Hierarchical Visual Perception and Two-dimensional Compressive Sensing for Effective Content-based Color Image Retrieval

Yan Zhou[1], Fan-Zhi Zeng[1], Hui-min Zhao[2], Paul Murray[3] and Jinchang Ren[3]

zhouyan791266@163.com, coolhead@126.com, zhaohuimin66@yahoo.com, paul.murray@strath.ac.uk and jinchang.ren@strath.ac.uk (corresponding author)

[1]*Dept. of Computer Science, Foshan University, Foshan, 528000, China*
[2]*School of Electronic and Information Engineering, Guangdong Polytechnic Normal University, Guangzhou, 510665, China*
[3]*Dept. of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, G1 1XW, United Kingdom*

Abstract: Content-based image retrieval (CBIR) has been an active research theme in the computer vision community for over two decades. While the field is relatively mature, significant research is still required in this area to develop solutions for practical applications. One reason that practical solutions have not yet been realized could be due to a limited understanding of the cognitive aspects of the human vision system. Inspired by three cognitive properties of human vision, namely, hierarchical structuring, color perception and embedded compressive sensing, a new CBIR approach is proposed. In the proposed approach, the Hue, Saturation and Value (HSV) color model and the Similar Gray Level Co-occurrence Matrix (SGLCM) texture descriptors are used to generate elementary features. These features then form a hierarchical representation of the data to which a two-dimensional compressive sensing (2D CS) feature mining algorithm is applied. Finally, a weighted feature matching method is used to perform image retrieval. We present a comprehensive set of results of applying our proposed Hierarchical Visual Perception Enabled 2D CS approach using publicly available datasets and demonstrate the efficacy of our techniques when compared with other recently published, state-of-the-art approaches.

Structure Abstract:

Background: Although content-based image retrieval (CBIR) has been an active research theme in the computer vision community for over two decades, there are still challenging problems in properly understanding the process in feature extraction and image matching. Consequently, significant research is still required to develop solutions for practical applications, especially in exploring and making the best using of the cognitive aspects of the human vision system.

Methodology: Motivated by three cognitive properties of human vision, namely hierarchical structuring, color perception and embedded compressed sensing, we proposed a novel framework for CBIR. First, we use a hierarchical approach to perform discrete cubic partitioning of the image in the HSV space. Then, we propose a new hierarchical mapping of the image data through the use of hierarchical operators: SGLCM. These features are then integrated in a 2D CS model, which extracts refined features and suppresses noise. Finally, the resultant features are used for similarity based ranking to perform CBIR.

Results and Conclusions: Experiments were performed using two Corel image datasets, i.e. the Corel-1000 dataset which contains 1000 images in 10 image categories and the Corel-10000 dataset which contains 10000 images in 100 image categories where each category contains 100 images. In comparison to three other state-of-the-art approaches, the proposed method has demonstrated much improved retrieval accuracy, especially for images with rich color contents and detail, yet the computational complexity has been significantly reduced to meet the needs for real-time online applications. The implication of the study is that the exploitation of cognitive properties of our human vision systems in effective CBIR. Future research work can be further explored to address some limitations for optimised parameter setting, adaptive feature fusion and improved machine learning.

Key words: hierarchical visual perception; two-dimensional compressive sensing (2D CS); content-based image retrieval (CBIR).

1. Introduction

Content Based Image Retrieval (CBIR) aims to retrieve images from a database which are similar to each other in terms of their visual contents [1-2]. Over the years, various feature descriptors based on color, texture, and shape information have been proposed to characterize visual elements of image and video data. These features have subsequently been employed to improve the automated search and retrieval of image and videos in CBIR applications. However, selecting and developing methods for correctly identifying and effectively integrating suitable visual features for a specific vision task remains a very challenging problem. In [3], a novel Sparse Multimodal Learning approach was proposed to combine heterogeneous features using joint structured sparsity regularizations. In order to address the correlation problem whilst preserving the benefits of a high recall rate, a Bayesian merging approach was used to down-weight the indexed features in the intersection set [4]. In [5], to overcome the problems of information loss during quantization, SIFT (Scale Invariant Feature Transform) features are combined with color features in a coupled Multi-Index (c-MI) framework. In this approach, weighted feature fusion is used to improve the retrieval accuracy. The joint application of SIFT and color features significantly reduces the impact of false matches and improves recall [6]. However, in the context of CBIR, research is still required to develop robust methods that are capable of accurately describing image contents in an objective way.

In CBIR, color, texture and shape features are considered to be the three most important for identifying images which are similar to each other [1-3]. Shape information can be extracted directly from regions which are segmented based on the color and texture properties of an image. Therefore, both color and texture can be considered as fundamental visual features from which shape information can be obtained. It has been pointed out in [7] that image retrieval algorithms driven by human visual perception can accurately represent the contents of image data and thus offer improved retrieval performance and efficiency. As a result, extracting low-level features by simulating the mechanisms of the primary visual cortex focus heavily in [8-9].

Since the HSV (Hue, Saturation and Value) color model is most consistent with our human perceptions, this color space is widely used when extracting color feature descriptors [10-11]. In [12], a histogram generation approach was proposed for application in the HSV color space. By extracting color features from the HSV space and texture features from the Gray-Level Co-occurrence Matrix (GLCM), a new image retrieval algorithm was developed in [13]. Recently, CBIR technology which exploits the HSV color space has become an area of interest and active research. However, one of the key challenges lies in developing techniques which make use of human visual perception mechanisms to accurately describe image features.

In recent years, some researchers have begun to study frameworks for applying hierarchical approaches to extract image content features. In [14-15] a hierarchical method was used to analyze the fractal dimensions of the grayscale image layers in the data. The authors constructed a vector to represent the content features of the image and this was then used for image classification. The classification effectiveness of this approach depends on whether or not the texture is obvious. Furthermore, in situations where the image texture is fuzzy, this approach may fail. Thus, the techniques need further improvement.

CS theory has been applied extensively since it was proposed in last decade [16-17]. CS exploits the fact that when the measurement matrix satisfies the Restricted Isometry Property (RIP) [18], the original signal can be reconstructed accurately using only a few measurement features. In [19-20], image data was converted from 2D to 1D by a column priority method, and was analyzed using 1D CS theory. However, this method suffers from dimensionality issues and the positional information that exists among image pixels in the 2D spatial domain is lost. To address these issues, researchers extended 1D CS to 2D CS and constructed a 2D CS measurement model to analyze images. In [21], a 2D CS measurement model was proposed to compress and reconstruct images using an iterative gradient descent method. In low-field MRI systems, a 2D CS method was introduced in [22] which used cross-sampling and self-calibrated off-resonance correction. More recently, it has been found that 2D CS actually is embedded in several stages of our human vision system [23]. This has inspired us in our work to combine 2D CS with other visual features for improved CBIR.

Deep learning has also attracted recent attention in image retrieval applications [25, 27]. However, it is unclear as to whether or not features extracted from deep learning offer any improvement over conventional ones or not. It is however anticipated that improved similarity learning may help to enhance the performance if combined with deep learning [25, 27]. While this is beyond the scope of the work presented here, it will be considered in our future research.

Motivated by three cognitive properties of human vision, namely hierarchical structuring [8-9], color perception [10-11] and embedded compressed sensing [23], we proposed a novel framework for CBIR. First, we use a hierarchical approach to perform discrete cubic partitioning of the image in the HSV space. Then, we propose a new hierarchical mapping of the image data through the use of hierarchical operators: SGLCM. HSV features provide color based similarity measurements while SGLCM features represent texture based local structure. These features are then integrated in a 2D CS model which extracts refined features and suppresses noise. Finally, the resultant features are used for similarity based ranking to perform CBIR.

The remainder of the paper is organized as follows: In Section 2, a 2D CS model and reconstruction procedure are presented. Section 3 discusses techniques for extracting hierarchical HSV features based on 2D CS. By fusing multiple features, the overall image similarity is computed and a hierarchical HSV image retrieval framework based on 2D CS is constructed and described in Section 4. In Section 5, we discuss and analyze the experimental results before drawing some conclusions in Section 6.

