ISSN 2278-3091
Volume 7, No.2, March - April 2018
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse02722018.pdf
https://doi.org/10.30534/ijatcse/2018/02722018

Content-Based Image Retrieval Using Hierarchical Color and Texture Similarity Calculation

D. Sarala[1], Tejas Kanikdaley [2], Sharad Jogi [3], Rahul K. Chaurasiya [4]
Dept. of Electronics & Telecommunication, National Institute of Technology Raipur
G.E. Road Raipur, India
[1] sarala.336@gmail.com
[2] tejaskanikdaley1996@gmail.com
[3] sharad.jogi100696@gmail.com
[4] rkchaurasiya.39@gmail.com

ABSTRACT

With the evolution of multimedia technology, the usage of large image database has rapidly increased. Content-based image retrieval has come up as an effective method for image retrieval and management. The use of a single image feature for image retrieval gives unsatisfactory results, therefore the usage of multiple image features is more often preferred. This paper introduces a novel method that combines both color and texture features of an image for image retrieval in a hierarchical manner and shows its advantage. This paper also introduces an effective method of image segmentation for feature extraction. The proposed hierarchical approach was applied to the standard INRIA dataset. A significant improvement in the recall and precision is achieved and is shown in the result section.

Keywords : Content-based image retrieval, feature extraction, color, texture, local binary pattern.

1. INTRODUCTION

With the rapid progression in multimedia related technology, a huge database of images is reserved on the internet. Content-based image retrieval (CBIR) is an approach used for retrieving related images from an image database. Image retrieval techniques can be widely classified in two categories, annotation-based image retrieval (ABIR) [1, 2] and content-based image retrieval (CBIR) [3].

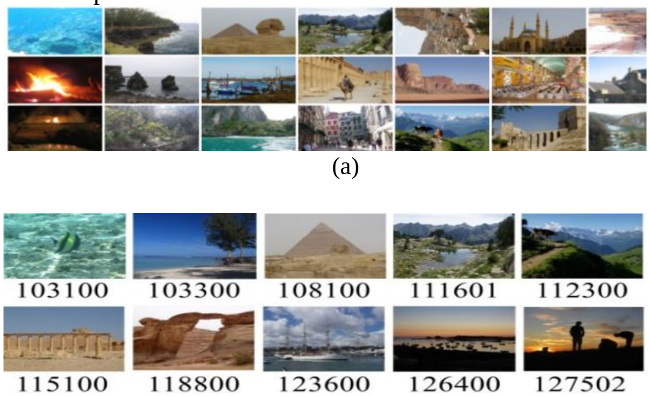
In ABIR, given a textual query and a set of images with their annotations (phrases or keywords), annotation-based image retrieval systems retrieve images according to the matching score of the query and the corresponding annotations. Even though it can provide fair retrieval interpretation, certain limitations still persist. For example, relatively abstract images cannot be described only by using several keywords. CBIR image retrieval is based on the visual composition. Query and target images are represented into vectors constructed by using various types of features like shape, color, texture etc. It then calculates the similarity between those vectors based on the similarity values, an ordered list of images is finally returned as retrieval results [1, 2]. CBIR efficiently solves many concerns related to ABIR. Hence, CBIR has emerged out as a topic of growing interest in recent times. The accuracy of the system comparatively increases when multiple feature vectors are used instead of a single feature vector as the matching procedure now uses more number of features for image comparison [4].This paper concentrates on the methods and strategies for extraction of the image from the dataset using both color and texture features for precise results. The proposed method segments the whole image into several regions and calculates the local histograms for each region in order to achieve localization, giving more accurate results [5, 6].The effectiveness of the system can be significantly improved by restricting the search to the images that are alike to the query image [7, 8]. Hence, the proposed method implements a hierarchical approach by refining results at each stage, thus improving the computational efficiency. Also, the HSV color space is used which mimics how humans perceive color instead of RGB color space, producing better results [9].

2. DATASET

The INRIA (Institut National de Recherche en Informatique et en Automatique) dataset is used in this paper [10]. This dataset mainly contains holiday images. Some irrelevant images were included intentionally in order to examine the robustness of the system against changes like blurring, sharpening, rotation, viewpoint, illumination, etc. The dataset includes a multitude of images in high resolution belonging to various scene types. The dataset includes 802 such images. From these images, 10 images have been randomly selected as query images. It is ensured that a remarkable number of images similar to each of the query image are present in the dataset for comparison. Figure 1 shows sample images from INRIA holiday dataset and figure 2 shows the randomly selected 10 query images.

