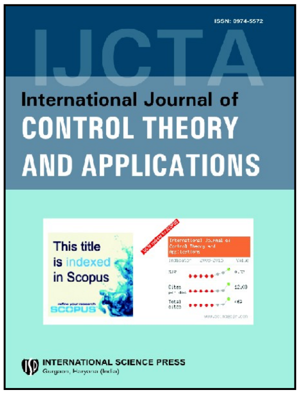

International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 29 • 2017

Enhanced Bat with Kernel Clustering (EBKC) and Multi-feature Analysis for Content Based Image Retrieval System

V. Elizabeth Jesi[1] and S. Govindarajan[2]

[1] *Asst- Prof, Dept of Information Technology, SRM University, Chenai*
[2] *Prof, Dept of Computer Applications, SRM University, Chenai*
E-mail: ve_jesi@yahoo.co.ingovindaraja.n@ktr.srmuniv.ac.in

Abstract: Content Based Image Retrieval (CBIR) is regarded as one of the most effective ways of accessing visual data. It deals with the image content itself such as color, shape and image structure instead of annotated text. Huge amounts of data retrieval challenge the traditional database technology, but the traditional text-object database cannot satisfy the requirements of an image database. CBIR many visual features like color, shape, and texture are extracted in order to match query image with stored database images. Matching the query image with each image of large scale database results in large number of disc scans which in turns slows down the systems performance. So efficient indexing technology is content based image retrieval can play advantage in large image databases. Multi-feature large-scale image retrieval algorithm based on semantic parsing and modified Enhanced Bat with Kernel Clustering (EBKC) algorithm for CBIR. First, Hue Saturation Value (HSV), Hue Saturation Lightness (HSL) color space is quantified rationally. Color histogram and texture features based on a co-occurrence matrix are extracted to form feature vectors. Then the characteristics of the global color histogram, local color histogram and texture features are analyzed for CBIR. Multidimensional feature indexing technology is used to express the visual features of image content has its own representation method; it greatly increases the difficulty of the index. The proposed work suggested an EBKC algorithm, in which the database images are clustered into optimized clusters for further retrieval process. From the features of semantic description from the angle of the general convenient user query expression needs to consider, the image semantic classification is given the retrieval speed and accuracy.

Index terms: Content Based Image Retrieval (CBIR), Feature Extraction, clustering, feature analysis, kernel clustering, Enhanced Bat with Kernel Clustering (EBKC)

1. INTRODUCTION

With the development of digital image processing technology, it has become imperative to find a method to efficiently search and browse images from large image collections. The explosive growth of digital libraries due to Web cameras, digital cameras, and mobile phones equipped with such devices is making the database management by human annotation an extremely tedious and clumsy task. Thus there exists a dire need for developing an efficient expert technique that can automatically search the desired image from the huge database.

Content-based image retrieval (CBIR) [1] is one of the commonly adopted solutions for such applications. The feature extraction in CBIR is a prominent step whose effectiveness depends upon the method adopted for extracting features from given images. The CBIR utilizes visual contents of an image such as color, texture, shape, faces, spatial layout, etc. [2], to represent and index the image database. These features can be further classified as general features such as color, texture, and shape, and domain-specific features such as human faces, fingerprints, etc [3]. The difficulty to find a single best representation of an image for all perceptual subjectivity is due to the fact that the user may take photographs in different conditions such as view angle, illumination changes, etc.

The most commonly used low-level features include those reflecting color, texture, shape, and salient points in an image [4]. Because of the robustness, effectiveness, implementation simplicity and low storage requirements advantages, color has been the most effective feature and almost all CBIR systems employ colors. HSL, HSV or CIE Lab and LUV spaces are used to represent color instead of the RGB space as they are much better with respect to human perception [5]. Generally, the distribution of color was represented by color histograms and formed the images' feature vectors. Shape [6], texture [7] and spatial features [8] etc. were adopted to improve the color based CBIR as different images may have similar or identical color histograms and images taken under different ambient lighting may produce different histograms. The texture feature is another wildly used feature in CBIR, which intended to capture the granularity and repetitive patterns of surfaces within an image [4]. In the MPEG-7 standard, a set of color and texture descriptors including histogram-based descriptors, spatial color descriptors and texture descriptors were defined to interpret natural images [9]. Image clustering [10] and categorization is a means for high-level description of image content. The goal is to find a mapping of the archive images into classes (clusters) such that the set of classes provide essentially the same information about the image archive as the entire image-set collection.

In this paper, a multi-feature large-scale image retrieval algorithm based on the semantic parsing and Enhanced Bat with Kernel clustering (EBKC) is employed for image retrieval. Initially the HSV and HSL color space is quantified rationally. The multi-dimensional feature indexing technology helps in improving the indexing algorithm to adapt large scale database for representing the features of the image content. However as the difficulty increases in indexing, the proposed method utilizes EBKC for optimized clustering, thus enhancing the image retrieval performance. The remainder of this article is organized as: Section 2 describes the related research works briefly. Section 3 explains the proposed methodology. Section 4 presents the evaluation results of the proposed methodology. Section 5 makes a conclusion of this research article.