2. 2D CS model Description

Given a 1D signal $x \in R^N$, the traditional 1D CS measurement can be defined as follows:

$$y = \Phi \cdot x = \Phi \cdot \Psi \cdot a = \Theta \cdot a \qquad (1)$$

where $x = \sum_{i=1}^{N} a_i \cdot \Psi_i = \Psi \cdot a$, Ψ is a matrix composed of orthogonal bases $\Psi_i (i = 1, 2, \ldots, N)$. If there are exactly $K \left(K << N \right)$ nonzero coefficients in the transform vector $a \in R^N$, then the signal x is known as a sparse signal. $\Phi \in R^{M \times N} (M << N)$ is the CS measurement matrix, $y \in R^M$ is the measurement vector and $\Theta = \Phi \cdot \Psi$ denotes the sensing matrix.

When the RIP coefficient δ_k of the sensing matrix Θ satisfies $\delta_{2k} \leq \sqrt{2} - 1$, the 1D sparse signal x can be reconstructed from the measurement vector y by solving the following optimization problem [18]:

$$\min_x \| \Psi^T x \|_1 \quad s.t. \quad y = \Phi \cdot x \quad (2)$$

The above equation is an ℓ -norm convex optimization problem. Many algorithms have been proposed for solving the problem in Eq.(2) such as: Orthogonal Matching Pursuit (OMP); and Sparsity Adaptive Matching Pursuit (SAMP) [25, 28]. In [28], a modified SAMP algorithm based on Regularized Backtracking was proposed. This performed more efficiently than the traditional SAMP algorithm and resulted in an improvement in the reconstruction quality too.

The CS approach allows the original signal to be reconstructed precisely using just a few measurement features. This indicates that the measurement vector y can therefore be used as representive features of the original signal x. Unfortunately, when 1D CS is applied to 2D images by column priority conversion, there are usually has two problems. These are

(1)The increasing dimensions of the measurement matrix will lead to significant computational complexity for sparse signal reconstruction and,

(2) The positional relationship between image pixels in the 2D spatial domain is lost during the conversion.

To overcome the problems stated above, a 2D CS model is proposed in [29]. Let $\Phi_1,\ \Phi_2 \in R^{M\times N}$ denote the row and column of a CS measurement matrix, respectively. Now, the 2D CS measurement model can be defined as:

$$Y = \Phi_1 \cdot X \cdot \Phi_2^T \qquad (3)$$

where $X \in R^{N\times N}$ is a 2D image and $Y \in R^{M\times M}$ represents the 2D CS measurement vector. In general, any natural image can be converted to a sparse signal by applying e.g. the: Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT) or Discrete Wavelet Transform (DWT). This means that the coefficient matrix $S = \Psi^T \cdot X \cdot \Psi \in R^{N\times N}$ will become sparse, where Ψ is a matrix composed of DCT bases. In a similar fashion to 1D CS, the original 2D signal X can be reconstructed from the 2D measurement vector Y if the RIP is satisfied. In practice, the DCT and DWT are more widely used than the DFT, especially for block-based image coding. Once obtained by applying either transform, the sparse coefficients can be readily employed in the CS framework to allow reconstruction of the original signal.

A stretching operator $Vec(\cdot)$ is introduced and arranged by column priority, i.e. $\overline{X} = Vec(X)$, $\overline{Y} = Vec(Y)$ and $\overline{X}, \overline{Y} \in R^{N^2}$.

Reconstructing a 2D signal is equivalent to solving the following optimization problem [18]:

$$\min_{x} \| \Psi \otimes \Psi \overline{X} \|_0 \quad s.t.\ \ \overline{Y} = \Phi_2 \otimes \Phi_1 \cdot \overline{X} \qquad (4)$$

where $\otimes$ is the Kronecker matrix product. If Φ_1, Φ_2 are standard Gaussian random matrixes [29], it is possible for $\Phi_2 \otimes \Phi_1$ to satisfy the RIP condition with a high probability - close to 1. Provided that the measurement matrix is chosen correctly, the original signal X can be reconstructed precisely from the 2D CS measurement vector Y. Thus, in the context of CBIR, the CS measurement matrix Y can be considered as one class of features which accurately describe the original image.

In our proposed approach, the image is first processed using hierarchical operators. Then, each resultant hierarchical mapping matrix is processed further to compute a corresponding 2D CS measurement. Let's now assume that Y_i is the resultant hierarchical CS measurement vector and Y_i' is the CS measurement vector of some query image that is already in the database. Given the previous statement, the difference $\Delta Y_i = Y_i - Y_i'$ can be readily used as a robust evaluation metric for determining image similarity in order to facilitate fast and accurate image retrieval.

3. Extraction of hierarchical HSV features in 2D CS

In general, image retrieval is performed by exploiting both color and texture features which can be extracted directly from the image data. In terms of robustness, color features are known to have the advantage that they are not affected by rotation, scaling or other similar transformations. The color feature of an image is usually represented by a color histogram which simply requires quantization of the data in the selected color space. In this paper, the HSV color space is used because it is perceptually more uniform than other color spaces. The image is therefore first processed using a hierarchical mapping in HSV space before SGLCM is applied in the cubic space. The procedure of feature extraction is described in the following steps.

Step1: Color conversion from RGB to HSV space

The color components of each pixel in a 2D image $X \in R^{N\times N}$ in the RGB color space can be denoted by (R,G,B). Similarly, the color components of an image pixel in the HSV color space can be denoted by (H,S,V).

Let $m_x = \max(R,G,B)$, $m_n = \min(R,G,B)$ and $m_d = m_x - m_n$. The conversion of an image from RGB space to HSV space can now be defined as follows:

$$V = m_x$$

$$S = \begin{cases} m_d / m_x, & if \quad m_x != 0 \\ 0, & if \quad m_x == 0 \end{cases}$$

$$H = \begin{cases} 0, & if (m_x == m_n) \\ 60(G-B)/m_d, & if (R == m_x) \\ 60(B-R)/m_d + 120, & if (G == m_x) \\ 60(R-G)/m_d + 240, & if (B == m_x) \end{cases} \qquad (5)$$

Step2: Cube discretization in HSV color space

Let's now assume that the intervals of the three color components are $[0,\overline{H}],[0,\overline{S}],[0,\overline{V}]$ in the HSV space. This means that the three coordinates of HSV space can be segmented individually into point sets $\{H_i\}_0^{L_1}$, $\{S_i\}_0^{L_2}$, $\{V_i\}_0^{L_3}$ as follows:

$$\begin{cases} H_0 = 0 < H_1 < H_2 ... < H_{L_1} = \overline{H} \\ S_0 = 0 < S_1 < S_2 ... < S_{L_2} = \overline{S} \\ V_0 = 0 < V_1 < V_2 ... < V_{L_3} = \overline{V} \end{cases} \qquad (6)$$

When computing 72-layer HSV features, $L_1 = 8$, $L_2 = L_3 = 3$, and the specific parameters are defined as

$$\begin{cases} \{H_i\}_0^{L_1} = [0,20,45,75,155,190,270,295,360] \\ \{S_i\}_0^{L_2} = \{V_i\}_0^{L_3} = [0,0.2,0.65,1.0] \end{cases} \qquad (7)$$

If computing 256-layer HSV features, we have $L_1 = 16$, $L_2 = L_3 = 4$, and the specific parameters are given by:

$$\begin{cases} \{H_i\}_0^{L_1} = \begin{bmatrix} 0, 15, 25, 45, 55, 80, 108, 140, 165, \\ 190, 220, 255, 275, 290, 315, 330, 360 \end{bmatrix} \\ \{S_i\}_0^{L_2} = \{V_i\}_0^{L_3} = [0, 0.15, 0.4, 0.75, 1.0] \end{cases} \quad (8)$$

With the parameters given above, the HSV color space can be divided into small cubic grids as shown in Fig. 1. For one cubic grid V_{ijk} = $\{(H, S, V)^T\}$, we have $H_{i-1} \le H < H_i$, $S_{j-1} \le S < S_j$, $V_{k-1} \le V < V_k$, where $i = 1, 2, \ldots, L_1$, $j = 1, 2, \ldots, L_2$ and $k = 1, 2, \ldots, L_3$.

