Research Article

Application of Artificial Bee Colony Optimization Algorithm for Image Classification Using Color and Texture Feature Similarity Fusion

D. Chandrakala¹ and S. Sumathi²

¹ Department of Computer Science and Engineering, Kumaraguru College of Technology, Tamil Nadu, Coimbatore 641049, India

² Department of Electrical and Electronics Engineering, PSG College of Technology, Tamil Nadu, Coimbatore 641004, India

Correspondence should be addressed to D. Chandrakala, vishnuchandra2k3@gmail.com

Received 9 September 2011; Accepted 24 October 2011

Academic Editor: C. Chen

Copyright © 2012 D. Chandrakala and S. Sumathi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the advancement in image capturing device, the image data is being generated in high volumes. The challenging and important problem in image mining is to reveal useful information by grouping the images into meaningful categories. Image retrieval is extensively required in recent decades because CBIR is regarded as one of the most effective ways of accessing visual data. Conventionally, the way of searching the collections of digital image database is by matching keywords with image caption, descriptions and labels. Keyword based searching method provides very high computational complexity and user has to remember the exact keywords used in the image database. Instead, our paper proposes image retrieval system with Artificial Bee Colony optimization algorithm by fusing similarity score based on color and texture features of an image thereby achieving very high classification accuracy and minimum retrieval time. In this scheme, the color is described by color histogram method in HSV space and texture represented by contrast, energy, entropy, correlation and local stationary over the region in an image. The proposed Comprehensive Image Retrieval scheme fuses the color and texture feature based similarity score between query and all the database images. The experimental results show that the proposed method is superior to keywords based retrieval and content based retrieval schemes with individual low-level features of image.

1. Introduction

Images are rich in content and they facilitate international exchanges without language restrictions. Retrieving images from a vast collection has become one of the interesting challenges and has drawn the attention of researchers towards the development of retrieval approaches based on low-level features of image. In recent years, CBIR system has very broad and important application in areas including military affairs, medical science, education, architectural design, and agriculture. The traditional Keywords-Based Image Retrieval (KBIR) [1] retrieves images by matching keywords with image annotations, labels, often returning irrelevant results and consuming more time. In addition, it is clearly partial to describe content-rich multimedia information with a small amount of text [2]. The CBIR system deals with the low-level features to describe the

content of the image and it breaks through the limitation of traditional text query techniques. The main idea of image retrieval system is to analyze image information by low level features of an image [3] which include color, shape, structure, and texture of object and to set up feature vectors of an image as its index. The implementation of image retrieval system based on single feature describes the content of an image from a specific angle that might be suitable for simple images, besides relating an image with a single feature that is incomplete. Representing an image with multifeatures from multiangles is expected to achieve better results since different features reflect the uniqueness of the image. In the proposed method, the similarity scores of color and texture features are combined in a suitable way to achieve the intended results. The stages of the Comprehensive Image Retrieval (CIR) scheme are depicted in Figure 1.

FIGURE 1: Stages of comprehensive image retrieval scheme.

Initially the query image is resized to 150×150 in the preprocessing stage and it is followed by the feature extraction stage. In this paper, the feature extraction stage derives the color and texture features from the image. Color features are extracted using color histogram method and the texture features are derived from cooccurrence matrix of the image. Different features reflect the different characteristics of the image; if those features are integrated reasonably, the retrieval process will be a complete one. In our proposed system, the similarity between query image and each of the images from target database are derived and they are normalized. The normalized similarity values of color and texture are combined using fusion algorithm and fusion weights are assigned adaptively by Artificial Bee Colony optimization algorithm [4] to improve the image retrieval performance.

Yue et al. [5] proposed a technique to retrieve images by combining the similarity score between query image and the database image based on color and texture features. The basic experimental idea is extracted from this paper. The local and global color histogram is used as color feature and texture feature vectors are derived from cooccurrence matrix of the image. The fusion weights of color and texture are assigned equal values. As fusion weights were not optimum, the effectiveness of retrieval performance was inconsistent. Prasad et al. [6] proposed a technique to retrieve images by region matching using a combined feature index based on color, shape, and location within the framework of MPEG-7. Simulation results show that the image retrieval was done only based on dominant regions within each image and hence the retrieval performance was incomplete. Chun et al. [7] introduced a CBIR method based on an efficient combination of multiresolution color and texture features. Autocorrelograms of the hue and saturation component of image in HSV space are considered as color feature and value component from Block Difference of Inverse Probabilities (BDIP)-Block Variation of Local Correlation Coefficients (BVLC) extracted in the wavelet transform domain were considered as texture feature. Experimental results for various image resolutions provided considerable retrieval accuracy only with high-dimension feature vectors.