2. RELATED WORKS

Lin *et al.* [11] proposed a color-texture and color-histogram based image retrieval system (CTCHIR). They proposed three image features, based on color, texture and color distribution, as color co-occurrence matrix (CCM), difference between pixels of scan pattern (DBPSP) and color histogram for K-mean (CHKM) respectively and a method for image retrieval by integrating CCM, DBPSP and CHKM to enhance image detection rate and simplify computation of image retrieval. Hiremath & Pujari [12] proposed CBIR system based on the color, texture and shape features by partitioning the image into tiles. The features computed on tiles serve as local descriptors of color and texture features. The color and texture analysis are analyzed by using two level grid frameworks and the shape feature is used by using Gradient Vector Flow. Rao et al. [13] proposed CTDCIRS (color-texture and dominant color based image retrieval system), they integrated three features like Motif co-occurrence matrix (MCM) and difference between pixels of scan pattern (DBPSP) which describes the texture features and dynamic dominant color (DDC) to extract color feature.

Shyu *et al.* [14] proposed a Markov model mediator (MMM) to facilitate the capturing of high level image concepts in the CBIR. It learns the high level concepts of the images from the history of the user access patterns, and the access frequencies of the images in the database. The affinity relationship among the images in the database is contained in the relative affinity matrix A. The user selects one query image. The query message is sent to the server. The server retrieves the related images from the database and sends the query results to the client. Upon receiving the results, user selects the relevant images. This feedback is sent back to the server. When the server receives this feedback message, it updates the user access patterns and access frequencies accordingly.

Liu *et al.* [15] proposed a semantic clustering scheme for region based image retrieval. The semantic similarities between the image regions are obtained with the help of the user query and feedback history. The affinity matrix is used to store the users'feedback about the image. Initially, the query region is entered as a new semantic cluster and its feature vector is treated as the centroid of the semantic cluster. For every image that is labeled as positive, the system finds out which region of the positive image has the shortest Euclidean distance from the query region, and puts this positive region in the same cluster of the query region. All the other regions in that image have the label unknown. All the regions in the images marked as negative are marked as negative in the corresponding query region.

Nugroho *et al.,* [16] proposed a rotation invariant indexing for images, using Zernike moments and the R–tree. The Zernike moments has a rotation invariant characteristic. The R-tree algorithm stores and searches the magnitude of the Zernike moments.In [17], an efficient image retrieval technique which uses dominant color and texture features of an image has been proposed. This methodyielded higher average precision and average recall with reduced feature vector dimension.

A new and effective color image retrieval scheme for combining all the three i.e. color, texture and shape information, which achieved higher retrieval efficiency, has been presented in [18]. By using fast color quantization algorithm with clusters merging, the image is predetermined, and then a small number of dominant colors and their percentages can be obtained. Then using a steerable filter decomposition which offers an efficient and flexible approximation of early processing in the human visual system, the spatial texture features are extracted. After that the pseudo-Zernike moments of an image are used for shape descriptor, which have better features representation capabilities and are more robust to noise than other moment representations. Finally, the combination of the color, texture and shape features provided a robust feature set for image retrieval.

Trademark image retrieval (TIR) system has been proposed in [19] to deal with the vast number of trademark images in the trademark registration system. The proposed approach commences with the extraction of edges using the canny edge detector, performs a shape normalization procedure, and then extracts the global and local features. The local features describe the interior details of the trademarks, while the global features capture the gross essence of the shapes. To measure the similarity between the query and database images, a two-component feature matching strategy is used.

A content-based image retrieval method based on an efficient combination of multi resolution color and texture features has been proposed in [20]. Color auto correlograms of the hue and saturation component images in HSV color space are used as its color features. BDIP (Block Difference of Inverse Probabilities) and BVLC (Block Variation of Local Correlation coefficients) moments of the value component image are adopted as its texture features. In multi-resolution wavelet domain, the color and texture features are extracted and combined. At a point where the retrieval accuracy becomes saturated, the dimension of the combined feature vector is determined.

In [21] two image indexing approach were proposed for image retrieval system namely, Local binary pattern and Gabor transform feature. In terms of performance evaluation measures this technique achieved an efficient result, when made in comparison with other existing methods. Anandh et al, [22] proposed a technique for the generation of image content descriptor with three features viz., Color auto-Correlogram, Gabor Wavelet and Wavelet Transform. Color Auto-Correlogram Feature is associated with color information of an image which is derived from the RGB color space of an image. The Gabor Wavelet Feature has the texture information to extract the textural features associated with the image and the Wavelet Transform Feature is linked with shape information in the extraction of edges in an image. The feature extraction process has been accomplished based on the input query image from the IDB and the features are stored in a feature dataset. The Manhattan distance has been applied on the user given query image and feature vector computed from database images for measuring similarity. Finally, the presented technique retrieves the meaningful image from the image database which satisfies the user expectation.

Similarly many researchers focused on content based image retrieval; however there are certain drawbacks that are less recognized and caused degradation in the overall retrieval performance. Hence this research work aims to develop a multi-feature analysis method and EBKC for enhanced CBIR.