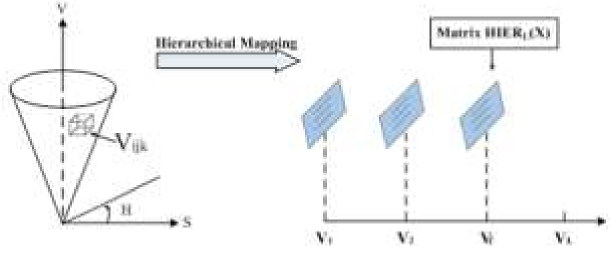


Figure 1. HSV space segment and hierarchical mapping

Step3: Hierarchical HSV mapping matrix

The Hierarchical HSV mapping matrix is computed in two steps. Firstly, the cubic sequence can be obtained by row priority arrangement $V_l \mid l \in [1, L], L = L_1 \times L_2 \times L_3$. For $V_l = V_{ijk}$, i, j, k are determined as $i = l / (L_1 \times L_2) + 1$, $j = (l\%(L_1 \times L_2)) / L_3 + 1$ and $k = (l\%(L_1 \times L_2))\%L_3$. Secondly, hierarchical mapping operators are defined in the cube V_l :

$$HIER_l(i, j) = \begin{cases} 1, & if (H(i, j), S(i, j), V(i, j))^T \in V_l \\ 0, & \text{else} \end{cases} \quad (9)$$

$$(l = 1, 2, \ldots, L; i, j = 1, 2, \ldots, N)$$

where (i, j) denotes the pixel of the image under consideration. Finally, the hierarchical HSV mapping matrix $HIER_l(X) = (HIER_l(i, j))_{N \times N}$ is obtained.

Step4: Extraction of hierarchical HSV features

The hierarchical HSV mapping matrix $HIER_l(X)$ reflects the distribution of the locations of image pixels whose color components are contained within the same cube, V_l of the HSV color space. $HIER_l(X)$ is therefore a 2D sparse signal provided that an appropriate cube discretization is chosen in the HSV color space. As a result, using a 2D CS model and choosing standard Gaussian random matrices $\Phi_1, \Phi_2 \in R^{M \times N}$, the CS measurement vector Y_l can be calculated as:

$$Y_l = \Phi_1 \cdot HIER_l(X) \cdot \Phi_2^T \in R^{M \times M}, l = 1, 2, \ldots L \cdot \quad (10)$$

In Eq.(10), Y_l reflects the hierarchical features of the original image in HSV color space and, as such, it is called a hierarchical HSV feature. This is used as one class of image content features for image retrieval.

Step5: Extraction of hierarchical texture features by SGLCM

With the direction parameter θ and the distance parameter d, the SGLCM $P(\theta, d)$ can be computed in the HSV space as follows:

$$P(\theta, d) = (P_{l_1 l_2}(\theta, d))_{L \times L} \quad (12)$$

$$P_{l_1 l_2}(\theta, d) = \#\{(i_1, j_1), (i_2, j_2) \in N \times N\}. \quad (13)$$

#(•) denotes the total number of set elements, and $l_1, l_2 \in [1, L]$, $\theta = \theta_i \mid i \in [1, L_4], d = d_j \mid j \in [1, L_5]$.

The pixels (i_1, j_1), (i_2, j_2) satisfy the equations below:

$$(H(i_1, j_1), S(i_1, j_1), V(i_1, j_1))^T \in V_{l_1} \quad (14)$$

$$(H(i_2, j_2), S(i_2, j_2), V(i_2, j_2))^T \in V_{l_2} \quad (15)$$

$$\| (i_1, j_1)^T - (i_2, j_2)^T \|_2 \le d, arctg(\frac{j_2 - j_1}{i_2 - i_1}) = \theta \quad (16)$$

If the parameters θ and d take on different directions and distances, the matrix $P(\theta, d)$ can be viewed as an extension of traditional GLCM. This extension is called SGLCM and is considered to reflect hierarchical features related to image texture. Since SGLCM is a 2D sparse signal, the hierarchical texture features PY_l, which we will refer to as “SGLCM texture features”, can be extracted using the following 2D CS measurement model

$$PY_l = PY_{ij} = \Phi_1 \cdot P(\theta_i, d_j) \cdot \Phi_2^T \in R^{M \times M} \quad (17)$$

where $l = (i - 1) \times L_4 + j$, $i \in [1, L_4], j \in [1, L_5], l \in [1, L]$.

If $l > L_4 \times L_5$, then $PY_l = 0$.

Step6: Extraction of traditional texture features by GLCM

The procedure for extracting traditional texture features may be described as follows. Firstly, a GLCM is constructed based on the direction and distance among image pixels. Secondly, statistical features including: energy, entropy, contrast, uniformity, etc. are extracted from the GLCM. Assuming the graylevel of a given image is n_g, the traditional GLCM is defined as:

$$\tilde{P}(\theta, d) = (\tilde{P}_{l_1 l_2}(\theta, d))_{n_g \times n_g} \quad (18)$$

$$\tilde{P}_{l_1 l_2}(\theta, d) = \#\{((i_1, j_1), (i_2, j_2)), i_1, j_1, i_2, j_2 \in [1, N]\} \quad (19)$$

In Eq.(19), the pixels (i_1, j_1), (i_2, j_2) satisfy the following constraint conditions:

$$g(i_1, j_1) = l_1, g(i_2, j_2) = l_2 \quad (20)$$

$$\sqrt{(i_2 - i_1)^2 + (j_2 - j_1)^2} = d, arctg(\frac{j_2 - j_1}{i_2 - i_1}) = \theta \quad (21)$$

where $l_1, l_2 = 0, 1, \ldots, n_g - 1$, $\theta = \theta_1, \theta_2, \ldots;$

$d = d_1, d_2, \ldots$, and $g(i_1, j_1)$ denotes the grey degree of pixel.

Depending on the image characteristics and computational complexity, we select several texture features. These include: energy (ASM), contrast (CON), uniformity (IDM) and entropy (ENT) which can be used to synthesize the final texture features based on GLCM.

ASM is used to measure the uniformity of texture and can be defined as:

$$ASM = \sum_{i=0}^{n_g-1}\sum_{j=0}^{n_g-1}(\tilde{P}_{ij}(\theta,d))^2 \tag{22}$$

CON is used to measure the expectation for graylevel difference:

$$CON = \sum_{i=0}^{n_g-1}\sum_{j=0}^{n_g-1}(i-j)^2\,\tilde{P}_{ij}(\theta,d) \tag{23}$$

IDM is used to measure the local change of image texture and to reflect the homogeneity of texture:

$$IDM = \sum_{i=0}^{n_g-1}\sum_{j=0}^{n_g-1}\frac{\tilde{P}_{ij}(\theta,d)}{1+(i-j)^2} \tag{24}$$

ENT is used to measure the disorder degree of image texture and to embody the intensity of texture:

$$ENT = \sum_{i=0}^{n_g-1}\sum_{j=0}^{n_g-1}\tilde{P}_{ij}(\theta,d)\log(\tilde{P}_{ij}(\theta,d)) \tag{25}$$

If we choose eight directions such as θ=0^0, 45^0, 90^0, 135^0, 180^0, 225^0, 270^0, 315^0, and let d=2, we obtain different results when extracting the aforementioned four features. Once computed, the average parity of these features can be used to create the final texture feature vector namely “GLCM texture features”.

$$G = [G_1, G_2, G_3, G_4] = [ASM, CON, IDM, ENT] \tag{26}$$

4. System architecture and image similarity calculation

The hierarchical HSV feature extraction and retrieval framework is shown in Figure 2.

