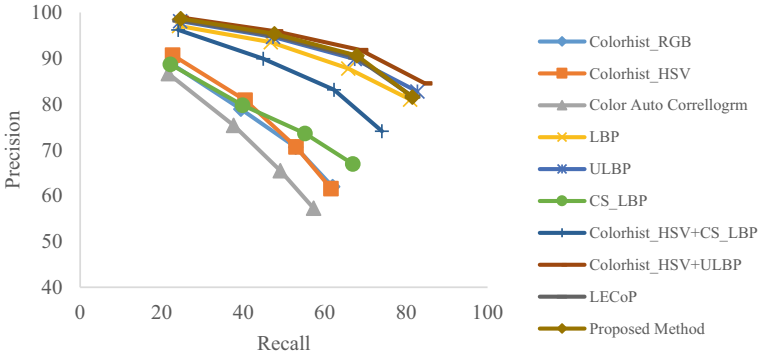

Fig. 24 Precision vs. Recall Graph with the different number of retrieved images for VisTex database

other methods mentioned, by retrieving a different number of images. Similarly, Fig. 30 shows ARR for this database. The APR, ARR, F-measure, ANMRR, and TMRE values for the STex database are shown in Table 6. It is apparent that precision is increased by 2%, recall is increased by 3%, F-measure is increased by 2%, ANMRR is improved by 0.02 and TMRE is improved by 2.

4.10 Database-6 (Color Brodatz)

For 6th database, color Brodatz texture image database [9] is used. It is also a collection of colored textured images. The database contains 112 different color texture images each having a size of 640 × 640. Each image is divided into 25 nonoverlapping images of size 128 × 128 which create a total of 2800 texture images. Some of the textures images from this database are shown in Fig. 31 Precision, Recall, F-measure, ANMRR and TMRE are obtained for this image database. In this experiment, different sets of images (5,10,15, 20, 25) for each query image are retrieved separately, and performance is evaluated. Figure 32 shows the APR of the proposed method as well as all ten othe methods mentioned, by retrieving a different number of images. Similarly, Fig. 33 shows ARR for this database. The APR, ARR, F-measure, ANMRR, and

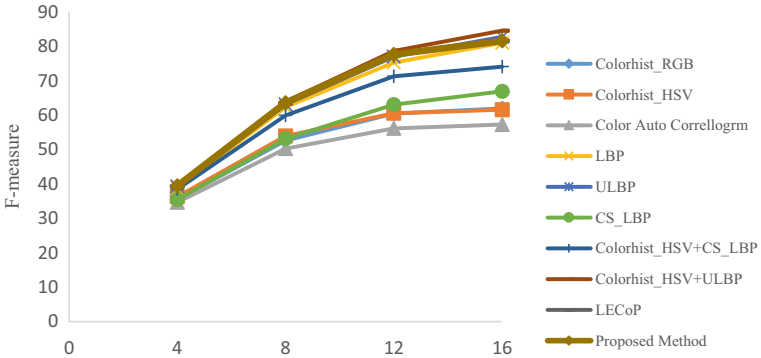

Fig. 25 F-measure graph with the different number of retrieved images for VisTex database