3. PROPOSED METHODOLOGY

In the proposed methodology, initially the Hue Saturation Value (HSV), Hue Saturation Lightness (HSL) color space are quantified rationally followed by the extraction of color histogram and texture features. The extracted features forms the feature vectors while the characteristics of the global and local color histogram and texture features are analyzed for CBIR. Multidimensional feature indexing technology is used to express the visual features of image content and the proposed EBKC provides optimal clusters of image content for efficient image retrieval with higher accuracy.

HSV Color space

The HSV color space attempts to characterize colors according to their hue, saturation, and value (brightness).

Hue- The hue of a color identifies what is commonly called “color”. For example, all reds have a similar hue value whether they are light, dark, intense, or pastel [23]. Ranges from 0-360 (but normalized to 0-100% in some applications).

Saturation-The saturation of a color identifies how pure or intense the color is. A fully saturated color is deep and brilliant as the saturation decreases, the color gets paler and more washed out until it eventually fades to neutral[23]. Ranges from 0-100%.Also sometimes called the “purity” by analogy to the colorimetric quantities excitation purity andcolorimetric purity. The lower the saturation of a color, the more “grayness” is present and the more faded the color will appear, thus useful to define de-saturations the qualitative inverse of saturation

Brightness-The brightness of a color identifies how light or dark the color is. Any color whose brightness is zero is black, regardless of its hue or saturation. There are different schemes for specifying a color’s brightness and depending on which one is used; the results of lightening a color can vary considerably. Ranges from 0-100%

This color space is based on a so-called hex-cone model which can be visualized as a prism with a hexagon on one end that tapers down to a single point at the other. The hexagonal face of the prism is derived by looking at the RGB cube centered on its white corner. The cube, when viewed from this angle, looks like a hexagon with white in the center and the primary and secondary colors making up the six vertices of the hexagon. This color hexagon is the one Picture Window uses in its color picker to display the brightest possible versions of all possible colors based on their hue and saturation. Successive cross sections of the HSV hex-cone as it narrows to its vertex are illustrated below showing how the colors get darker and darker, eventually reaching black [23] [24]. The algorithm to convert images from RGB to HSV color space is given as follows:

Algorithm 1: RGB to HSV

1. V = Maximum(R,G, B)
2. Delta = V - Minimum(R, G, B)

3. S = choose(V=0, 0, Delta * 255 / V)
4. if S = 0
5. H = 0
6. Else if Red = V
7. H0 = (Green - Blue) / Delta
8. else if Green = V
9. H0 = 2 + (Blue-Red) / Delta
10. Else
11. H0 = 4 + (Red-Green) / Delta
12. End
13. If H0 < 0
14. H0 += 6
15. End
16. H = H0 * 255/6
17. End

HSL Color space

The HSL color space, also called HLS or HIS or HSB. HSI stands for Hue, Saturation, Lightness (also Luminance or Luminosity) / Intensity/ Brightness.

Luminance- The luminance of a color is a measure of its perceived brightness. The computation of luminance takes into account the fact that the human eye is far more sensitive to certain colors (like yellow-green) than to others (like blue) [23].

This color space is based on a double hexagon model which consists of a hexagon in the middle that converges down to a point at each end. Like the HSV color space, the HSL space goes to black at one end, but unlike HSV, it tends toward white at the opposite end. The most saturated colors appear in the middle. Note that unlike in the HSL color space; this central cross-section has 50% gray in the center and not white [23] [25]. The algorithm to convert images from RGB to HSL color space is given as follows:

Algorithm 2: RGB to HSL

1. Max = Maximum (R, G, B)
2. Min = Minimum (R, G, B)
3. if Max = Min
4. H = 0
5. S = 0
6. Else
7. A1 = Max/255 + Min/255
8. S1 = Max/255 - Min/255
9. L = (Max + Min) / 2
10. if L < 128
11. S = (S1 / A1) * 255

```
Else
S = (S1 / (2-S1) )*255
End
Case Max of R
H = ((G - B)/255 / S1 )*60
Case Max of G
H = ( (2 + (B - R)/255)/ S1 )*60
Case Max of B
H = ( (4 + (R - G)/255)/S1)*60
End
if H < 0
H += 255
End
End
```

Colour histogram extraction

A color histogram is the most used method to extract color features. A color histogram is a frequency statistic for different colors in a certain color space. The advantage is that it describes the global color distribution for images. It is especially suited for those images difficult to segment and neglect spatial locations. However, its drawback is that it cannot describe the local distribution of the image in color space and the spatial position of each color. It means that the color histogram cannot describe specific objects or things in the image. The color space needs to be divided into several small ranges in order to calculate the color histogram. Each interval is regarded as a bin. Thus, the color is quantized. The color histogram can be calculated through counting pixels where the colors fall into each interval. Color features include global color histogram and local color histogram [26].

Global color histogram based CBIR

For an image selected from an image database, because an RGB color space does not meet the visual requirements of people, for image retrieval, the image normally converts from an RGB space to other color spaces. The HSV and HSL space are used in this approach as it is a more common color space. The global color histogram can be calculated as follows:

Step 1: Convert the images from RGB space to HSV space or HSL space.