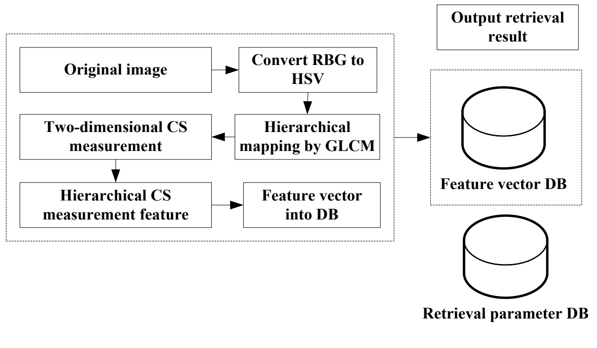


Figure 2. Image retrieval framework based on hierarchical HSV feature

The image retrieval framework shown in Figure 2 can be summarized as follows:

Algorithm: Retrieval Algorithm Based on Two-dimensional Compressive Sensing

Inputs: the standard Gaussian random matrix $\Phi_1, \Phi_2 \in R^{M\times N}$ and the hierarchical number *L*.

Output: A set of images contained within the database which are deemed similar to the query example image *T'* by our proposed approach.

Initial conditions: A query example image *T'* and an image database which contains at least one image *T*.

Hierarchical Procedure:

Step1: *Hierarchical HSV Feature Extraction*. For the query example image *T'*, we first convert the data from RGB format into the HSV color space using Eq. (5). Then, the hierarchical mapping matrix is computed using Eq. (9). The hierarchical HSV features $Y_l^{'} \mid l \in [1, L]$ are then calculated according to the 2D CS measurement model using Eq. (10).

Step2: *Hierarchical Texture Feature Extraction*. The next step is to calculate the SGLCM from the HSV color space. Then, we extract the hierarchical texture features $PY_i'(i = 1, 2, \ldots, L)$ according to Eq. (17). Once this is completed, we compute standard texture features *G_i*' (*i*=1,2,3,4) using the traditional GLCM following Eq. (22-26).

Step3: *Retrieval of candidate image T from database for comparison*. Having computed the hierarchical measurement features for the query image *T'*, the precomputed hierarchical features for candidate image T should be extracted from the database. $Y_i, PY_i \mid i \in [1, L]$ and traditional texture features $G_i \mid i \in [1,4]$ which describe image *T* are therefore retrieved in this step.

Step4: *Comparison of retrieved and query images*. Having computed the hierarchical color and texture features from the query image *T'* in Step 1 and retrieved the corresponding features for a candidate image *T* in the database, an image comparison can be performed. The differences among the hierarchical measurement and traditional texture features which describe the images are computed as follows:

$$\begin{cases} Sim(Y) = 1/(1+\sum_{i=1}^{L}\|\Delta Y_i\|^2) \\ Sim(P) = 1/(1+\sum_{i=1}^{L}\|\Delta PY_i\|^2) \\ Sim(G) = 1/(1+\sum_{i=1}^{L}\|\Delta G_i\|^2) \end{cases} \tag{27}$$

where $\Delta Y_i = Y_i - Y_i'$, $\Delta PY_i = PY_i - PY_i'$ with $i \in [1, L]$ and $\Delta G_i = G_i - G_i^{'} \mid i \in [1,4]$.

Step5: *A normalized image similarity score*

The differences between images *T* and *T'* are quantified and normalized using:

$$H = \frac{\lambda_1 Sim(Y) + \lambda_2 Sim(P) + \lambda_3 Sim(G)}{\lambda_1 + \lambda_2 + \lambda_3}\ . \tag{28}$$

Here, the non-negative weighted coefficients λ_1, λ_2

and λ_3 can be set to different values according to different combinations of retrieval models. In this paper, for different experiments, we have $\lambda_1 \in [0.5, 0.8]$, $\lambda_2 \in [0.2, 0.4]$ and $\lambda_3 \in [0.1, 0.3]$ where these parameters have been determined empirically.

Step6: *Image similarity ranking*
The database images which are compared with the query image T' are ranked based on their similarity to T' as computed in Step 5. The database image which is most similar is ranked 1st down to the image deemed to be least similar which is ranked last.

5. Results and analysis

All algorithms were implemented and tested using Matlab2012 running on a PC with the following specifications: **CPU**: Intel(R) I5-4200U4*2.4GHz, **RAM**: 4GB DDR3L, **OS**: Windows7 SP1 of 32 bits.

Experiments were performed using two Corel image datasets [33]. One image database used was the Corel-1000 dataset which contains 10 image categories (landscapes, horses, elephants, human beings, bus, flowers, buildings, mountains, food and dragons) and each category contains 100 images. The other image database used was the Corel-10000 dataset. This is larger than Corel-1000 and consists of 100 image categories where each category contains 100 images.

5.1 Measurement matrix construction

The Gaussian Random Measurement Matrix (GRMM) is used widely in CS. The elements Φ_{ij} of the matrix should be independent of each other and must satisfy a normal distribution i.e. zero mean with variance equal to $1/\sqrt{M}$.

$$\Phi_{i,j} \sim N(0, \frac{1}{\sqrt{M}}) \tag{29}$$

The advantage of using the GRMM is that it satisfies the RIP condition with a high probability using fewer measurement features [34]. For a given signal of length N and sparsity K, only $M \geq c \cdot K \cdot log(N/K)$ measurement features are required to accurately reconstruct the signal, where c is a small constant. In this paper, when the number of discrete image layers is 256, N=256, M=64. When the number of discrete image layers is 72, then N=72, M=36. At the same time, the GRMM is normalized by columns.

5.2 Image feature extraction

In this section we observe the results of applying the methods discussed in Section 3 to extract hierarchical HSV features from two image categories (horses and flowers) of the Corel-1000 data set. Two randomly selected images from each category are shown in Fig. 3.

Since the background color in these examples is fairly uniform, and the background areas are large, the hierarchical features extracted from the image pairs are highly correlated with each other as shown in Fig. 4. Additionally, the similarity between the features from different groups of images is low. These results and observations verify the effectiveness of the proposed approach for identifying images which are similar to each other while retaining the ability to differentiate these from images which are quite different.

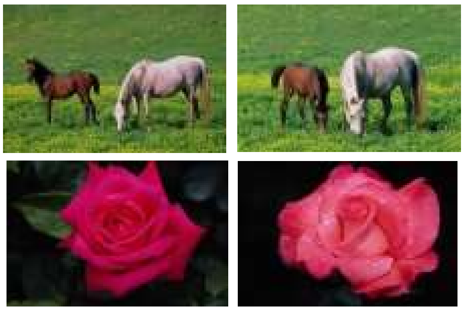


Fig. 3. Two pairs of randomly selected horse (top) and rose (bottom) images.

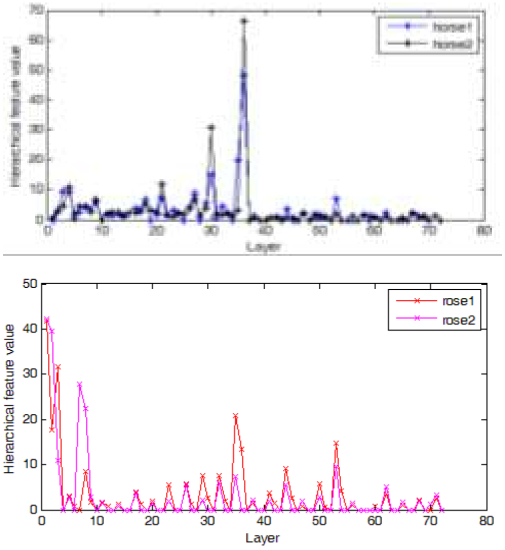


Fig. 4. Extracted hierarchical HSV features from the two horse images (top) and two rose images (bottom).

5.3 Algorithm evaluation

The effectiveness of image retrieval depends on 1) the performance of the feature extraction algorithms used to extract descriptive features from the data and 2) the accuracy and reliability of the similarity computation used to compare images based on the features extracted. We evaluate the retrieval performance of CBIR based on the proposed approach by analysing: precision and recall rates, and F-Scores as defined below. We also measure the image retrieval time and use this in our comparisons.