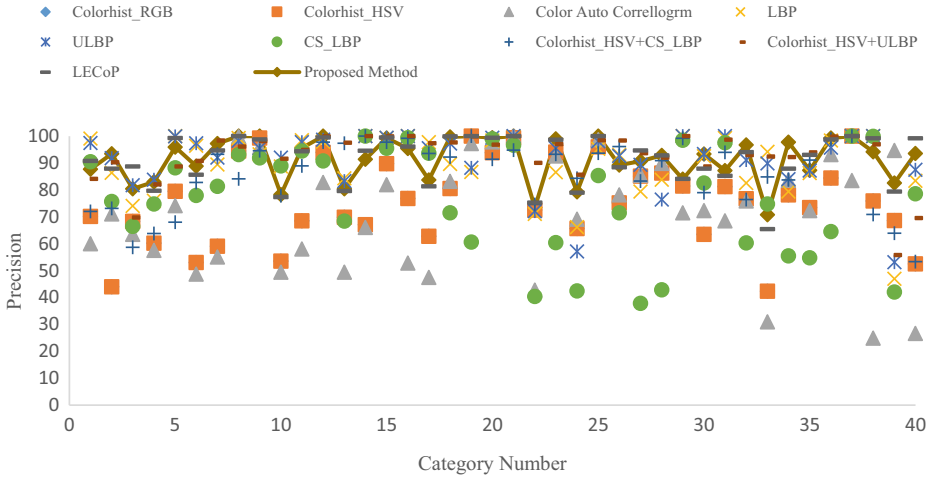

Fig. 26 Category wise Precision graph for VisTex database

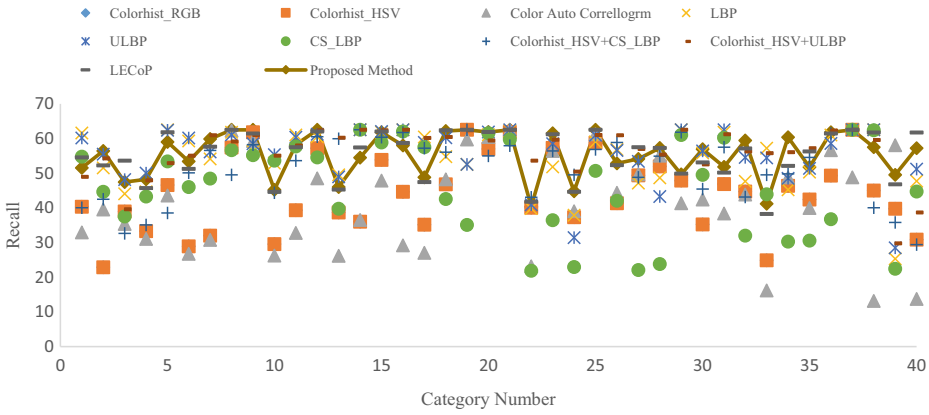

Fig. 27 Category wise Recall graph for VisTex database

Table 5 Performance measures for VisTex database

		VisTex				
		APR	ARR	F-Measure	ANMRR	TMRE
1	Colorhist_RGB	88.91	61.96	52.658	0.2866	13.44
2	Colorhist_HSV	90.70	61.58	53.085	0.2995	12.00
3	Color Auto Correllogrm	86.64	57.28	49.576	0.3462	15.56
4	LBP	97.07	81.00	64.357	0.1216	4.38
5	ULBP	98.13	82.75	65.522	0.1081	**4.00**
6	CS_LBP	88.67	66.92	54.663	0.2448	5.75
7	Colorhist_HSV + CS_LBP	96.21	74.08	60.938	0.1785	8.63
8	Colorhist_HSV + ULBP	98.91	**84.55**	66.679	**0.0951**	4.19
9	LECoP	98.28	81.63	65.402	0.1254	5.06
10	Proposed Method	**98.67**	81.56	**65.564**	0.1280	5.25

Fig. 28 Fifty of the STex texture image

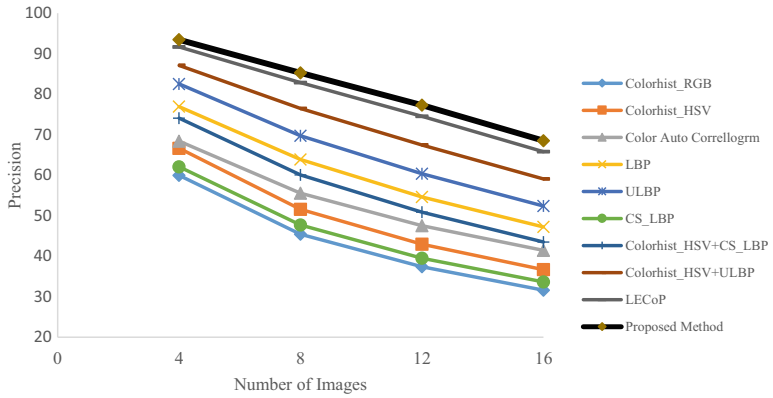

Fig. 29 Precision Graph with a different number of retrieved images for STex database

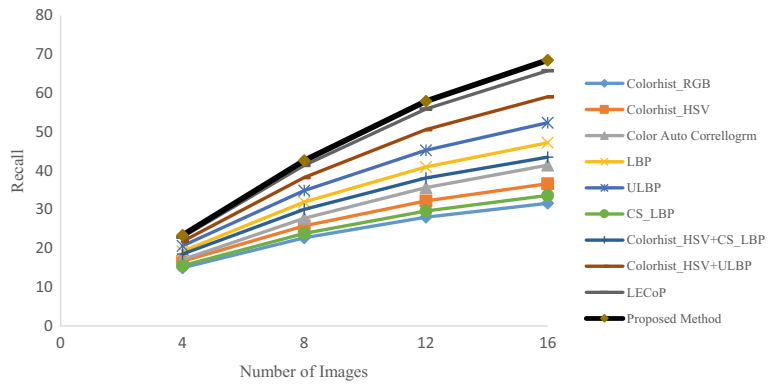

Fig. 30 Recall Graph with the different number of retrieved images for STex database