3. THE PROPOSED CBIR METHOD

A novel CBIR system based on color and texture similarity calculation is introduced in this paper. In the proposed system, first the image is segmented into several regions and the color similarity is used to get all the possible related images. Then texture similarity is applied to all the resultant images and they are ranked according to their similarity. The proposed approach will go through the steps as shown in figure 2(a). Each step will be described in sub-sections 3.1.



(a)

(b)

Figure 1: (a): Sample images from the dataset; (b) Sample query images with image ID.

3.1 Color Feature Extraction

A. Image Descriptor

Any image can be well described by an image descriptor which utilizes some of the features of the image to do so. These features include color, texture, shape etc. The color histogram, which defines the distribution of different intensity levels of color in an image, is used as an image descriptor in this paper. If the content of the images is significantly different, and their histogram is similar, they will still be treated relevant to each other. It is better to convert the given RGB image to HSV image as HSV separates the chroma or the color information from the luma, or the image intensity. In other words, computer treats color in the form of its RGB components, whereas HSV tries to capture the components in the way humans perceive color. Here we deal with the histogram of the color component (hue) so we leave the intensity components alone. The HSV representation is shown in figure 2(b). Hue (H) refers to the original color it resembles. For example, all shades and tints of green have the same hue. Hue is a fraction ranging between 0 and 1 based on the position of the color in the color wheel. Saturation (S) of the color describes the degree of whiteness of the color. For example, saturation for white is 0 and for pure red, it is 1. The value (V) of the color describes the amount of darkness of the color. Value 0 represents black and 1 represents white. In order to normalize, the RGB values are divided by 255.In order to normalize, the RGB values are divided by 255. R', G' and B' represents the normalized component. Hue, saturation, and value are calculated as per the given formula:

$Dmax = max(R', G', B'), Dmin = min(R', G', B'), \varDelta = Dmax - Dmin$

$$Hue = \begin{cases} 0° & ,\varDelta = 0 \\ 60° \times \frac{G'-B'}{\varDelta} mod6 & ,Dmax = R' \\ 60° \times \frac{B'-R'}{\varDelta} + 2 & ,Dmax = G' \\ 60° \times \frac{R'-G'}{\varDelta} + 4 & ,Dmax = B' \end{cases} \quad (1)$$

$$Saturation = \begin{cases} 0 & ,Dmax = 0 \\ \frac{\varDelta}{Dmax} & ,Dmax \neq 0 \end{cases} \quad (2)$$

$$Value = Dmax \quad (3)$$

The histogram used is a color (hue) histogram in the HSV domain with 256 bins in hue channel. Instead of calculating the histogram for the entire image, the image is divided into different segments and histogram for each segment is calculated in order to simulate locality in a color distribution. The image is divided as shown in figure 2(c).

B. Feature Extraction

Features are extracted by applying the image descriptor to images in the database. The image is quantified and represented using a list of numbers. Representation for each of the 5 image segments is contained in this list of numbers. A histogram with 256 values is used to describe each segment. The dimension of the overall feature vector is 5 x 256 = 1280 for every image. Thus, 1280 features are used to represent a particular image.

C. Similarity Calculation and Searching

The query image is also passed through the image descriptor to extract its feature vector. Similarity calculation is done between the query image and all the images in the database. Here, chi-squared distance is used for measurement of similarity between two images. The chi-squared distance for 2 histograms is given by:

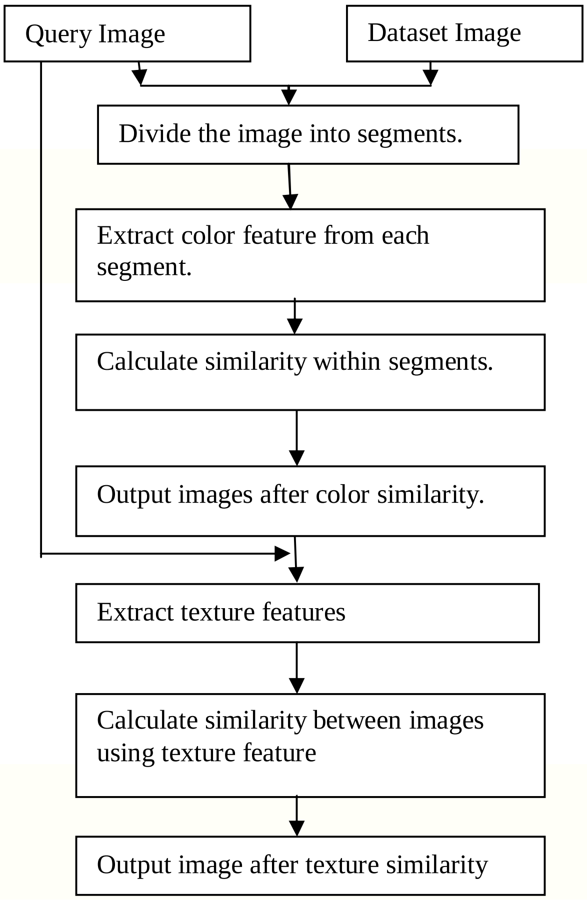
$$D = \frac{1}{2}\sum_{i=0}^{n} \frac{(x_i-y_i)^2}{(x_i+y_i)} \quad (4)$$