Let: A denote the total number of relevant images in retrieval result; B denote the total number of images in retrieval result; and C represent the total number of relevant images in the database.

- Precision - The retrieval precision is defined as

$$precision = A/B \tag{30}$$

- Recall - The retrieval recall is defined as

$$recall = A/C \tag{31}$$

- F-Score - By combining the precision and recall, a quota called F-Score is obtained. The F-Score measures the image retrieval accuracy and is defined as:

$$\text{F-Score} = 2 \times \frac{precision \times recall}{precision + recall} \qquad (32)$$

5.4 Retrieval performance analysis

In the following experiments, all 10 categories of the Corel-1000 dataset are used. For each category, the average precision is computed for the top 20 images returned in retrieval result. Additionally, the average recall and the retrieval time are also computed from the top 100 images in retrieval result.

Experiment 1: Retrieval performance comparison between different HSV features

In this experiment, the average precision, recall and F-Score were computed and analyzed for both 72-layer and 256-layer HSV features. The image retrieval time of these approaches was also measured and the accuracy of both methods for all categories can be compared in Figure 5. Additionally, the analysis in Table 1 demonstrates that the precision and recall rates are improved when 256-layer HSV features are used. However, the increase in the number of layers also results increases the associated retrieval time.

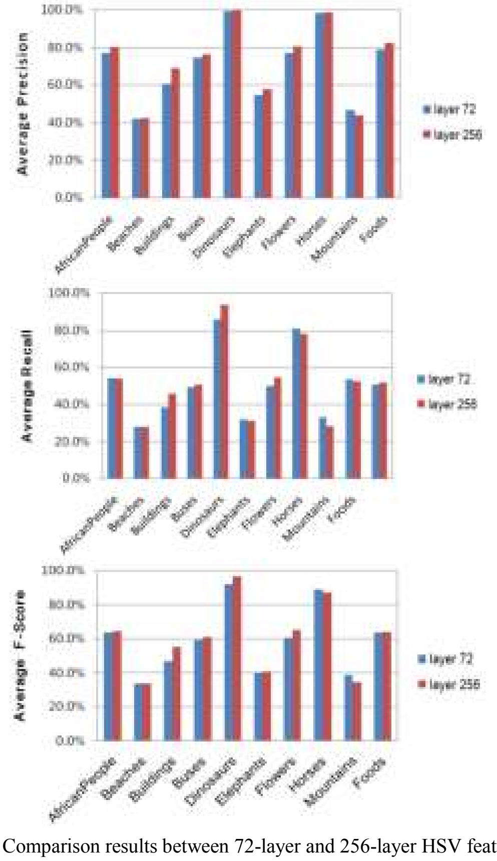


Fig. 5. Comparison results between 72-layer and 256-layer HSV features in terms of precision (top), recall (middle) and F-score (bottom).

Table 1. Comparison result for retrieval performance between layer-72 and layer-256 HSV features

	Precision	Recall	F-Score	Retrieval Time
Layer-72	70.9%	50.4%	58.7%	0.089s
Layer-256	73.0%	51.6%	60.1%	0.093s

Experiment 2: Retrieval performance comparison between hierarchical HSV and texture features

In this experiment, 72-layer and 256-layer HSV features were combined with SGLCM and GLCM texture features in different ways and used to perform image retrieval. The four retrieval models used were:

- **Model-1**: 72-layer HSV+GLCM texture features
- **Model-2**: 256-layer HSV+GLCM texture features
- **Model-3**: 72-layer HSV+SGLCM texture features
- **Model-4**:256-layer HSV+SGLCM texture features

Again, average precision, recall and F-Score were calculated for each of the four retrieval models when applied to the Corel-1000 dataset. The retrieval time when applying each method was also measured and used along with the performance metrics to analyze the retrieval performance and compare the results. From the results in Table 2 it is evident that the 256-layer HSV feature method always performs best regardless of the type of texture features used. Furthermore, by comparing Model-1 with Model-3 and Model-2 with Model-4, the average precision, recall, and F-Scores indicate that SGLCM texture features perform better than traditional GLCM. This improved performance does however come at a small cost in terms of the associated retrieval time as shown in Table 2.

Table 2. Comparison of retrieval results from different models

	Model-1	**Model-2**	**Model-3**	**Model-4**
Precision	75.6%	78.4%	76.2%	80.1%
Recall	54.8%	56.5%	55.6%	58.8%
F-Score	63.6%	65.6%	64.3%	67.8%
Retrieval Time	0.113s	0.118s	0.156s	0.162s

For each category of images in the Corel-1000 database, the four retrieval models were applied and analyzed based on the average precision, recall, F-Score computed and the associated retrieval time. The results are shown in Figure 6, which demonstrates that, in general, Model-4 (as proposed in this paper) performs better than the others. In fact, this is true for every category of images except those containing buildings. The average precision for horse images reaches 98.6%, and the average precision for Africa, bus, flower and food image categories reaches 80%. The only drawbacks of Model 4 are that 1) the retrieval time for the 256-layer HSV feature extractor is longer than 72-layer, and 2) the retrieval time of SGLCM texture feature approach is longer than the traditional GLCM (see Table 2). However, the difference in retrieval time for Model-4 is negligible when compared to the others – especially for the proposed application domain and given the offered improvement in terms of accuracy and performance.

By observing the results in Figure 6, it also appears that the proposed SGLCM features are better suited for application to images with rich color content and finer details which can be captured by the algorithm and used for retrieval. In contrast, GLCM is more applicable to greylevel images with little color content. As a result, GLCM perform better than SGLCM when applied to the category of images containing buildings which lack color content and appear more like greyscale images.

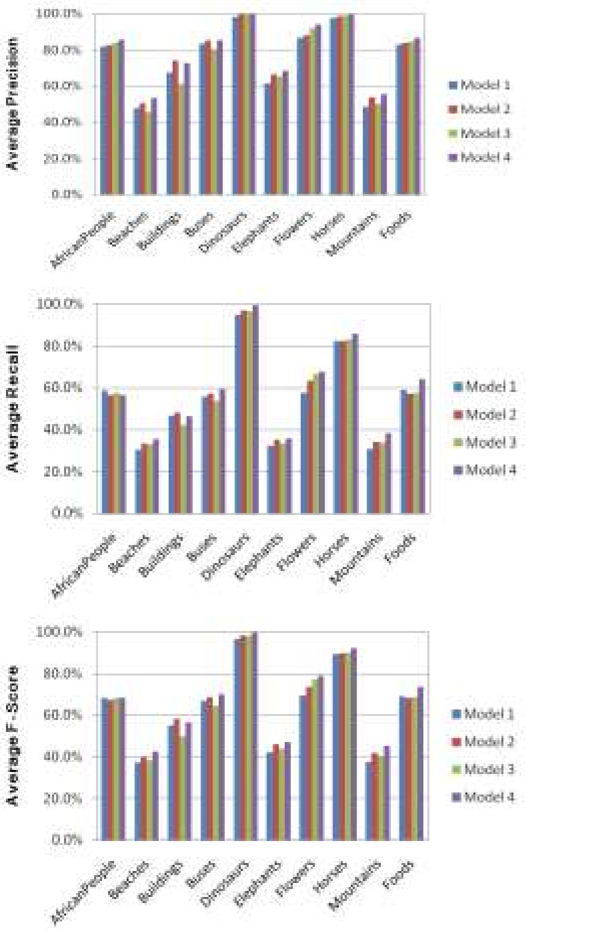


Fig. 6. Comparison result among different models in terms of precision (top), recall (middle) and F-score (bottom).