Table 6 Performance measures for STex database

		STex				
		APR	ARR	F-Measure	ANMRR	TMRE
1	Colorhist_RGB	59.96	31.62	29.480	0.6218	185.88
2	Colorhist_HSV	66.66	36.67	33.639	0.5606	115.13
3	Color Auto Correllogrm	68.41	41.44	36.642	0.5020	104.94
4	LBP	76.92	47.22	41.849	0.4526	104.13
5	ULBP	82.54	52.39	45.915	0.4027	86.56
6	CS_LBP	62.08	33.65	31.033	0.5987	139.13
7	Colorhist_HSV + CS_LBP	74.08	43.51	39.201	0.4928	131.06
8	Colorhist_HSV + ULBP	87.11	59.08	50.683	0.3333	73.56
9	LECoP	91.67	65.80	55.397	0.2657	44.44
10	Proposed Method	**93.44**	**68.53**	**57.245**	**0.2422**	**42.44**

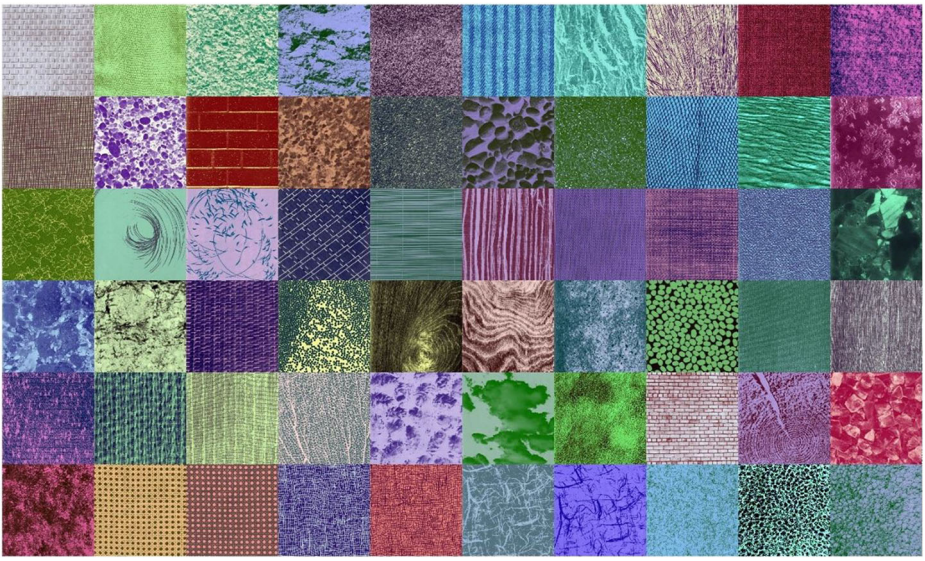

Fig. 31 Sixty of the Color Brodatz texture images

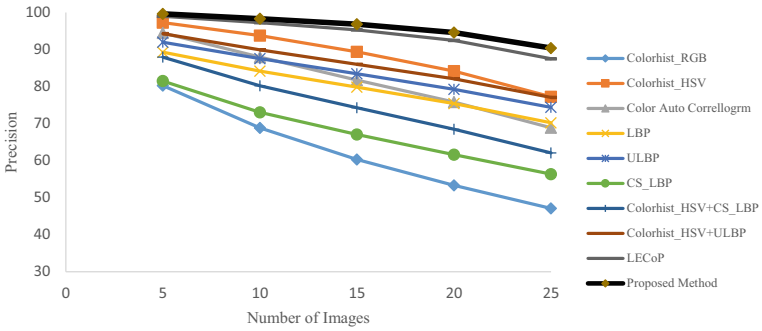

Fig. 32 Precision Graph with the different number of retrieved images for Color Brodatz texture database