Step 2: Quantify the images using the formulas given below.

$$H = \begin{cases} 0 & h \in [316, 360] \\ 1 & h \in [1,25] \\ 2 & h \in [26, 40] \\ 3 & h \in [41, 120] \\ 4 & h \in [121, 190] \\ 5 & h \in [191, 270] \\ 6 & h \in [271, 295] \\ 7 & h \in [295, 315] \end{cases} \tag{1}$$

$$S = \begin{cases} 0 & s \in [0, 0.2] \\ 1 & s \in [0.2, 0.7] \\ 2 & s \in [0.7, 1] \end{cases} \tag{2}$$

$$V = \begin{cases} 0 & v \in [0, 0.2] \\ 1 & v \in [0.2, 0.7] \\ 2 & v \in [0.7, 1] \end{cases} \tag{3}$$

Step 3: Count each feature value.

Step 4: Calculate similarity by Euclidean distance

$$D = \sum_{i=1}^{n} (A_i - B_i)^2 \tag{4}$$

Here, A and B are two feature vectors; n is the dimension of feature vectors.

The global color histogram is a simple way of extracting image features. High effective calculation and matching are its main advantage. The feature is invariable to rotation and translation. The drawback is that the global color histogram only calculates the frequency of colors. The spatial distribution of color information is lost. Two completely different images can get the same global color histogram, which will cause retrieval errors. Thus the color features can be extracted without errors.

Local color histogram based CBIR

For the local color histogram, an image is separated into n×n blocks. Each block will have less meaning if the block is too large, while computation of retrieval process will be increased if the block is too small. Through the comparative analysis, a two-dimensional space divided into 3×3 will be more effective. For each block, a calculation of the color space converting and color quantization is carried out. The normalized colorfeatures for each block can be calculated.Usually in order to highlight the specific weight of different blocks, a weight coefficient is distributed to each block. Andthe weight of middle block is often larger.

Extracting texture features

The texture features are extracted in 5 steps in this approach.

Step 1: Image color conversion.

The color image will be converted to a grey-scale image using the below formula [27], the number of the grey-scale is 256.

$$Y = 0.29 \times R + 0.587 \times G + 0.114 \times B \tag{5}$$

where Y is the grey-scale value. R, G, B represent red, green and blue component values respectively.

Step 2: Grey-scale quantification.

Because the grey-scale is 256, the corresponding co-occurrence matrix is 256×256. According to the human visual feature, the similarity of most images is mainly distinguished by the relative coarse texture features. The grey-scale of the initial image will be compressed to reduce calculations before the co-occurrence matrix is formed. 16 compression levels were chosen in the paper to improve the texture feature extracting speed.

Step 3: Feature value calculation.

Four co-occurrence matrices are formed according formula 3 to formula 6 in four directions. The four texture parameters: capacity, entropy, moment of inertia and relevance are calculated. Finally, the means and standard deviations of each parameter are taken as each component of the texture features.

Step 4: Internal normalization.

For an image and its corresponding feature vector $H^i = [h^{i,1}, h^{i,2}, \dots, h^{i,N}]$, assume the feature component valuesatisfies a Gaussian distribution. The Gaussian normalization approach is used to implement internal normalization in orderto make each feature of the same weight.

$$h^{i,j\prime} = \frac{h^{i,j} - m_j}{\sigma_j} \tag{6}$$

where, m_j is the mean and σ_j is the standard deviation. $h^{i,j}$will be unitized on range [–1,1].

Step 5: Texture feature comparison.

The texture feature of each image is calculated according to the above steps. The texture values are compared by Euclidean distance, the closer the distance the higher the similarity.

Multi-dimensional Feature indexing

In the CBIR process, the quality of the image features directly affects the efficiency of retrieval. As the image library is large, the features are also large and the right to access and index classification are very important.However the image as an intuitive and simple method of stored information, become an important part of the database, but its capacity is big, so the efficiency of image retrieval becomes very low. Establishing an effective index structure in the image database can enhance the efficiency of image retrieval. The multi-dimensional features are most commonly extracted in the multimedia image retrieval. Organization arrangement of texture feature contains object structure and links with the surrounding environment, therefore widely used in multimedia image retrieval. The color and texture features are extracted as illustrated in the previous sub-sections. The feature vectors of the images increased with the number of features and form the dimension disaster problem. This high dimension increases the complexity of the calculationat the same time reduce the classification performance in the retrieval process. As to tackle this situation, the dimensionality reduction has to be applied. The idea of dimension reduction is achieved by function mapping from high dimension space to low dimension space using the following equation

$$F(x) = \begin{Bmatrix} F_1(x) \\ F_2(x) \\ \vdots \\ F_n(x) \end{Bmatrix} = \begin{Bmatrix} F_1(x_1, x_2, x_3, \dots x_m) \\ F_2(x_1, x_2, x_3, \dots x_m) \\ \vdots \\ F_n(x_1, x_2, x_3, \dots x_m) \end{Bmatrix} \tag{7}$$

This algorithm firstly will make certain assumptions in embedded map or low dimensional manifold aspects, or be able to maintain a certain property of high-dimensional data remains the same: and then through the internal data structure of the nonlinear manifold, map the data dimension reduction problem is transformed into solving Eigen value problem and does not require iteration method, finally a low-dimensional subspace of non-linear is obtained. The advantage of such a nonlinear algorithm is in most data can reflect the nature of the data. After the dimensionality reduction, the semantic parsing and clustering is performed.