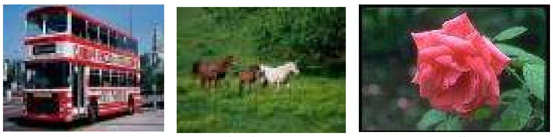
5.5 Retrieval result analysis

Experiment 1: Retrieval results for Corel-1000 dataset

In this experiment, four different models are used to perform image retrieval. These are:

- **Model-A**: 72-layer HSV features only
- **Model-B**: 256-layer HSV features only
- **Model-C**: SGLCM texture+72-layer HSV features
- **Model-D**: SGLCM texture+256-layer HSV features

Here, bus and horse images are chosen as the query example images from the Corel-1000 dataset and the flower image is used as a query example image for the Corel-10000 database.



(a) Bus (b) horse (c) flower

Fig. 7. Three query example images.

(1) Retrieval result for bus image – Corel-1000

The bus image used to query the database is shown in Figure 7(a), and the retrieval results are shown in Figure 8. There are four pages of retrieval results and each page contains 32 images which are displayed to a user. From Figure 8, we can see that if Model-A is used, two irrelevant images will erroneously appear in the first page of retrieval result. However, if Model-B is used, all images in the first page of retrieval result are relevant.

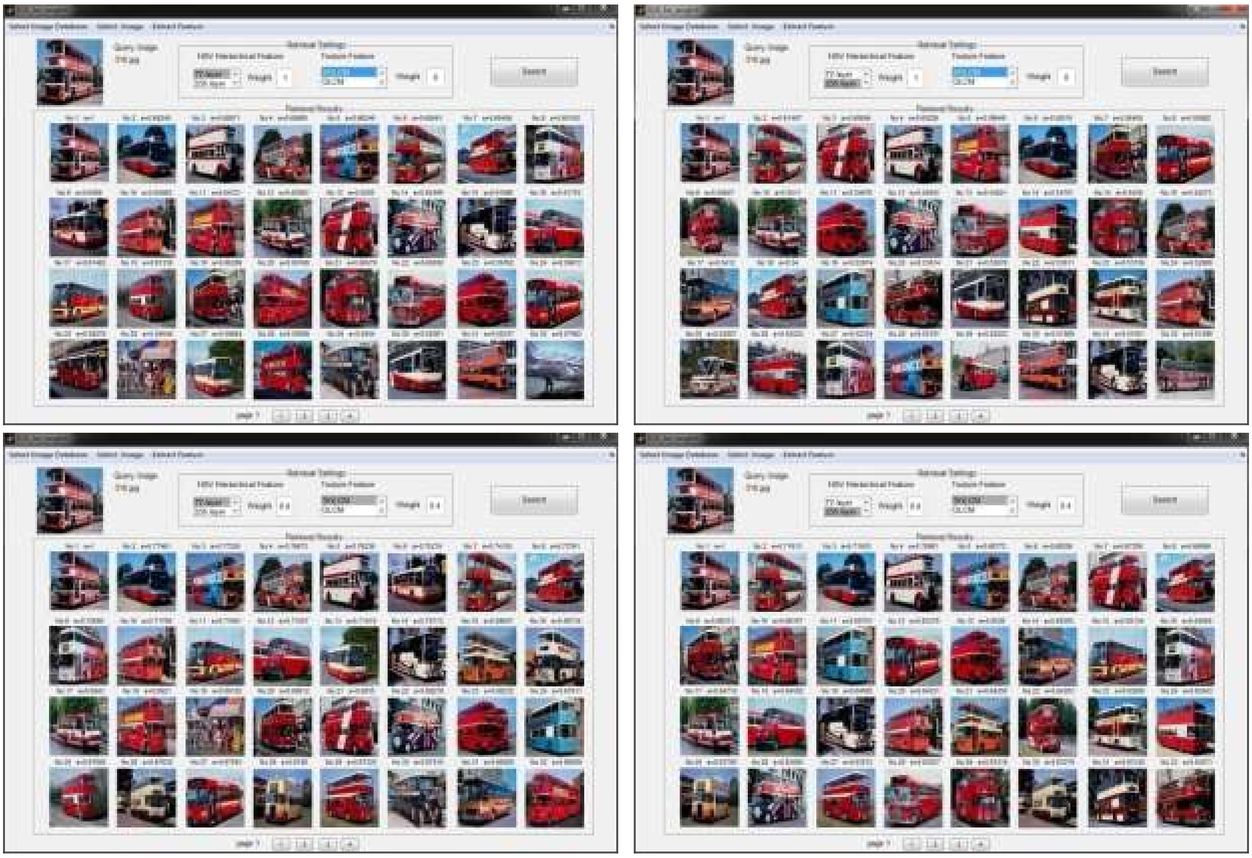


Fig.8. Retrieval results for the bus image from Model-A (top-left), Model-B (top-right), Model-C (bottom-left) and Model-D (bottom-right).

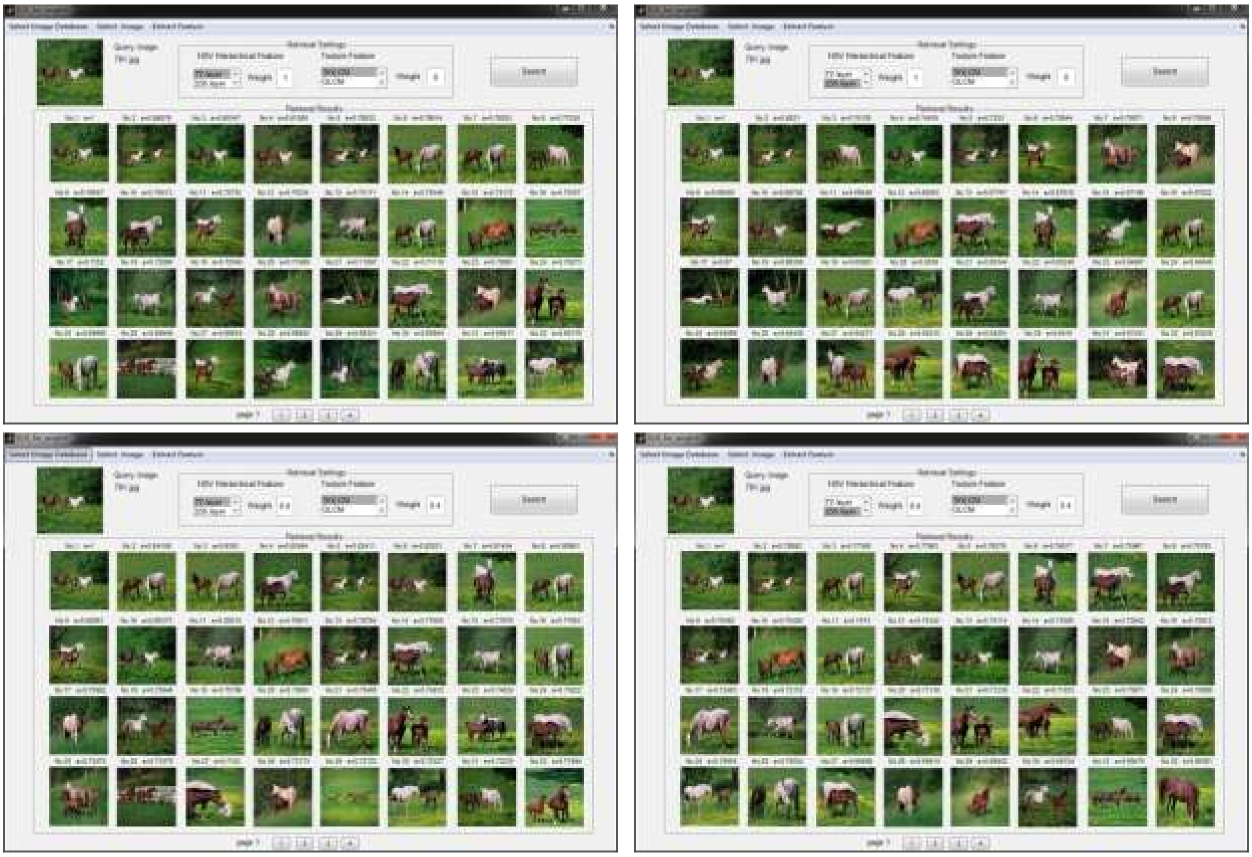


Fig. 9. Retrieval results for the horse image from Model-A (top-left), Model-B (top-right), Model-C (bottom-left) and Model-D (bottom-right).