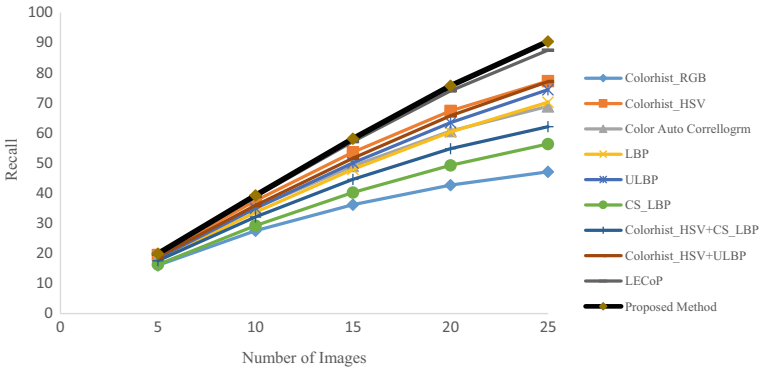

Fig. 33 Recall Graph with a different number of retrieved images for Color Brodatz texture database

TMRE values for Color Brodatz database are shown in Table 7. Once again, precision is increased by 1%, recall is increased by 3%, F-measure is increased by 2%, ANMRR is improved by 0.02 and TMRE is improved by 2. The results of all three color texture databases: VisTex, STex, and Color Brodatz, make it clear that the method prposed in this paper yield fat better results than existing methods.

4.11 Experiment results with different weighted factors

To observe the performance, four different weighted factors (1,2), (2,1), (1,3) and (3,1) for color and texture features are used. Comparison between all four weighted factors results are shown in Table 8 and Table 9 for natural image databases and color texture image databases, respectively in terms of ARR, APR, F-Measure, ANMRR, and TMRE. The data shows that Corel databases give better performance with more weightage to texture. Corel-1 K and Corel-5 K give better result with a weighted factor of 2 to texture, whereas Corel-10 K gives better performance for a weighted factor of 3. In the case of color texture databases, both STex and Color Brodatz give better result with a weighted factor 2 to texture component, whereas VisTex dataset performs better with a texture weightage of factor 3.

Table 7 Performance measures for Color Brodatz texture database

		Color Brodatz				
		APR	ARR	F-Measure	ANMRR	TMRE
1	Colorhist_RGB	80.24	47.10	41.164	0.4493	35.72
2	Colorhist_HSV	97.28	77.25	61.027	0.1709	19.00
3	Color Auto Correllogrm	94.47	68.87	55.862	0.2658	34.20
4	LBP	89.29	70.22	54.997	0.2247	13.96
5	ULBP	91.97	74.39	57.630	0.1940	12.36
6	CS_LBP	81.48	56.36	46.057	0.3515	19.96
7	Colorhist_HSV + CS_LBP	87.94	62.09	50.771	0.2983	24.16
8	Colorhist_HSV + ULBP	94.28	77.07	59.472	0.1710	13.32
9	LECoP	98.98	87.51	65.949	0.0837	8.04
10	Proposed Method	**99.64**	**90.41**	**67.304**	**0.0648**	**6.28**

Table 8 Performance measures of the proposed method with different weighted factors for natural image databases

Weightage Factor (Color, Texture)	Corel 1 K					Corel 5 K					Corel 10 K				
	APR	ARR	F Measure	ANMRR	TMRE	APR	ARR	F Measure	ANMRR	TMRE	APR	ARR	F Measure	ANMRR	TMRE
(1,1)	80.15	50.94	39.50	0.39	7.35	62.60	30.23	25.25	0.63	36.56	53.57	23.10	19.73	0.71	73.57
(1,2)	81.01	50.98	39.65	0.39	7.33	64.80	31.53	26.36	0.61	36.45	55.57	24.28	20.68	0.70	73.77
(2,1)	77.94	49.21	38.28	0.40	7.62	59.63	28.31	23.66	0.65	37.50	50.79	21.50	18.40	0.73	75.17
(1,3)	80.77	50.17	39.13	0.40	7.38	64.97	31.48	26.39	0.61	36.80	55.96	24.36	20.77	0.70	74.61
(3,1)	76.94	47.75	37.31	0.42	7.82	57.90	27.28	22.84	0.66	38.18	49.20	20.65	17.69	0.74	76.45

Table 9 Performance measures of the proposed method with different weighted factors for color texture image databases