Where '*n*' is a number of bins, 'x_i' is a value of first bin, 'y_i' is a value of second bin. If the chi-squared similarity is 0 then the images will be identical. Higher the chi-squared similarity value lesser the similarity between the images. Finally, the 'm' similar images are displayed as the result, where 'm' is the search limit set by the user.

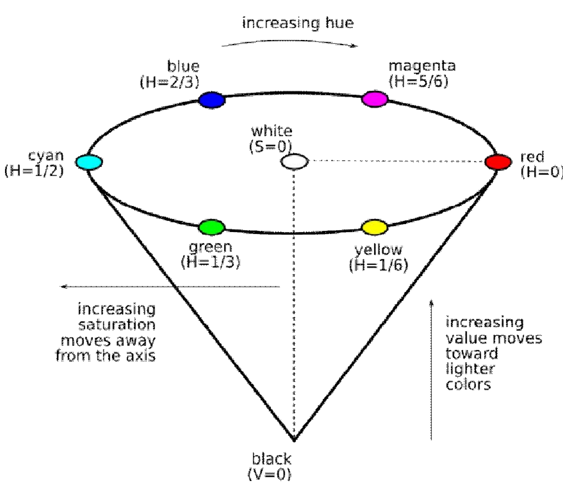
3.2 Texture Feature Extraction

The results obtained after color analysis were used as the training set and the texture analysis was applied to these images to obtain a more filtered output. The texture matching was performed using local binary pattern (LBP) descriptor [10]. Initially, the image is converted from RGB domain to grayscale. The calculation of the local binary pattern (LBP) value takes place as follows:

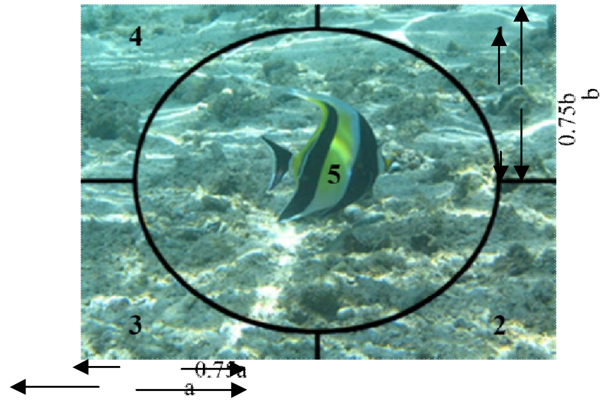
On comparing the pixel's intensity level with the adjoining pixel's intensity level, if the latter is greater or equal, then bit 1 is stored in an array, otherwise, a bit 0 is stored in the array. Thus we get an array of 0's and 1's, which is converted to decimal form. This value is the LBP value as shown in figure 3(a).



(a)

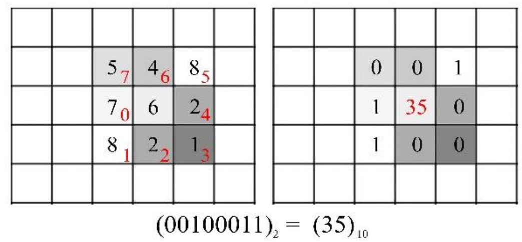


(b)

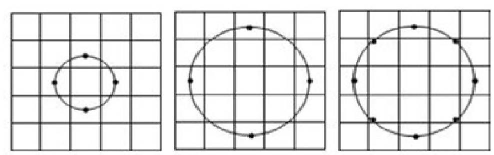


(c)

Figure 2: (a) Flowchart of the proposed method (b) HSV Cone (c) Segmented Image



(a)



(b)

Figure 3: (a) Binary to Decimal Conversion; (b) Uniform LBP

To improvise the results, uniform LBP with a circular neighborhood is used. Steps to obtain LBP mask using uniform LBP are:

Algorithm 1: Texture feature extraction

1. Choose a radius and the number of adjacent points. For experimental purpose, the radius is taken as 3 and number of adjacent points as 24.
2. For each comparison, compare the center pixel with each adjacent point.
3. Make a 1-D array whose size equals the number of adjacent points.
4. If the center pixel has a greater intensity value then store bit 1 in the array, else store bit 0.
5. Convert the binary number stored in the array to decimal form.
6. Create an LBP mask with same dimensions as the original image and update the corresponding pixel value with the decimal value.
7. Calculate and normalize the histogram of the mask.
8. Use the calculated histogram for similarity calculation.