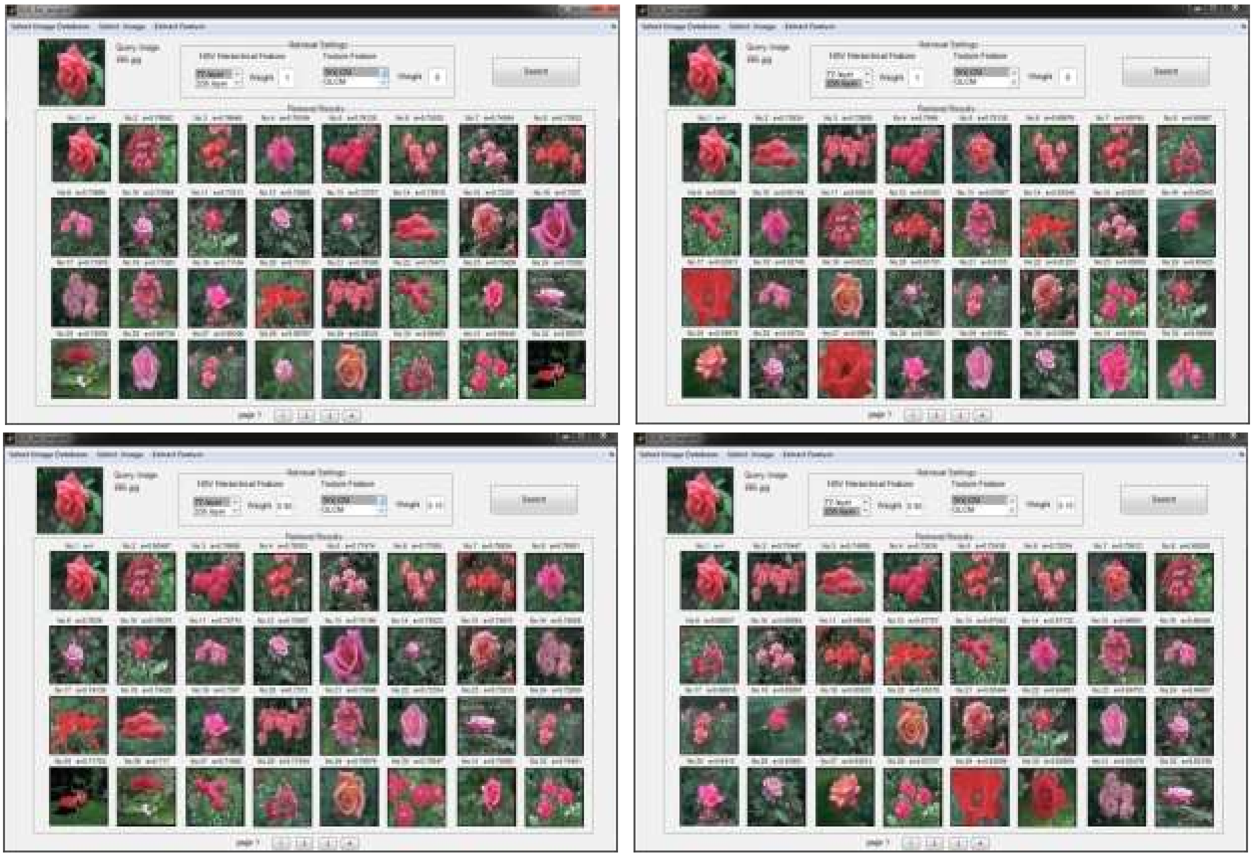


Fig. 10. Retrieval results for the flower image from Model-A (top-left), Model-B (top-right), Model-C (bottom-left) and Model-D (bottom-right).

When Model-C is used, there is one irrelevant image in the first page of retrieval result. Finally, if Model-D is used, all images in the first page of retrieval result are relevant. These results demonstrate that the retrieval accuracy will improve with the increase of hierarchical layers and the infusion of multiple features.

(2) Retrieval result for horse image – Corel-1000

The horse image used to query the database is shown in Figure 7(b), and the retrieval results are shown in Figures 9. Figure 9 illustrates that the retrieval result for the category of images named horse is better than the category of bus. For the above four retrieval models, all images in the first page of retrieval results are relevant.

Experiment 2: Retrieval result analysis for Corel-10000 dataset

In this experiment, flowers were chosen as the query example image from database-2, and the query image used is shown in Figure 7(c). The above four retrieval models (Model-A-D) were used to perform image retrieval using this query image and the results are shown in Figure 10. From the figures, it is clear that if Model-A or Model-C is used, there will be one irrelevant image in the first page of the retrieval results. However, if Model-B or Model-D are used, all images on the first page of retrieval results are relevant.

5.6 Retrieval result analysis

Finally, in this experiment we compare our proposed approach with several leading techniques. This comparison is based on the precision of each technique and its associated retrieval time when applied to all 10 categories of the Corel-1000 dataset. In the Core1-1000 dataset, 20 images for each category were chosen randomly as query images to compute the average precision. By fixing the weights of all images, the retrieval Model-D which infuses 256-layer HSV features and SGLMC texture features was used to obtain the comparison result shown in Table 3. From the table, we can see that the proposed algorithm offers better performance where the average precision achieved reaches 80.1%. Furthermore, for six of the ten possible image categories, the performance of the proposed algorithm is better than all others. The average precision is 3.2% higher than the one proposed in [30], and 24.8% higher than that proposed in [32].

We have also compared the proposed algorithm with all other algorithms [30-32] in terms of retrieval time as shown in Table 3. In spite of the much improved retrieval performance offered, the proposed method also exhibits minimal computational complexity in comparison to those from [30-32] by a factor of up to 30 in some cases.

Table.3. Comparison of retrieval results in terms of precision and running time among different algorithms

Retrieval precision	Our Method	Nishant et al [30]	Malay et al [31]	Wang et al [32]
African	85.5%	74.8%	74.2%	44.0%
Beaches	53.4%	58.2%	59.8%	32.0%
Buildings	72.8%	62.1%	61.8%	52.0%
Buses	85.1%	80.2%	70.0%	60.0%
Dinosaurs	100.0%	100.0%	99.3%	40.0%
Elephants	68.7%	75.1%	81.6%	80.0%
Flowers	94.2%	92.3%	87.3%	57.0%
Horses	99.6%	89.6%	90.4%	75.0%
Mountains	55.7%	56.1%	59.8%	57.0%
Foods	86.5%	80.3%	75.0%	56.0%
Average precision	80.1%	76.9%	75.9%	55.3%
Running time (s)	0.13	1.22	0.47	3.98

As shown in Table 3, the proposed approach provides significant improvements in terms of image retrieval performance for most image categories tested. However, its performance when applied to images in three categories, namely: Beaches, Elephants and Mountains is slightly worse than for the other state-of-the-art methods evaluated. As explained earlier, the main reason for this is that the proposed techniques perform particularly well when applied to images which are rich in colour and contain fine detail. Unfortunately, the large monochrome regions in images contained in these three categories has led to poor discrimination for image retrieval using the proposed approach. However, as expected, for categories which contain images with rich colour information and finer more complicated details, our approach performs the best.

6 Conclusions

In this paper, motivated by the cognitive properties of human vision systems, hierarchical visual structuring, color perception and compressed sensing are used to perform CBIR. Once the image has been converted from RGB to HSV color space, our algorithm performs discrete cubic partitioning. Then, by introducing hierarchical operators and defining SGLCM in the HSV space, the 2D CS measurement model is used to extract two classes of hierarchical features. One of them, known as "hierarchical HSV features" reflects the image's color and the positional relationship among pixels. The other is known as "SGLMC texture features" and these accurately describe and facilitate the comparison of texture features between images. The similarity among images is computed by fusion of the two hierarchical features and the traditional GLMC and SGLCM texture features. Experimental results show that the proposed method offers better performance when compared to several other state-of-the-art techniques.