Semantic Parsing and EBKC based algorithm

Semantic parsing is the process to derive a structural logistic representation of the image vectors. It uses a structured regression model with features defined over utterance–logical form pairs. In this proposed approach, the CBIR must define a distance measure to calculate the similarity between two images relative distances between the query image and the target image, thus finding the most similar images with the query image. Description based on the histogram of image content, the simplest method is direct histogram similarity matching. The traditional histogram method due to the information does not include the image pixel space, affecting the precision of the retrieval and correctness. Because traditional color histogram can only record the overall image, so with the same histogram of the image may be very different from the semantics. The texture areas of high frequency and low frequency sometimes have the identical color histogram. This problem creates discrimination problem in larger databases and does not match with the human visual system. Usually in the compressed format image database storage, it is difficult to directly extract content characteristics and the decoding characteristics after extraction efficiency is very low, because the decoding will spend extra time. Hence the semantic parsing and Enhanced Bat based kernel clustering is highly efficient.

The Enhanced Bat based kernel clustering is employed for clustering the image feature vectors based on similarity. The kernel clustering is performed in which the optimal clusters are obtained using the Enhanced Bat algorithm. First the training feature vectors, label matrix, hash functions, the objective function, bat population, pulse frequency, pulse rate and loudness are initialized. Then for each training vector the kernels are assigned and the kernel updating is analyzed. The Eigen value problem is resolved by obtaining the largest Eigen vector and computing the similarity values. Based on these values the clustering is achieved. However inorder to achieve optimal clustering, the Enhanced Bat is utilized. Each bat is assigned with the function of finding the new solution and finally the best solution is selected from each bat. Thus the optimal kernel clustering is achieved which aids in the enhanced retrieval of images.

Algorithm 3: Semantic Parsing & EBKC

Input: Training feature vectors $F = \{f_i \in \mathbb{R}^d\}_{i=1}^n$,, label matrix $S \in \mathbb{R}^{l \times l}$, old hash functions h(f) if updating

Initialize $R_0 = r_S$ and $T_{max} = 500$:

Objective function $f(x)$, $x = (x_i, \ldots x_d)^T$

Initialize bat population x_i ($i = 1, 2, \ldots$ n) and v_i

Define pulse frequency pf_i at

Initialize pulse rate and the loudness

For k = 1; $k \leq r$ do

If is updating then

$$a_k^0 \leftarrow a_k^*$$

End

Else

Solve the generalized Eigen value problem

$\bar{K}_l^T R_{k-1} \bar{K}_l a = \lambda \bar{K}_l^T \bar{K}_l$ obtaining the largest Eigen

vector a_k^0 such that $(a_k^0)^T \bar{K}_l^T \bar{K}_l a_k^0 = l$;;

End

If $(h^0)^T R_{k-1} h^0 > (h^*)^T R_{k-1} h^*$ then

$$a_k^* \leftarrow a_k^0 h^* \leftarrow h^0$$

End
Use Enhanced Bat to optimize
While (t< Max number of iterations)
Generate new solutions by adjusting frequency, and updating velocities and locations/solutions
If (ran d > r_i)
Select a solution among the best solutions
Generate a local solution around the selected best solution
End if
Generate a new solution by flying randomly
If (rand < A_i & $f(x_i) < f(x_*)$)
Accept the new solutions
Increase and reduce
End if
Rank the bats and find the current best
End while
Post-process results and visualization
Output: Optimal hash function $h^* (fx)$

4. PERFORMANCE EVALUATION

The experiments are conducted in MATLAB for the performance evaluation of the proposed methodology. The image library is utilized in this evaluation is from the Corel image library and atlas websites. A total of 4000 images are utilized that are divided into 40 classes, each class containing around 20 to 130 images. The quantization and indexing are employed significantly to minimize the number of similar images in each of the clusters. The