Tai and Wang [8] proposed a novel image feature called color-texture correlogram which is the extension of color correlogram, and the simulation performance shows that the retrieval of images using spatial correlation of color-texture feature vectors gives good retrieval accuracy. Almost all the related work concentrated on extraction methodologies of a range of low-level features of the image and comprehension of those features to have efficient retrieval routine without considering the optimum fusion weight value. In [9], two-dimensional histograms of the CIELab

chromaticity coordinates were chosen as color features and variance extracted by discrete wavelet frames analysis were preferred as texture features. The mean and covariance of RGB values and the energies of Discrete Cosine Transform (DCT) coefficients are used as color and texture features in [10]. A number of approaches have been proposed in literature to obtain substantial classification accuracy by combining low-level features. Typical examples of the CBIR retrieval systems include QBIC [11], Virage [12], Photobook [13], VisualSEEK [14], Netra [15] and SIMPLiCity [16].

The proposed ABC algorithm-based similarity score fusion technique is capable of estimating a high classification accuracy and reasonable retrieval rate with less retrieval time by means of optimal fusion weights. Image retrieval results are analyzed based on Multifeature Similarity Score Fusion (MSSF) without fusion weight optimization and proposed MSSF with ABC optimization algorithm. The sections of the paper are organized as follows: Section 2 deals with the details and the scheme of comprehensive retrieval model with color and texture features. Section 3 covers the experimental results and performance analysis of the multifeature similarity score fusion-based image retrieval method and its hybrid version with ABC algorithm. Section 4 depicts the overall conclusion and the scope of future work.

2. Architecture of Comprehensive Image Retrieval Scheme

Image classification and retrieval utilizes the visual content of an image, to search for similar images in large-scale image databases, according to user's interest. The CBIR problem is motivated by the need to search the exponentially increasing space of image and video databases efficiently and effectively. In a typical image retrieval system, features related to visual content such as color, shape, structure, or texture are extracted from the query image; the similarity between the individual or set of features of the query image and that of each image in a benchmark database is computed, and target images which are most similar to the query image are retrieved. In our proposed system, the content of an image is analyzed in terms of combined low-level features such as color and texture extracted from the query image and used for retrieval. The proposed image retrieval scheme with the stages such as image preprocessing, feature extraction, similarity measurement with normalization, multi-feature similarity score fusion, and fusion weight optimization are illustrated in Figure 2.

2.1. Data Preprocessing and Feature Extraction. Images with various sizes are reduced to 150×150 , since resizing is an

FIGURE 2: Framework of comprehensive image retrieval scheme.

important task that must often be done when processing digital images before features are derived. The first step in feature extraction process is to pull out the image features to a distinguishable extent. When the input data to an algorithm is too large to be processed and if the data has no much information, then the input data will be transformed into a reduced representation set of features and also named as feature vector. Transforming the input data into the set of features is called feature extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input. Color information is the most intensively used feature for image retrieval because of its strong correlation with the underlying image objects. The texture is another widely used feature in image retrieval, which is intended to capture the granularity and repetitive patterns of surfaces within an image [3]. Color and texture feature extraction methods are elaborated in this section.

2.1.1. Color Feature Extraction. The goal of color feature extraction is to obtain compact, perceptually relevant representation of the color content of an image. Color is one of the most widely used visual features and is invariant to image size and orientation [17, 18]. The image retrieval problem is motivated by the need to search the exponentially increasing space of image and video databases efficiently and effectively. The visual content of an image is analyzed in terms of low-level features extracted from the image. One of the most important features that make possible the recognition of images by humans is color. Color is a property that depends on the reflection of light to the eye and the processing of that information in the brain. Colors are defined in three-dimensional color spaces such as RGB (Red, Green, and

Blue), HSV (Hue, Saturation, and Value), or HSB (Hue, Saturation, and Brightness).

In this proposed scheme the color features are extracted using Color Histogram in HSV color space. The steps to find the color feature are as follows.

Step 1. RGB image is transformed to HSV model and the perceived distance between colors is in proportion to Euclidean distance between corresponding pixels in HSV color model.

Step 2. Quantization in terms of color histograms refers to the process of reducing the number of bins by taking colors that are very similar to each other and putting them in the same bin. In the HSV space the Hue is quantized into 16 bins and is among [0, 15], saturation is quantized into 4 bins and is among [0, 3], and value is quantized into 4 bins and is among [0, 3]. Among these, the human cognition about color is mainly based on hue, then saturation and finally value. Thus the quantized results are coded as in

$$C = 16 \times H + 4 \times S + V, \quad (1)$$

where C is a integer between 0 and 255.

Step 3. Obtain the color histogram of an image in HSV color space. Color histogram is a representation of the distribution of colors in an image. For digital images, a color histogram represents the number of pixels that have colors in each of a fixed list of color ranges that span the image's color space, the set of all possible colors.