Weightage Factor (Color, Texture)	VisTex					Stex					Color Brodatz				
	APR	ARR	F Measure	ANMRR	TMRE	APR	ARR	F Measure	ANMRR	TMRE	APR	ARR	F Measure	ANMRR	TMRE
(1,1)	98.67	81.56	65.56	0.13	5.25	93.44	68.53	57.25	0.24	42.44	99.64	90.41	67.30	0.06	6.28
(1,2)	99.02	85.07	67.09	0.10	3.75	94.22	69.87	58.14	0.23	44.38	99.59	89.95	67.14	0.06	6.40
(2,1)	97.89	78.05	63.69	0.16	7.19	91.80	65.34	55.19	0.27	49.63	99.51	89.25	66.84	0.07	7.80
(1,3)	97.89	86.59	67.73	0.08	3.31	94.07	69.44	57.93	0.23	50.81	99.35	88.60	66.60	0.07	7.12
(3,1)	97.58	76.60	62.81	0.18	8.19	90.64	63.20	53.75	0.29	55.25	99.39	87.96	66.26	0.08	8.96

5 Conclusion and future work

This paper proposes a novel method for content based image retrieval by integrating color and texture information in an image. It combines global and local characteristics to get the features. This approach highlights the effect of combining color and intensity information of an image. For this purpose, we use inter-channel relationship between all possible permutations of H, S and I. To get the local feature descriptor, diagonally symmetric pattern followed by GLCM is applied. The proposed method has been evaluated on a total of six image databases; three belong to natural image database: Corel-1 K, Corel-5 K and Corel-10 K while three others belong to color texture image databases: VisTex, STex and Color Brodatz. Performance measures: APR, ARR, F-Measure, and ANMRR are evaluated. The results show the proposed method is better than existing methods in terms of all four performance measures. This paper proposes a new performance measure TMRE, which is applied to all the image databases for all the methods. When we compare the TMRE value of the proposed method with other methods, out of six image databases considered, in four databases, the method yields minimum value. The paper also suggests applying some weighted factor to color and texture information individually to improve the performance, based on the dataset.

Future work The work can be extended by using different pre-trained CNNs': AlexNet, GoogleNet, VGG, ResNet, etc. The work can be extended further to include extreme learning machine (ELM) which would provide a generalized performance of the neural network with fewer user intervention and comparatively faster training. In using distributed frameworks and Map-Reduce implementations, especially HADOOP and SPARK, has given rise to the popularity of dealing with tons of images. Faster results may be achieved by using these technologies.

References

1. Al-Shemarry MS, Li Y, Abdulla S (2019) An Efficient Texture Descriptor for the Detection of License Plates from Vehicle Images in Difficult Conditions. IEEE Trans Intell Transp Syst. https://doi.org/10.1109/TITS.2019.2897990
2. Aptoula E, Lefevre S (2009) Morphological Description of Color Images for Content-Based Image Retrieval. IEEE Trans Image Process 18(11):2505–2517. https://doi.org/10.1109/TIP.2009.2027363
3. Bhunia AK, Bhattacharyya A, Banerjee P, Roy PP, Murala S (2018) A Novel Feature Descriptor for Image Retrieval by Combining Modified Color Histogram and Diagonally Symmetric Co-occurrence Texture Pattern. Preprint Submitted. arXiv preprint arXiv:1801.00879.
4. Chen JJ, Rong CS, Grimson WEL, Liu JL, Shiue DH (2011) Object Segmentation of Database Images by Dual Multiscale Morphological Reconstructions and Retrieval Applications. IEEE Trans Image Process 21(2):828–843. https://doi.org/10.1109/TIP.2011.2166558
5. Chun YD, Kim NC, Jang IH (2008) Content-based image retrieval using multiresolution color and texture features. IEEE Trans Multimedia 10(6):1073–1084. https://doi.org/10.1109/TMM.2008.2001357
6. Clausi DA (2002) An analysis of co-occurrence texture statistics as a function of grey level quantization. Can J Remote Sens 28(1):45–62. https://doi.org/10.5589/m02-004
7. Connolly C, Fleiss T (1997) A study of efficiency and accuracy in the transformation from RGB to CIELAB color space. IEEE Trans Image Process 6(7):1046–1048. https://doi.org/10.1109/83.597279
8. Daschiel H, Datcu M (2005) Design and Evaluation of Human–Machine Communication for Image Information Mining. IEEE Trans Multimedia 7(6):1030–1046. https://doi.org/10.1109/TMM.2005.858383
9. Dong-Chen, Abdelmounaime Safia, Multiband Texture (MBT) database, https://multibandtexture.recherche.usherbrooke.ca/index.html (Accessed 14 April 2019).
10. Gnaneswara Rao N, Sravani T, Vijaya Kumar V (2014) OCRM: Optimal Cost Region Matching Similarity Measure for Region Based Image Retrieval. Int J Multimed Ubiquit Eng 9(4):327–342. https://doi.org/10.14257/ijmue.2014.9.4.34