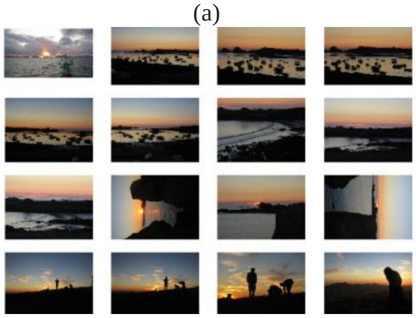
3.3 Calculation of Similarity

In order to evaluate similarity, the obtained histograms are normalized. Then the histogram of the query image and that of all the images in the training set obtained from color filter evaluations are matched. This analysis is done using the chi-squared distance. A score ranging from 0 to 1 is assigned for each comparison, the lowest score having the maximum relevance.

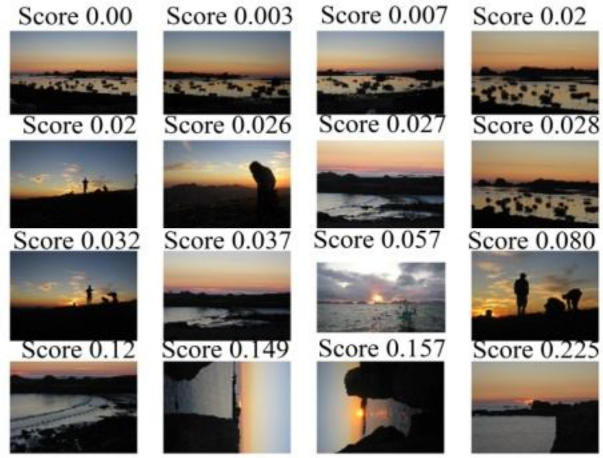
4. EXPERIMENTAL RESULTS AND DISCUSSIONS

The performance of a CBIR can be measured with the help of recall. Recall of a system is its ability to retrieve all the images that are relevant to the query image from the dataset. Suppose that Xr is the total number of relevant images that are retrieved from the database and C represents the total number of relevant images in the database. Then recall, R= Xr/C. Precision is another way to measure the performance of CBIR. The precision of the system is its ability to retrieve a given number of images that are relevant to the query image. Suppose that A is the number of images that are asked by a user to retrieve such that A<= C and Xp represents the number of relevant images that are retrieved. Then precision, P= Xp/A. The results for precision and recall for randomly selected 10 images is shown in table 1.Three different CBIR methods viz. (I) color and texture analysis without segmentation (II) color and texture analysis with segmentation (III) proposed hierarchical color and texture analysis with segmentation were used to obtain the results shown in the table. The average precision and recall are also given in the table. The proposed method has an improvement of 28.5%in precision and 28.3% in recall with method (I) and 17.1% in precision and 18% in recall with method (II).The results presented in the table and figures 4 and 5 validate the effectiveness of the proposed method.

(a)



(b)

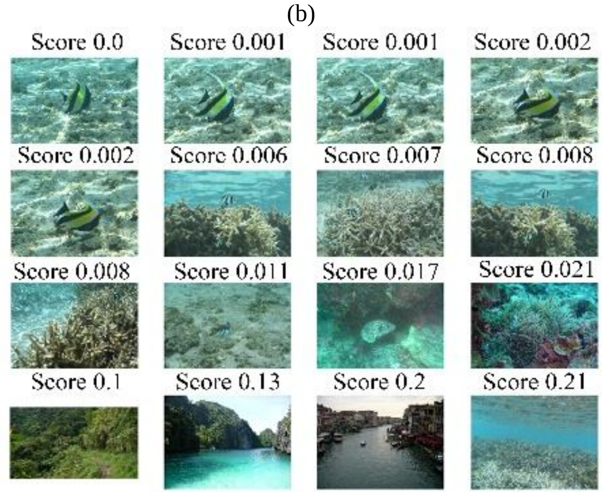


(c)

Figure 4: (a) Query image (b) Training set after color analysis (c) Filtered images after LBP with a corresponding similarity measure

(a)

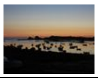
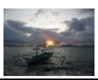
(b)



(c)

Figure 5: (a) Query image (b) Training set after color analysis (c) Filtered images after LBP with a corresponding similarity measure.