Step 4. The similarity between color feature values of query image and that of the database image is calculated using

Euclidean distance similarity measure given in (2); the closer the distance, the higher the similarity

$$D(q, s) = \left(\sum_{i=0}^{L-1} (q_i - s_i)^2 \right)^{1/2}, \quad (2)$$

where $q = \{q_0, q_1, \dots, q_{L-1}\}$ is the feature vector of query image, and $s = \{s_0, s_1, \dots, s_{L-1}\}$ is the feature vector of the database images, L is the number of dimensions of image features.

2.1.2. Texture Feature Extraction. Texture is a visual feature that refers to inherent surface properties of an object and their relationship to the surrounding environment. This section proposes a texture feature representation scheme based on image cooccurrence matrix. Cooccurrence matrix is widely used to extract texture feature in gray-scale image and has been shown to be very efficient. The color image will be converted to a gray-scale image by (3) and the number of the gray scale value is 256:

$$Y = 0.29 \times R + 0.589 \times G + 0.114 \times B, \quad (3)$$

where Y is the gray-scale value and R , G , B represent red, green, and blue components, respectively. The cooccurrence probabilities provide a second-order method for generating texture features [18]. These probabilities represent the conditional joint probabilities of all pair wise combinations of gray levels in the spatial window of interest given two parameters: inter pixel distance (δ) and orientation (θ). The probability measure can be defined as

$$Pr(x) = \{C_{ij} \mid (\delta, \theta)\}, \quad (4)$$

where C_{ij} (the cooccurrence probability between gray levels i and j) is defined as

$$C_{ij} = \frac{P_{ij}}{\sum_{i,j=0}^G P_{ij}}, \quad (5)$$

where P_{ij} represents the number of occurrences of gray levels i and j within the given window, given a certain (δ, θ) pair; and G is the quantized number of gray levels. The sum in the denominator thus represents the total number of gray level pairs (i, j) within the window. Statistics applied to the cooccurrence probabilities to generate the texture features are defined in

$$\text{Contrast} = \sum C_{ij} (i - j)^2, \quad (6)$$

$$\text{Energy} = \sum (C_{ij})^2, \quad (7)$$

$$\text{Entropy} = \sum C_{ij} \log C_{ij}, \quad (8)$$

$$\text{Correlation} = \sum \frac{(i - \mu_x)(j - \mu_y)C_{ij}}{\sigma_x \sigma_y}, \quad (9)$$

$$\text{Local Stationary} = \sum C_{ij} |i - j|. \quad (10)$$

The gray-scale quantification is made and the corresponding cooccurrence matrix of size 256×256 is obtained. The statistical properties such as contrast, energy, entropy, correlation, and local stationary are calculated using (6)–(10) to describe the image content. The texture features are extracted in the following five steps.

Step 1. The color image is converted to gray-scale image and the image cooccurrence matrix is derived using (4) and (5).

Step 2. The five statistical properties such as contrast, energy, entropy, correlation, and local stationary are calculated using (6)–(10) in four orientations such as 0° , 45° , 90° , and 135° , so that, totally 20 texture features are obtained.

Step 3. Mean and variance of the above five parameters are taken. The results are the ultimate texture features and are denoted as

$$T = (\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5). \quad (11)$$

Step 4. The similarity value between the query image and that of the database images are calculated using Euclidean distance by (2); the closer the distance, the higher the similarity.

2.2. CIR Based on Color and Texture Feature Similarity Scores. The features such as color, shape, structure, and texture are used to describe the image content from different angles. In image retrieval, usage of additional features provides more information on the image content than the usage of single feature. Hence, in our paper, the similarity scores based on color and texture features are combined with a weighted fusion function for describing every image in the database. Since the physical meanings of different features are different, the similarity value ranges are totally different. Before comprehending the features the similarity scores must be normalized. Normalization is a scaling down transformation of the features to appreciably low values. This is important for the classification algorithm to have better performance with less computation time. The min-max normalization method is the most preferred method in majority of the research work because it yields a steady range of normalized values. It performs a linear transformation on the original data scores. Hence the similarity scores of color and texture are normalized by means of min-max normalization using (8). The normalization procedure is given as follows: let Q be the query image and $\{S_i\}$ be the similarity score where $i = 1, 2, \dots, N$, and N is the number of images in the database. If $\min(S_i)$ and $\max(S_i)$ are the minimum and maximum scores of feature S_i , mapping is done in the interval $[\text{new_min}(S_i), \text{new_max}(S_i)]$. Consequently, every value from the original interval will be mapped into new values using min_max normalization formula which preserves relationships among the original values. Thus, normalized similarity scores are obtained for each image in target database using

$$S_{Ni} = \frac{S_i - \min(S_i)}{\max(S_i) - \min(S_i)}. \quad (12)$$

The weighted fusion algorithm is applied to find comprehensive similarities to improve the retrieval accuracy and precision using

$$S_{Fi} = \frac{S_{Ci}W_C + S_{Ti}W_T}{W_C + W_T}, \quad (13)$$

where S_{Ci} and S_{Ti} are normalized similarity values of color and texture features, W_C and W_T are the weight values for color and texture similarity scores, respectively. In this CIR scheme, the two weight values are approximately equal, that is, the weights of the color and texture are 0.5 and 0.5. The retrieval performance of combined color and texture feature similarity scores method is better than the retrieval methods with single feature.