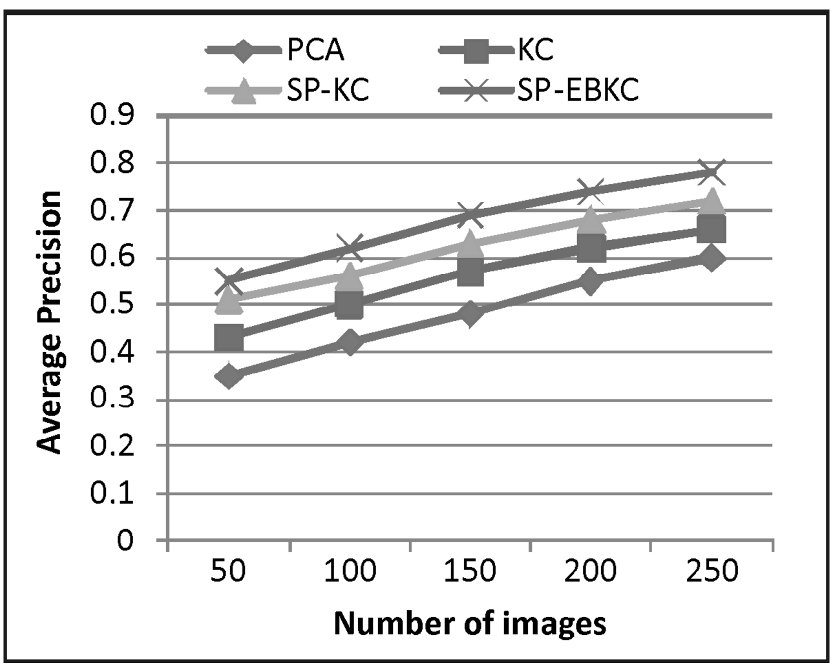

Figure 1: Comparison of Average Precision

performance of the proposed image retrieval system based on Semantic parsing and EBKC algorithm (referred as SP-EBKC in this section) is compared with that of the existing schemes namely Principal component analysis based retrieval (PCA), Kernel clustering based retrieval (KC) and Semantic parsing and Kernel clustering based retrieval (SP-KC). The performance metrics namely average values of precision, recall, accuracy and retrieval time are employed for evaluation.

Figure 1 shows the comparison of the image retrieval methods in terms of average precision. It can be seen that even with the increase in number of images, the precision values of the proposed SP-EBKC outperforms the existing retrieval methods. For instance, when the number of images is 250, the average precision in SP-EBKC is 0.78 which is 6%, 12% and 18% higher than SP-KC, KC and PCA based retrieval methods respectively.

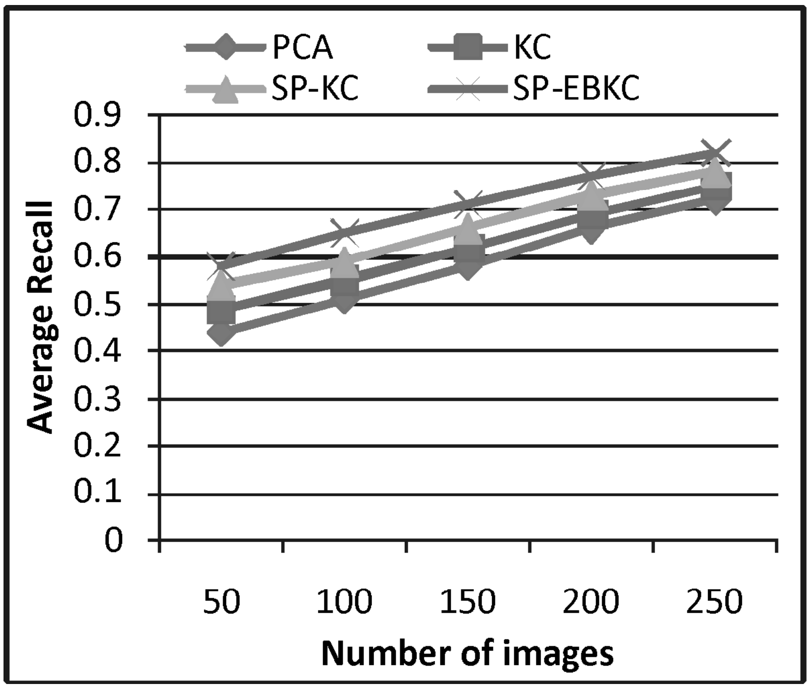

Figure 2: Comparison of Average Recall

Figure 2 shows the comparison of the image retrieval methods in terms of average recall. It can be seen that, the recall values of the proposed SP-EBKC outperforms the existing retrieval methods. For instance, when the number of images is 250, the average recall in SP-EBKC is 0.82 which is 4%, 7% and 10% higher than SP-KC, KC and PCA based retrieval methods respectively.

Figure 3 shows the comparison of the image retrieval methods in terms of average accuracy. It is found that the proposed SP-EBKC outperforms the existing retrieval methods in terms of accuracy. When the number of images is 250, the average accuracy in SP-EBKC is 93% which is 4%, 7% and 11% higher than SP-KC, KC and PCA based retrieval methods respectively. Thus it is proved that the proposed method provides highly accurate content based image retrieval.

Figure 4 shows the comparison of the image retrieval methods in terms of retrieval time. From the results it can be seen that the proposed SP-EBKC has less retrieval than the existing retrieval methods. When the number of images is 250, the retrieval time in SP-EBKC is 0.41 seconds which is 0.06 seconds, 0.1 seconds and 0.17 seconds less than SP-KC, KC and PCA based retrieval methods respectively. Thus it is proved that the proposed method retrieves images quickly than the other methods.