The proposed approach does however have some limitations. For example, it requires the empirical setting of parameters in the feature fusion step. Furthermore, inconsistent performance was observed when comparing the proposed SGLMC feature extraction and traditional GLMC for different image categories. As a result, adaptive feature fusion and content-based optimized feature selection will be the focus of future investigations aiming to address these shortcomings. In addition, deep learning based feature extraction will also be explored and benchmarked with conventional hand-crafted features in the future. Finally, new approaches such as learning of common visual patterns [24] and granular computing of structured data [35] will also be investigated for improved performance going forward.

7. Compliance with Ethical Standards

We confirm that there are no potential conflicts of interest; also the work involves no human participants and/or animals. All the authors are consent to the submission of the paper.

8. Acknowledgements

The authors would like to thank the editors and the anonymous reviewers for their constructive comments to further improve the quality of this paper. This work is partially supported by the following projects in China: Natural Science Foundation of Guangdong Province (No. 2015A030313635, 2016A030311013), the National Science Foundation of China (61272381), Science and Technology Project of Guangdong Province (No. 2014A010103037), Special Fund for Science and Technology Innovation of Foshan (No. 2015AG10008), Education Department of Guangdong Province (2015KTSCX153, 2014KZDXM060, 2015KGJHZ021) and Outstanding Youth Teacher Training Program of Foshan University (No. FSYQ201411).

References

[1]. G.-H. Liu, J.-Y. Yang et al, Content-based image retrieval using computational visual attention model, Pattern Recognition, 48(8), 2554-2566, 2015.

[2]. L. Zhang, L. Wang, W. Lin. Generalized biased discriminant analysis for content-based image retrieval. IEEE Trans. Syst. Man Cybern. B, 2012, 42(1): 282-290, 2012.

[3]. H. Wang, F. P. Nie, H. Huang, et al. Heterogeneous visual features fusion via sparse multimodal machine. CVPR, 2013, pp.3097-3102.

[4]. L. Zheng, S. Wang, W. Zhou, et al. Bayes merging of multiple vocabularies for scalable image retrieval. CVPR2014, pp.1963-1970, in Columbus, Ohio, USA, June, 2014.

[5]. L. Zheng, S. Wang, Z. Liu, et al. Packing and padding: coupled multi-index for accurate image retrieval, CVPR2014, pp.1947-1954, in Columbus, Ohio, USA, June, 2014.

[6]. X. Wang, M. Yang, T. Cour, et al. Contextual weighting for vocabulary tree based image retrieval, ICCV2011, pp.209-216, in Barcelona, Spain, November, 2011.

[7]. G. P. Qiu et al. Visual guided navigation for image retrieval, Pattern Recogn, 2007, Vol. 40(6), pp.1711-1721.

[8]. G. Papari, N. Petkov. An improved model for surround suppression by steerable filters and multilevel inhibition with application to contour detection. Pattern Recogn, 2011, Vol. 44 (9), pp.1999-2007.

[9]. M. Ursino, G. Emiliano, L. Cara. A model of contextual interaction and contour detection in primary visual cortex. Neural Networks, 2004, Vol. 17(5-6), pp.719-735.

[10]. R. Kimchi. The perception of hierarchical structure. The Oxford Handbook of Perception Organization. Aug. 2015.

[11]. K. G. Gegenfurtner, Cortical mechanisms of colour vision, Nature Reviews: Neuroscience, vol. 4, July 2003, 563-572.

[12]. T. Deselaers, et al, “Features for image retrieval: an experimental comparison,” Information Science, 11(2): 77-107, 2008.

[13]. C. Lai, Y. Chen. A user-oriented retrieval system based on interactive genetic algorithm. IEEE Trans. on Instrumentation and Measurement, 2011, Vol. 60(10), pp.3318-3325.

[14]. Y. Xu, H. Ji. Viewpoint invariant texture description using fractal analysis. Int. J Comput Vis, 2009, Vol.83, pp. 85-100.

[15]. Y. Xu, S. Huang, H. Ji. Scale-space texture description on SIFT-like textons. Computer Vision and Image Understanding, 2012, Vol.116, pp.999-1013.

[16]. E. J. Candès. Compressive sampling. European Mathematical Society. 2006, pp.1433-1452.

[17]. E. Candès, M. Wakin. An introduction to compressive sampling. IEEE Signal Processing Magazine, 2008, Vol. 25(2), pp.21-30.

[18]. E. J. Candès. The restricted isometry property and its implications for compressed sensing. Comptes rendus - Mathématique, 2008, Vol.346 (9), pp.589-592.

[19]. H. Han, L. Gan, S. Liu, et al. A novel measurement matrix based on regression model for block compressed sensing. Journal of Mathematical Imaging and Vision, 2015, Vol. 51(1), pp.161-170.

[20]. B. Han, D. Wu. Image representation by compressed sensing for visual sensor networks. J. Vis Commun., 2010, Vol. 21(4) , pp.325-333.

[21]. G. Chen, D. Li, J. Zhang. Iterative gradient projection algorithm for two-dimensional compressed sensing sparse image reconstruction. Signal Processing, 2014, Vol. 104, pp.15-26.

[22]. D. Tamada. Two-dimensional compressed sensing using the cross-sampling approach for low-field MRI systems. IEEE Trans. on Medical Imaging, 2014, Vol. 33(9) , pp.19 05-1912.

[23]. H. Zhao and J. Ren, Cognitive computation of compressed sensing for watermarking signal measurement, Cognitive Computation, 8(2): 246-260, 2016

[24]. M. Zhao, B. Jiang, B. Luo and J. Tang, “Common visual patterns discovery with an elastic matching model,” Cognitive Computation, in press, 2016

[25]. J. Wan et al, “Deep learning for content-based image retrieval: a comprehensive study,” Proc. 22nd ACM Int. Conf on Multimedia, pp. 157-166, 2014.

[26]. K. Ning, Compressed sensing image processing based on stagewise orthogonal matching pursuit. Sensors & Transducers, 2014, Vol.181 (10), pp.134-140.

[27]. A. Babenko and V. Lempitsky, “Aggregating local deep features for image retrieval**,”** IEEE International Conference on Computer Vision (ICCV), 2015.

[28]. R. Zhao, X. Ren, X. Han, et al. An improved sparsity adaptive matching pursuit algorithm for compressive sensing based on regularized backtracking. Journal of Electronics, 2012, Vol. 29(6), pp.580-584.

[29]. A. Eftekerhar, M. Babaie-Zadeh, H. A. Moghaddam. Two-dimensional random projection, Signal Processing, 2011, Vol.91(7) , pp.1589-1603.

[30]. N. Shrivastava, V. Tyagi. An efficient technique for retrieval of color images in large databases. Computers and Electrical Engineering, 2014, Vol. 11(9), pp.1 -14.

[31]. M. K. Kundu, M. Chowdhury, S. R. Bulò. A graph-based relevance feedback mechanism in content-based image retrieval. Knowledge-Based Systems, 2015, Vol. 73, pp.254–264.

[32]. X.Y. Wang, H.Y. Yang, Y.W. Li, F.Y. Yang, Robust color image retrieval using visual interest point features of significant bit-planes, Digital Signal Process, 2013, Vol. 23(4) , pp.1136-1153.

[33]. J. Li, J. Z. Wang, Automatic linguistic indexing of pictures by a statistical modeling approach, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 9, pp. 1075-1088, 2003

[34]. H. Dehghan, R. M. Dansereau and A.D.C. Chan, Restricted Isometry Property on Banded Block Toeplitz Matrices with Application to Multi-Channel Convolutive Source Separation, IEEE Trans. Signal Processing, 63(21): 5665-5676, 2015.

[35]. F. M. Bianchi, S. Scardapane, A. Rizzi, A. Uncini and A. Sadeghian, “Granular computing techniques for classification and semantic characterization of structured data,” Cognitive Computation, 8(3): 442-461, 2016