2.3. Application of Artificial Bee Colony Algorithm for CIR Scheme. During the course of similarity score fusion, a key problem is how to assign the weights of similarity score. It affects directly the retrieval performance of the system. To resolve the problem of obtaining the optimum weight value for color and texture feature similarity score and to implement a fast and robust CIR scheme, Artificial Bee Colony algorithm is used. In ABC algorithm, the position of a food source represents a possible solution to the optimization problem and the amount of nectar of a food source corresponds to the quality of the associated solution. The number of employed bees or the onlooker bees is equal to the number of solutions in the population. At the first step, the ABC generates a randomly distributed initial population (P) of SN solutions where SN denotes the size of employed bees or onlooker bees. Each solution x_i is a D -dimensional vector where D is the number of optimization parameters and $i = 1, 2, 3, \dots, SN$. After initialization, the population of the positions is subject to repeat Maximum Number of Iterations (MNIs) of the search processes of the employed bees, the onlooker bees, and the scout bees. An employed bee produces a modification on the position in bee's memory depending on the local information and tests the amount of nectar at the source. If the amount of nectar of the new one is higher than that of the previous one, the bee memorizes the new position and forgets the old one. Otherwise the bee keeps the position of the previous one in its memory.

The ABC algorithm is shown below.

Step 1. Create an initial population of artificial bees within the search space x_{ij} .

Step 2. Evaluate the fitness of the population using sphere function given in

$$f(x) = \sum_{i=1}^{SN} x_i^2. \quad (14)$$

Step 3. While (stopping criterion not met) where stopping condition is Maximum Number of Iterations (MNI).

- (i) Produce new solutions (food source positions) v_{ij} in the neighborhood of x_{ij} for the employed bees using

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}), \quad (15)$$

where $k \in \{1, 2, 3, \dots, SN\}$ and $j \in \{1, 2, 3, \dots, D\}$.

- (ii) Evaluate the fitness value using (14) and apply the selection process between x_{ij} and v_{ij} using (16) and (17). An artificial onlooker bee chooses a food source depending on the new positions, using

$$P_i = \begin{cases} v_i & \text{if } (f(x_i) \geq f(v_i)), \\ x_i & \text{if } (f(x_i) \leq f(v_i)). \end{cases} \quad (16)$$

In order to select the better nectar position found by an onlooker, O_b is defined as

(iii)

$$O_b = \underset{P_i}{\operatorname{argmin}} f(P_i), \quad 1 \leq i \leq n, \quad (17)$$

where P_i is the best fitness value of the solution i which is proportional to the nectar amount of the food source in the position i and n is the number of food sources which is equal to the number of employed bees.

- (iv) Produce new solutions (new positions) v_{ij} for the onlookers from the solutions x_{ij} selected depending on P_i and evaluate them.
- (v) Determine the abandoned solution (source) x_{ij} , if it exists and replace it with a new randomly produced solution x_{ij} for the scout bee using

$$x_i^j = x_{\min}^j + \operatorname{rand}[0, 1](x_{\max}^j - x_{\min}^j), \quad (18)$$

where $x_{\min}$ and $x_{\max}$ are the lower and upper limit, respectively, of the search scope on each dimension.

- (vi) Memorize the best food source position (solution) achieved so far.

Step 4. End while.

The application of this algorithm for assigning fusion weight in multifeature similarity score fusion method is helped to gain a better image retrieval performance.

3. Experimental Results and Performance Analysis

The experimental analysis and simulation determine the efficiency and capability of the work. In this simulation, results for various modules implemented, namely, image preprocessing, feature extraction, similarity score calculation and normalization, score fusion, and ABC optimization technique are explained to quantify the benefits of image retrieval based on multifeatures. The system is tested on Intel Core 2 Duo (1.66 GHZ, clock speed), 2 GB RAM desktop PC using MATLAB R2000b. For testing the validity of the proposed scheme, a set of 800 images from COREL database library are taken. This dataset was used in various scientific articles for content-based image retrieval systems.

The database contains 10 image classes with 100 images each. The classes are Africa, Beach, Buildings, Buses, Dinosaurs, Flowers, Elephants, Horses, Food, and Mountains. Six images from each class are randomly selected and used as query image. The application of ABC in the CIR scheme creates a significant improvement in the classification accuracy level and reduction in computational time apart from getting consistent results.