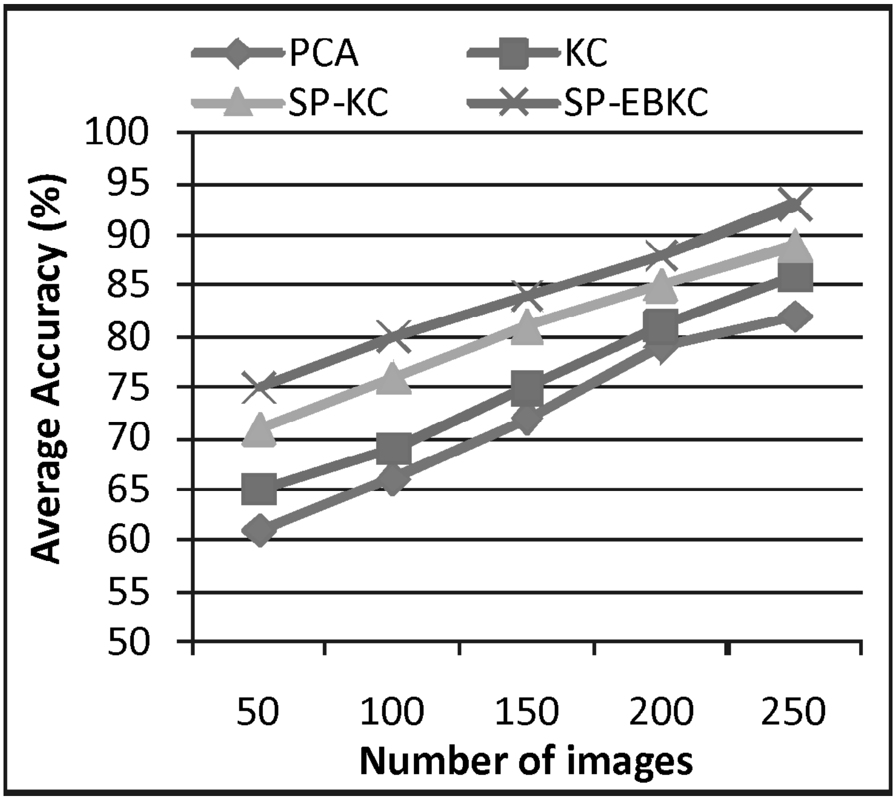

Figure 3: Comparison of Average Accuracy

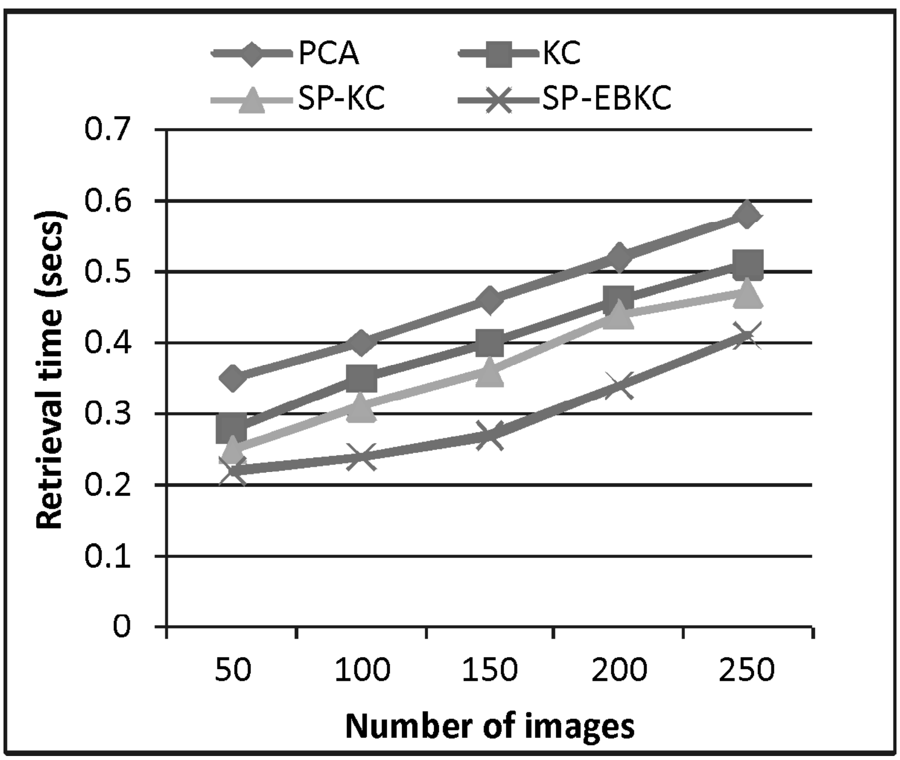

Figure 4: Comparison of Retrieval time

5. CONCLUSION

This paper developed an enhanced multi-feature analysis algorithm for the efficient retrieval of the images based on their content. The proposed algorithm utilizes Semantic Parsing and Enhanced Bat based Kernel clustering. This approach initially defines the images in HSV and HSL color spaces and then extracts the color features and texture features. Then the global and local color histogram and texture features are analyzed for the CBIR and then they are converted into feature vectors. This is followed by the dimensionality reduction and the optimized clustering of features. Thus the efficiency of the image retrieval is enhanced significantly and the evaluation results prove the same, indicating the performance efficiency of the proposed method.