11. Gonzalez RC, Woods RE (2018) Digital Image Processing – 4th Edition. Pearson, New York, USA
12. Haralick RM, Shanmugam K (1973) Textural features for image classification. IEEE Trans Syst Man Cybern 3(6):610–621. https://doi.org/10.1109/TSMC.1973.4309314
13. Heikkila M, Pietikainen M, Schmid C (2006) Description of interest regions with center-symmetric local binary patterns. In Computer vision, graphics and image processing Springer, Berlin, Heidelberg. 58–69. https://doi.org/10.1016/j.neucom.2015.03.015.
14. Hu R, Barnard M, Collomosse J (2010) Gradient field descriptor for sketch based retrieval and localization. IEEE International Conference on Image Processing, 1025–1028.
15. Hu RX, Jia W, Ling H, Zhao Y, Gui J (2014) Angular pattern and binary angular pattern for shape retrieval. IEEE Trans Image Process 23(3):1118–1127. https://doi.org/10.1109/TIP.2013.2286330
16. Huang J, Kumar SR, Mitra M, Zhu WJ, Zabih R (1997) Image indexing using color correlograms. Computer Vision and Pattern Recognition Proceedings, IEEE Computer Society Conference, 762–768
17. Joblove GH, Greenberg D (1978) Color Space for Computer Graphics. Program of Computer Graphics, Cornell University, 20–25.
18. Karargyris A, Siegelman J, Tzortzis D, Jaeger S, Candemir S, Xue Z, Santosh KC, Vajda S, Antani S, Folio L, Thoma GR (2015) Combination of texture and shape features to detect pulmonary abnormalities in digital chest X-rays. Int J Comput Assist Radiol Surg 11(1):99–106. https://doi.org/10.1007/s11548-015-1242-x
19. Kekre HB, Thepade S, Das RKK, Ghosh S (2012) Image Classification using Block Truncation Coding with Assorted Color Spaces. Int J Electr Comput Syst Eng 44(6):0975–8887. https://doi.org/10.5120/6265-8418
20. Konstantinidis K, Gasteratos A, Andreadis I (2005) A Image retrieval based on fuzzy color histogram processing. Opt Commun 248(4–6):375–386. https://doi.org/10.1016/j.optcom.2004.12.029
21. Roland Kwitt, Salzburg Texture Image Database, http://www.wavelab.at/sources/STex/, (Accessed 14 April 2019).
22. Li B, Sun F, Zhang Y (2019) Building Recognition based on Sparse Representation of Spatial Texture and Color Features Digital Object Identifier. Special Section On Theory, Algorithms, And Applications Of Sparse Recovery 7: 37220–37227. DOI: https://doi.org/10.1109/ACCESS.2019.2905304.
23. Lin CH, Chen RT, Chan YK (2009) A smart content-based image retrieval system based on color and texture feature. Image Vis Comput 27(6):658–665. https://doi.org/10.1016/j.imavis.2008.07.004
24. Linde Y, Buzo A, Gray R (1980) An algorithm for vector quantizer design. IEEE Trans Commun 28(1):84–95. https://doi.org/10.1109/TCOM.1980.1094577
25. Lipo LZ, Lin WW (2011) Semisupervised Biased Maximum Margin Analysis for Interactive Image Retrieval. IEEE Trans Image Process 21(4):2294–2308. https://doi.org/10.1109/TIP.2011.2177846
26. Liu L, Chen J, Zhao G, Fieguth P, Chen X, Pietikainen M (2019) Texture Classification in Extreme Scale Variations using GANet. IEEE Trans Image Process DOI: arXiv:1802.04441v1.
27. Guang-Hai Liu et al., Corel-10k dataset, http://www.ci.gxnu.edu.cn/cbir/Dataset.aspx (Accessed 14 April 2019).
28. Martínez JC, Hidalgo JMS, Jimenez PMM, Sanchez D (2017) Fuzzy Color Spaces: A Conceptual Approach to Color Vision. IEEE Trans Fuzzy Syst 25(5):1264–1280. https://doi.org/10.1109/TFUZZ.2016.2612259
29. Mathew SP, Balas VE, Zachariah KP (2015) A content-based image retrieval system based on convex hull geometry. Acta Polytechnica Hungarica 12(1):103–116. https://doi.org/10.12700/APH.12.1.2015.1.7
30. Murala S, Maheshwari RP, Balasubramanian R (2012) Local tetra patterns: a new feature descriptor for content-based image retrieval. IEEE Trans Image Process 21(5):2874–2886. https://doi.org/10.1109/TIP.2012.2188809
31. Naidu RR, Jampana P, Sastry CS (2016) Deterministic Compressed Sensing Matrices: Construction via Euler Squares and Applications. IEEE Trans Signal Process 64(14):3566–3575. https://doi.org/10.1109/TSP.2016.2550020
32. Jens-Rainer Ohm, Leszek Cieplinski, Heon Jun Kim, Santhana Krishnamachari, B. S. Manjunath, Dean S. Messing, Akio Yamada (2001) The MPEG-7 Color Descriptors, In Manjunath B.S. et al. (ed) Introduction to MPEG 7: Multimedia Content Description Language, John Wiley, and Sons, England, pp. 187–202.
33. Ojala T, Pietikainen M, Maenpaa T (2002) Multiresolution gray-scale and rotation invariant texture classification with Local Binary Patterns. IEEE Trans Pattern Anal Mach Intell 24(7):971–987. https://doi.org/10.1109/TPAMI.2002.1017623
34. Osowski S (2002) Fourier and wavelet descriptors for shape recognition using neural networks-a comparative study. Pattern Recogn 35(9):1949–1957. https://doi.org/10.1016/S0031-3203(01)00153-4
35. Alex Sandy Pentland and Ted Adelson, VisTex Dataset, http://vismod.media.mit.edu/pub/VisTex/, (Accessed 14 April 2019).
36. Quellec G, Lamard M, Bekri L, Cazuguel G, Roux C, Cochener B (2010) Medical Case Retrieval from a Committee of Decision Trees. IEEE Trans Inf Technol Biomed 14(5):1227–1235. https://doi.org/10.1109/TITB.2010.2053716