3.1. Image Preprocessing and Feature Extraction. The images from the COREL image library are resized to 150×150 to make the retrieval process efficient. Then the two features, namely, color and texture features are derived for all the images. Feature extraction is the process of interacting with images and performs extraction of meaningful information of images with descriptions based on properties that are inherent in the images themselves. A Feature extracted from an image is generally represented as a vector of finite dimension. The feature vector dimension is one of the most important factors that determine the amount of storage space, the retrieval accuracy, and the retrieval time [19]. All the RGB images in the database are transformed to HSV space model using simple nonlinear transformation. The transformed images are used to acquire color histogram, and then quantization is done to reduce the number of bins by taking colors that are very similar to each other and putting them in the same bin. The quantized results using (1) in Section 2 are stored in a text file for further processing. The coded value obtained is an integer between 0 and 255. Texture is another widely used feature in content-based image retrieval system. The statistical properties, namely, contrast, energy, entropy, correlation, and local stationary are calculated in all four orientations from cooccurrence matrix. Finally the mean and variance of each parameter are taken as a component of the texture feature. The mean and variance of the statistical properties in all four orientations are tabulated in Table 1 for sample images.

3.2. Similarity Score Calculation and Normalization. In image retrieval, the sum of squares of the difference between feature vectors of query image and the image from target database is calculated to get the Euclidean distance. The similarity score between query image and images in database is derived using the Euclidian distance with (2) in Section 2.1.1. Sample similarity scores are tabulated in Table 2; the closer the distance, the higher the similarity. If the query and database image are same then similarity score is equal to zero. If they are different the Euclidian distance is the similarity score and is stored in a text file for further processing. Since the physical meanings of different features are different, and value ranges are totally different, similarity scores of different features cannot be compared. So, before fusing the multifeature similarity scores, they should be normalized. The normalization is done using (12) given in Section 2.2 and the values obtained for few samples of color-based similarity score is tabulated in Table 3.

3.3. Standard and Evolutionary-Based Fusion of Similarity Scores. The individual features cannot express the semantic information of an image sufficiently. Hence, the color and texture features are combined to realize comprehensive image retrieval. The global similarity function S_{Fi} is computed as a weighted sum of the similarities shown in (13) in Section 2.2. In our experiment, the fusion weights W_C and W_T are obtained from Artificial Bee Colony algorithm.

ABC Parameters:

Number of colonies (P) = 50,

Number of food sources (SN) = 25,

Stopping condition (Number of iteration for foraging) = 100,

Upper Limit (ub) = 2,

Lower Limit (lb) = -2,

Number of dimensions (D) = 2,

Fitness Function (Sphere Function) = $f(x) = \sum_{i=1}^{SN} x_i^2$,

Best fitness selection, see (16) where v_i is new solution and x_i is the old solution.

The performance is analyzed using Confusion matrix method which contains information about actual and predicted categories of a retrieval system. The query images are presented to the CIR scheme with fusion weight optimization using ABC algorithm and the true-positive, true-negative, false-positive, and false-negative values are tabulated in Table 4. The precision rate, recall rate, classification accuracy, and error rate are calculated using (19) for all six query images of bus class and are shown in Table 4. The class wise performances are recorded in Table 5 for 800 images from COREL image library. The experiment shows that the classification accuracy is more by a factor of 4% than color-based retrieval and 10% more compared to texture-based image retrieval method

$$\begin{aligned} \text{Precision Rate} &= \frac{a}{a+b}, \\ \text{Recall Rate} &= \frac{a}{a+c}, \\ \text{Classification Accuracy} &= \frac{a+d}{a+b+c+d}. \end{aligned} \quad (19)$$

For the horse image, classification accuracy is 0.72% to more than color-based retrieval, 14% more than texture-based retrieval and 1.89% more than Fusion algorithm with equal weights. In our proposed method, the retrieval time is also 0.6 seconds less than the fusion algorithm with equal weights.

The variation in classification accuracy is recorded by increasing the number of retrieved images from 10 to 100, where 100 is the maximum number of images in each class. From Figure 3, it is clear that the classification accuracy of conventional color-based, texture-based, and MSSF with equal weights methods is less compared to our proposed method. For example, when we consider the number of images retrieved to be equal to 100, the classification

TABLE 1: Mean and variance of texture feature derived using cooccurrence matrix for contrast, energy, entropy, correlation, and local stationary in four orientations.