REFERENCES

[1] Smeulders, A. W., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. IEEE Transactions on pattern analysis and machine intelligence, 22(12), 1349-1380.

[2] Kokare, M., Chatterji, B. N., &Biswas, P. K. (2002). A survey on current content based image retrieval methods. IETE Journal of Research, 48(3-4), 261-271.

[3] Liu, Y., Zhang, D., Lu, G., & Ma, W. Y. (2007). A survey of content-based image retrieval with high-level semantics. Pattern recognition, 40(1), 262-282.

[4] Datta, R., Joshi, D., Li, J., & Wang, J. Z. (2008). Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys (CSUR), 40(2), 5.

[5] Müller, H., Michoux, N., Bandon, D., &Geissbuhler, A. (2004). A review of content-based image retrieval systems in medical applications—clinical benefits and future directions. International journal of medical informatics, 73(1), 1-23.

[6] Gagaudakis, G., & Rosin, P. L. (2002). Incorporating shape into histograms for CBIR. Pattern Recognition, 35(1), 81-91.

[7] Hiremath, P. S., &Pujari, J. (2007, December). Content based image retrieval using color, texture and shape features. In Advanced Computing and Communications, 2007. ADCOM 2007. International Conference on (pp. 780-784). IEEE.

[8] Chan, Y. K., & Chen, C. Y. (2004). Image retrieval system based on color-complexity and color-spatial features. Journal of Systems and Software, 71(1), 65-70.

[9] Manjunath, B. S., Ohm, J. R., Vasudevan, V. V., & Yamada, A. (2001). Color and texture descriptors. IEEE Transactions on circuits and systems for video technology, 11(6), 703-715.

[10] Lalitha, M., Kiruthiga, M., &Loganathan, C. (2013). A survey on image segmentation through clustering algorithm. International Journal of Science and Research, 2(2), 348-358.

[11] Lin, C. H., Chen, R. T., & Chan, Y. K. (2009). A smart content-based image retrieval system based on color and texture feature. Image and Vision Computing, 27(6), 658-665.

[12] Pujari, J., &Hiremath, P. (2007). Content Based Image Retrieval based on Color, Texture and Shape features using Image and its complement. International Journal of Computer Science and Security, 1(4), 25-28.

[13] Rao, M. B., Rao, B. P., &Govardhan, A. (2011). CTDCIRS: content based image retrieval system based on dominant color and texture features. International Journal of Computer Applications, 18(6), 40-46.

[14] Shyu, M. L., Chen, S. C., Chen, M., Zhang, C., & Sarinnapakorn, K. (2007). Capturing high-level image concepts via affinity relationships in image database retrieval. Multimedia Tools and Applications, 32(1), 73-92.

[15] Liu, Y., Chen, X., Zhang, C., & Sprague, A. (2009). Semantic clustering for region-based image retrieval. Journal of Visual Communication and Image Representation, 20(2), 157-166.

[16] Nugroho, S., &Utomo, D. (2013). Rotation Invariant Indexing For Image Using Zernike Moments and RTree. TELKOMNIKA (Telecommunication Computing Electronics and Control), 9(2), 335-340.

[17] Rao, M. B., Rao, D. B. P., &Govardhan, D. A. (2011). Content based image retrieval using Dominant color and Texture features. International Journal of Computer science and information security, 9, 41-46.

[18] Wang, X. Y., Yu, Y. J., & Yang, H. Y. (2011). An effective image retrieval scheme using color, texture and shape features. Computer Standards & Interfaces, 33(1), 59-68.

[19] Wei, C. H., Li, Y., Chau, W. Y., & Li, C. T. (2009). Trademark image retrieval using synthetic features for describing global shape and interior structure. Pattern Recognition, 42(3), 386-394.

[20] Chun, Y. D., Kim, N. C., & Jang, I. H. (2008). Content-based image retrieval using multiresolution color and texture features. IEEE Transactions on Multimedia, 10(6), 1073-1084.

[21] Murala, S., & Wu, Q. J. (2014). Expert content-based image retrieval system using robust local patterns. Journal of Visual Communication and Image Representation, 25(6), 1324-1334.

[22] Anandh, A., Mala, K., &Suganya, S. (2016). Content based image retrieval system based on semantic information using color, texture and shape features. In Computing Technologies and Intelligent Data Engineering (ICCTIDE), International Conference on (pp. 1-8). IEEE.

[23] Sachs, J. (1996). Digital image basics. Digital Light & Color, 1999.

[24] Cardani, D. (2001). Adventures in hsv space. Laboratorio de Robótica, InstitutoTecnológicoAutónomo de México.

[25] Ford, A., & Roberts, A. (1998). Colour space conversions. Westminster University, London, 1998, 1-31.

[26] Yue, J., Li, Z., Liu, L., & Fu, Z. (2011). Content-based image retrieval using color and texture fused features. Mathematical and Computer Modelling, 54(3), 1121-1127.

[27] Wang, H., & Shi, P. (2006). An approach for feature extraction of texture images. Journal of Communication University of China (Science and Technology), 13(1), 49-52.