37. Ramos J, Kockelkorn TTJP, Ramos I, Ramos R, Grutters J, Viergever MA, Ginneken BM, Campilho A (2016) Content-Based Image Retrieval by Metric Learning From Radiology Reports: Application to Interstitial Lung Diseases. J Biomed Health Inform 20(1):281–292. https://doi.org/10.1109/JBHI.2014.2375491
38. Santosh KC, Antani S (2018) Automated Chest X-Ray Screening: Can Lung Region Symmetry Help Detect Pulmonary Abnormalities? IEEE Trans Med Imaging 37(5):1168–1177. https://doi.org/10.1109/TMI.2017.2775636
39. Santosh KC, Wendling L, Antani S, Thoma GR (2016) Overlaid arrow detection for labeling regions of interest in biomedical images. IEEE Intell Syst 31(3):66–75. https://doi.org/10.1109/MIS.2016.24
40. Santosh KC, Vajda S, Antani S, Thoma GR (2016) Edge map analysis in chest X-rays for automatic pulmonary abnormality screening. Int J Comput Assist Radiol Surg 11(9):1637–1646. https://doi.org/10.1007/s11548-016-1359-6
41. Singha M, Hemachandran K (2012) Content based image retrieval using color and texture. Signal Image Process 3(1):39–57. https://doi.org/10.5121/sipij.2012.3104
42. Singha M, Hemachandran K (2012) Content Based Image Retrieval using Color and Texture. Signal Image Process 3(1):239–242. https://doi.org/10.1109/ICSIP.2010.5697476
43. Smeulders AWM, Worring M, Santini S, Gupta A, Jain R (2000) Content-based image retrieval at the end of the early years. IEEE Trans Pattern Anal Mach Intell 22(12):1349–1380. https://doi.org/10.1109/34.895972
44. Strat TM (1993) Employing contextual information in computer vision. DARPA93, 217–229
45. Swain MJ, Ballard DH, Rochester (1991) Color Indexing. Int J Comput Vis 7(1):11–32. https://doi.org/10.1007/BF00130487
46. Tominaga S (1992) Color classification of natural color images. Color Res Appl 17(4):230–239. https://doi.org/10.1002/col.5080170405
47. Vajda SA, Karargyris A, Jaeger S, Santosh KC, Candemir S, Xue Z, Antani S, Thoma G (2018) Feature Selection for Automatic Tuberculosis Screening in Frontal Chest Radiographs. J Med Syst 42(8):146. https://doi.org/10.1007/s10916-018-0991-9
48. Verma M, Raman B, Murala S (2015) Local extrema co-occurrence pattern for color and texture image retrieval. Neurocomputing 165:255–269. https://doi.org/10.1016/j.neucom.2015.03.015
49. Wan X, Kuo CCJ (1998) A new approach to image retrieval with hierarchical color clustering. IEEE Trans Circuits Syst Video Technol 8(5):628–643. https://doi.org/10.1109/76.718509
50. Wang JZ, Modeling objects, Concepts, Aesthetics, and Emotions in Big Visual Data. http://wang.ist.psu.edu/docs/home.shtml (Accessed 14 April 2019).
51. Wang XY, Zhang BB, Yang HY (2012) Content-based image retrieval by integrating color and texture features. Multimed Tools Appl 68(3):545–569. https://doi.org/10.1007/s11042-012-1055-7
52. Xia Z, Zhu Y, Sun X, Qin Z, Ren K (2015) Towards Privacy-Preserving Content-Based Image Retrieval in Cloud Computing. IEEE Trans Cloud Comput 6(1):276–286. https://doi.org/10.1109/TCC.2015.2491933
53. Zhang B, Gao Y, Zhao S, Liu J (2010) Local derivative pattern versus Local Binary pattern: face recognition with high-order local pattern descriptor. IEEE Trans Image Process 19(2):533–544. https://doi.org/10.1109/TIP.2009.2035882
54. Zheng Y, Jiang Z, Zhang H, Xie F, Ma Y, Shi H (2018) Histopathological Whole Slide Image Analysis Using Context-Based CBIR. IEEE Trans Med Imaging 37(7):1641–1652. https://doi.org/10.1109/TMI.2018.2796130