Orientations	Sl. no.	Bus (301.jpg)		Dinosaur (421.jpg)		Elephant (517.jpg)		Horse (726.jpg)	
		Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance
0°	1	0.1283	0	0.9022	0	0.2214	0	0.6007	0
	2	0.8867	0	0.0982	0	0.7918	0	0.4053	0
	3	0.5099	0	0.2811	0	0.4971	0	0.6068	0
	4	1.0000	0	0.0982	0.0102	0.8220	0.0006	0.4096	0.0125
	5	0	0	0	0.0004	0.0087	0.0001	0	0.0003
45°	6	0.0367	0	0.3508	0	0.0497	0	0.5000	0
	7	1.9600	0	1.3200	0	1.9333	0	2.0000	0
	8	0.1414	0	0.9248	0	0.2108	0	0	0
	9	1.0000	0	1.0000	0	1.0000	0	0	0
	10	0	0	0	0	0	0	0	0
90°	11	0.0367	0	0.1872	0	0.5000	0	0.5000	0
	12	2.9400	0	2.4800	0	3.0000	0	3.0000	0
	13	0.1414	0	0.6653	0	0	0	0	0
	14	1.0000	0	1.0000	0	0	0	0	0
	15	0	0	0	0	0	0	0	0
135°	16	0.9689	0	0.0694	0	0.3306	0	0.9928	0
	17	0.1209	0	3.7867	0	2.7221	0	0.0235	0
	18	0.1030	0	0.3004	0	0.6795	0	0.0241	0
	19	0.0178	0.0016	1.0000	0	0.7744	0.0012	0.0016	0.0001
	20	0.0060	0.0478	0	0	-0.0206	0.0066	0	0.0129

TABLE 2: Similarity score between query image and target database images based on color features.

Query image	Database image									
	Bus 1	Bus 2	Bus 3	Bus 4	Bus 5	Bus 6	Bus 7	Bus 8	Bus 9	Bus 10
Bus 1	0	10.18	9.00	7.85	11.66	9.95	10.06	8.34	7.79	8.29
Bus 2	9.14	0	8.76	9.62	9.72	8.44	9.53	8.99	9.61	8.47
Bus 3	8.07	8.55	0	7.77	8.92	8.47	9.30	8.74	8.25	7.31
Bus 4	9.77	9.92	9.19	0	11.49	9.24	10.21	9.44	9.61	8.92
Bus 5	9.59	10.14	9.28	10.71	0	11.20	11.57	10.55	9.00	10.07
Bus 6	10.45	9.11	9.36	8.92	10.48	0	9.16	8.76	10.15	7.70
Bus 7	8.86	10.14	9.07	8.73	11.12	8.53	0	7.70	9.22	7.43
Bus 8	8.22	10.44	9.44	7.92	11.34	8.58	8.68	0	8.43	7.50
Bus 9	8.52	11.62	10.79	8.96	10.30	11.21	11.40	10.03	0	9.48
Bus 10	8.12	9.05	8.45	7.43	9.67	7.52	9.14	7.65	8.10	0

TABLE 3: Normalization of similarity scores for Table 2.

Query image	Database image									
	Bus 1	Bus 2	Bus 3	Bus 4	Bus 5	Bus 6	Bus 7	Bus 8	Bus 9	Bus 10
Bus 1	0	0.87	0.77	0.67	1	0.85	0.86	0.71	0.66	0.71
Bus 2	0.94	0	0.90	0.99	1	0.87	0.98	0.92	0.98	0.87
Bus 3	0.87	0.92	0	0.84	0.95	0.91	1	0.94	0.88	0.79
Bus 4	0.85	0.86	0.79	0	1	0.80	0.88	0.82	0.84	0.77
Bus 5	0.83	0.87	0.80	0.93	0	0.96	1	0.91	0.77	0.87
Bus 6	0.99	0.86	0.89	0.85	1	0	0.87	0.83	0.97	0.74
Bus 7	0.79	0.91	0.82	0.79	1	0.76	0	0.69	0.82	0.66
Bus 8	0.72	0.92	0.83	0.69	1	0.76	0.76	0	0.74	0.66
Bus 9	0.73	1	0.93	0.77	0.88	0.96	0.98	0.86	0	0.82
Bus 10	0.84	0.94	0.87	0.77	1	0.77	0.94	0.79	0.84	0

TABLE 4: Performance of CIR scheme with ABC algorithm for bus image.

Sl. no.	Query image	True positive (a)	False positive (b)	False negative (c)	True negative (d)	Precision rate (%)	Recall rate (%)	Error rate (%)
1	Bus1	59	41	459	41	59	11.39	13.67
2	Bus2	79	21	479	21	79	14.16	7.00
3	Bus3	77	23	477	23	77	13.90	7.67
4	Bus4	56	44	456	44	56	10.94	14.67
5	Bus5	74	26	474	26	74	13.50	8.67
6	Bus6	78	22	478	22	78	14.03	7.33
Average (%)						70.50	12.99	9.83

TABLE 5: Comparison of performance of CIR with ABC optimization and other schemes.

Sl no.	Image class	Color-based retrieval		Texture-based retrieval		CIR with equal weights		CIR with ABC algorithm	
		Classification accuracy (%)	Retrieval time (s)	Classification accuracy (%)	Retrieval time (s)	Classification accuracy (%)	Retrieval time (s)	Classification accuracy (%)	Retrieval time (s)
1	Bus	86.83	4.743	80.11	4.307	88.51	4.982	90.17	4.381
2	Dinosaur	80.72	4.807	95.67	4.487	93.56	4.816	86.89	4.347
3	Elephant	81.83	4.711	82.72	4.506	85.17	4.890	85.72	4.257
4	Rose	78.33	4.789	96.00	4.503	94.89	4.965	86.67	4.282
5	Horse	93.67	4.799	80.61	4.577	92.50	5.033	94.39	4.310
6	Mountain	81.17	4.646	78.94	4.443	82.11	4.915	82.72	4.209

FIGURE 3: Scatter plot of classification accuracy of different retrieval algorithms for bus image based on number of images retrieved.

FIGURE 4: Scatter plot for the relationship between precision and Recall rate for various methods experimented.

accuracy of color-based retrieval method at that point is less by a factor of 10% from our proposed method for bus image.

Precision and recall rate are found to evaluate the performance of the proposed method and the number of returned images is ranging from 10 to 100. For comparison, the image retrieval methods based on color feature, texture feature, and two-feature similarity score fusions with equal weights are implemented. The precision and recall rate relationship of these methods are shown in Table 6. From Table 6, it is obvious that the image retrieval of CIR scheme using ABC algorithm presents precision 10% more than color-based retrieval and 20% more than the texture-based retrieval and 5% more than the CIR scheme with equal weights for almost same recall rate for the bus as the query image. Figure 4

shows that image retrieval based on multifeature similarity score fusion using Artificial Bee Colony algorithm ranked first. Compared with image retrieval method based on color feature, the performance of image retrieval based on texture feature is poor.

The average precision and recall rates for query images for various classes are shown in Table 7. With proposed system, the average precision is 10% more than the color-based retrieval for the bus as query image.

Figure 5 shows the overall performance of image retrieval methods, that is, color based, texture based, and CIR with equal weights and the proposed CIR scheme with ABC algorithm for various image classes considered. Our proposed method gives the average precision rate for bus, elephant, and

TABLE 6: Precision and recall rate of different retrieval algorithms for bus image based on number of retrieved images.

Sl. no.	Number of images Retrieved	Based on color feature		Based on texture feature		CIR with equal weights		CIR with ABC algorithm	
		Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)
1	10	90	2	80	2	95	2	100	2
2	20	85	3	65	3	89	3	94	3
3	30	80	5	53	3	87	5	89	5
4	40	75	6	53	4	80	6	84	6
5	50	72	7	48	5	79	8	82	8
6	60	70	8	47	6	77	9	80	9
7	70	67	9	44	6	74	10	79	10
8	80	65	10	43	7	70	11	76	11
9	90	62	11	41	8	68	11	73	12
10	100	61	12	40	8	66	12	71	13

TABLE 7: Class wise Average Precision and Recall rate based on different methods.

Sl no.	Image class	Color-based retrieval		Texture-based retrieval		CIR with equal weights		CIR with ABC algorithm	
		Average precision (%)	Average recall (%)	Average precision (%)	Average recall (%)	Average precision (%)	Average recall (%)	Average precision (%)	Average recall (%)
1	Bus	60.5	11.54	40.33	8.35	65.50	12.31	70.50	12.99
2	Dinosaur	42.17	8.63	87.00	15.12	80.50	14.31	60.67	11.57
3	Elephant	45.50	9.24	48.17	9.68	55.50	10.83	57.17	11.11
4	Rose	35.00	7.16	88.00	15.28	84.67	14.85	60.00	11.26
5	Horse	81.00	14.41	41.83	8.61	77.50	13.96	83.17	14.68
6	Mountain	43.50	8.76	36.83	7.75	46.33	9.29	48.17	9.64
Average (%)		49.61	9.96	54.69	10.80	66.50	12.59	69.00	11.88
Standard deviation		0.134	0.026	0.207	0.035	0.173	0.022	0.174	0.017

horse classes to be higher by a factor of 5%, 2%, and 6% when compared with CIR method with equal weights. However, it gives average precision rate less than CIR scheme with equal weights by a factor of 20% and 24.67% for dinosaur and rose classes, respectively. In this study, the color-based image retrieval is better because the color differences of different images are more obvious in COREL image library. In addition, the extracted texture feature from this image library may be insufficient to reflect the differences between different classes, which make performance of texture-based image retrieval poor.