Suresh Kumar Kanaparthi received the B.Tech. degree in Computer Science and Engineering from JNT University, Hyderabad, India in 2009 and M.Tech degree in Computer Science and Technology from Andhra University College of Engineering, Visakhapatnam, India in 2012. Currently he is pursuing the Ph.D degree in the Department of Computer Science and Engineering at National Institute of Technology, Warangal, India. His currently research area includes Computer Vision and Image Processing, Content Based Image Retrieval, and Big Image data.

U.S.N.Raju received the B.E. (CSE) degree from Bangalore University in 1998, M. Tech. (Software Engineering) from JNT University in 2002 and Ph.D.(CSE) from JNT University Kakinada in 2010. He worked in industry for two years and then in academics for 15+ years. Presently he is working in department of Computer Science and Engineering at National Institute of Technology Warangal, Telangana State, India since 6 years. He is a member of IEEE since four years and life member of ISTE, CSI, ISCA and IEI. He has visited UK, China, USA and Taiwan to present his research work.

Shanmukhi P is pursuing B.Tech. (3rd) degree in Computer Science and Engineering from National Institute of Technology Andhra Pradesh, Tadepalligudem, Andhra Pradesh, India. Her research interests are in the fields of Computer Vision, Machine Learning and Image Processing with an emphasis on image retrieval, object detection and classification.

Khyathi Aneesha G is pursuing B.Tech. (3rd) degree in Computer Science and Engineering from National Institute of Technology Andhra Pradesh, Tadepalligudem, Andhra Pradesh, India. Her research interests are in the fields of Computer Vision, Machine Learning and Image Processing with an emphasis on image retrieval, object detection and classification.

Mohammed Ehsan Ur Rahman pursuing B.Tech. (2nd) degree in Computer Science and Engineering from Kakatiya Institute of Technology and Science, Warangal, Telangana State, India. His research interests are in the fields of Computer Vision, Machine Learning and Image Processing with an emphasis on image retrieval, object detection and classification.

Affiliations

Suresh Kumar Kanaparthi [1] · U. S. N. Raju [1] · P. Shanmukhi [2] · G. Khyathi Aneesha [2] · Mohammed Ehsan Ur Rahman [3]

Suresh Kumar Kanaparthi
sureshkonline@gmail.com

P. Shanmukhi
shanmukhipuchakayala@gmail.com

G. Khyathi Aneesha
gkhyathian@gmail.com

Mohammed Ehsan Ur Rahman
ehsan81181@gmail.com

[1] Department of Computer Science and Engineering, National Institute of Technology Warangal, Warangal, Telangana State 506004, India

[2] Department of Computer Science and Engineering, National Institute of Technology Andhra Pradesh, Tadepalligudem, Andhra Pradesh 534101, India

[3] Kakatiya Institute of Technology and Science, Warangal, Telangana State 506015, India