4. Conclusion and Future Work

Image data mining has the potential to provide significant information about the images and extract the implicit knowledge stored in the image to the users. The content of an image can be expressed in terms of different features such as color, shape, and texture. These low-level features are extracted directly from digital representation of the image and do not necessarily match the human perception of visual semantics. In our proposed system, only color and texture features are considered and the similarity scores of individual features are fused. Fusion weights in the system implementation are incorporated with latest optimization technique, Artificial Bee Colony Algorithm which makes the image retrieval

FIGURE 5: Classwise average precision of image retrieval based on color, texture, multifeature with equal weights and multifeature with ABC algorithm.

process a unique one. The results demonstrate that the color and texture features are useful in retrieving similar images when a query image is presented. As per analysis, our scheme produces average precision rate of 95% for simple images and 82% for complex images. This model can be enhanced in future, by including more low-level features such as shape and spatial location features. In this paper, while evaluating the fitness of an individual, we considered

only the occurrence frequencies of an image in retrieval result and not the location of it. So, this factor should be taken into account when evaluating the fitness of an individual in future work.

References

- [1] Y. Gong, H. Zhang, H. C. Chuan, and M. Sakauchi, "Image database system with content capturing and fast image indexing abilities," in *Proceedings of the International Conference on Multimedia Computing and Systems*, pp. 121–130, Boston, Mass, USA, May 1994.
- [2] P. Muneesawang and L. Guan, "An interactive approach for CBIR using a network of radial basis functions," *IEEE Transactions on Multimedia*, vol. 6, no. 5, pp. 703–716, 2004.
- [3] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: ideas, influences, and trends of the new age," *ACM Computing Surveys*, vol. 40, no. 2, article 5, pp. 1–60, 2008.
- [4] D. Karaboga and B. Akay, "A comparative study of Artificial Bee Colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.
- [5] J. Yue, Z. Li, L. Liu, and Z. Fu, "Content-based image retrieval using color and texture fused features," *Mathematical and Computer Modelling*, vol. 54, no. 3-4, pp. 1121–1127, 2011.
- [6] B. G. Prasad, K. K. Biswas, and S. K. Gupta, "Region-based image retrieval using integrated color, shape, and location index," *Computer Vision and Image Understanding*, vol. 94, no. 1–3, pp. 193–233, 2004.
- [7] Y. D. Chun, N. C. Kim, and I. H. Jang, "Content-based image retrieval using multiresolution color and texture features," *IEEE Transactions on Multimedia*, vol. 10, no. 6, Article ID 4657457, pp. 1073–1084, 2008.
- [8] X. Y. Tai and L. D. Wang, "Medical image retrieval based on color-texture algorithm and GTI model," in *Proceedings of the 2nd International Conference on Bioinformatics and Biomedical Engineering (iCBBE '08)*, pp. 2574–2578, Shanghai, China, May 2006.
- [9] S. Liapis and G. Tziritas, "Color and texture image retrieval using chromaticity histograms and wavelet frames," *IEEE Transactions on Multimedia*, vol. 6, no. 5, pp. 676–686, 2004.
- [10] H. Permuter, J. Francos, and I. H. Jermyn, "Gaussian mixture models of texture and colour for image database retrieval," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, vol. 3, pp. 569–572, Hong Kong, April 2003.
- [11] W. Niblack, R. Barber, W. Equitz et al., "QBIC project: querying images by content, using color, texture, and shape," in *Proceedings of the Storage and Retrieval for Image and Video Databases, (SPIE 1908)*, pp. 173–187, San Jose, Calif, USA, February 1993.
- [12] A. Pentland, R.W. Picard, and S. Scarloff, "Photobook: tools for content based manipulation of image databases," in *Proceedings of the International Society for Optics and Photonics (SPIE 2185)*, pp. 34–47, San Jose, Calif, USA, 1994.
- [13] S. Mehrotra, Y. Rui, M. Ortega-Binderberger, and T. S. Huang, "Supporting content-based queries over images in MARS," in *Proceedings of the IEEE International Conference on Multimedia Computing and Systems (ICMCS '97)*, pp. 632–633, June 1997.
- [14] J. R. Bach, C. Fuller, A. Gupta et al., "Virage image search engine: an open framework for image management," in *Proceedings of the Storage and Retrieval for Still Image and Video Databases (SPIE 2670)*, pp. 76–87, San Jose, Calif, USA, February 1996.
- [15] J. R. Smith, *Integrated spatial and feature image systems: retrieval, analysis and compression*, Ph.D. thesis, Columbia University, New York, NY, USA, 1997.
- [16] J. Z. Wang, J. Li, and G. Wiederhold, "SIMPLiCity: semantics-sensitive integrated matching for picture libraries," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 947–963, 2001.
- [17] Y. Rui, T. S. Huang, and S. F. Chang, "Image retrieval: current techniques, promising directions, and open issues," *Journal of Visual Communication and Image Representation*, vol. 10, no. 1, pp. 39–62, 1999.
- [18] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 3, no. 6, pp. 610–621, 1973.
- [19] Y. D. Chun, S. Y. Seo, and N. C. Kim, "Image retrieval using BDIP and BVLC moments," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 9, pp. 951–957, 2003.