Table 1: The Average Recall and Precision of different models

Sno.	**Query Image**	**C**	**A**	**Color and Texture analysis without Segmentation**				**Color and Texture analysis with Segmentation**				**Proposed hierarchical Color and Texture analysis with Segmentation**			
				Xr	***R***	***Xp***	***P***	***Xr***	***R***	***Xp***	***P***	***Xr***	***R***	***Xp***	***P***
1		12	10	9	0.75	8	0.8	8	0.66	8	0.8	11	0.91	10	1
2		25	20	8	0.32	6	0.3	18	0.52	18	0.9	23	0.92	20	1
3		20	15	11	0.55	7	0.46	12	0.6	8	0.53	15	0.75	11	0.73
4		22	15	5	0.22	4	0.26	10	0.45	7	0.46	16	0.72	10	0.66
5		10	7	3	0.3	2	0.28	6	0.6	3	0.42	8	0.8	4	0.57
6		5	3	2	0.4	1	0.33	3	0.6	1	0.33	3	0.6	2	0.6
7		7	5	3	0.43	2	0.4	3	0.42	3	0.6	5	0.71	4	0.71
8		10	8	5	0.5	3	0.37	6	0.6	4	0.5	7	0.71	6	0.75
9		8	5	3	0.37	2	0.4	4	0.5	2	0.4	5	0.62	3	0.6
10		13	10	11	0.84	9	0.9	10	0.76	7	0.7	10	0.77	9	0.9
	Average				0.47		0.45		0.57		0.56		0.75		0.74

5. CONCLUSION

In this paper, a novel content-based image retrieval system based on color and texture similarity is proposed. An attempt has been made to design a computationally efficient system by implementing a hierarchical approach based on color and texture feature. Precision and recall are calculated to measure the performance of the proposed system. Experimental results for 10 class images show that the proposed method is more accurate than those based on color and texture features without and with segmentation respectively. Because the method uses hierarchy, results get refined at each stage making the system computationally efficient. Future work includes the introduction of shape feature for better results. The accuracy can also be improvised by implementing the model using Convolutional Neural Network (CNN).

REFERENCES

1. W. Liu, D. Tao, J. Cheng and Y. Tang. **Multiview Hessian discriminative sparse coding for image annotation**, *Computer Vision and Image Understanding*, vol. 118, pp. 50-60, 2014. https://doi.org/10.1016/j.cviu.2013.03.007
2. C. Jin and S. Ke. **Content-Based Image Retrieval Based on Shape Similarity Calculation", *3D Research*, vol. 8,** no. 3, 2017.
3. S.Pandey and P. Khanna. **Content-based image retrieval embedded with agglomerative clustering built on information loss**, *Computers & Electrical Engineering*, vol. 54, pp. 506-521, 2016. https://doi.org/10.1016/j.compeleceng.2016.04.003
4. X. Wang, J. Wu and H. Yang. **Robust image retrieval based on color histogram of local feature regions**, *Multimedia Tools and*

Applications, vol. 49, no. 2, pp. 323-345, 2009. https://doi.org/10.1007/s11042-009-0362-0

5. N. Shrivastava and V. Tyagi. **Content based image retrieval based on relative locations of multiple regions of interest using selective regions matching**, *Information Sciences*, vol. 259, pp. 212-224, 2014. https://doi.org/10.1016/j.ins.2013.08.043
6. M. E. ElAlami. **A novel image retrieval model based on the most relevant features**, *Knowledge-Based Syst*., vol. 24, no. 1, 2011.
7. S. Youssef. **ICTEDCT-CBIR: Integrating curvelet transform with enhanced dominant colors extraction and texture analysis for efficient content-based image retrieval**, *Computers & Electrical Engineering*, vol. 38, no. 5, pp. 1358-1376, 2012. https://doi.org/10.1016/j.compeleceng.2012.05.010
8. Simardeep Kaur, Vijay Kumar Banga. **Content Based Image Retrieval: Survey and Comparison between RGB and HSV model**, *International Journal of Engineering Trends and Technology (IJETT)*, vol. 4, no. 4, April 2013.
9. Herve Jegou, Matthijs Douze and Cordelia Schmid. **Hamming Embedding and Weak geometry consistency for large scale image search,** P*roceedings of the 10th European conference on Computer vision*, October, 2008 . Retrieved from http://lear.inrialpes.fr/~jegou/data.php#holidays..
10. T. Ojala, M. Pietika¨inen, and T. Ma¨enpa¨a¨,. **Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns,** *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, July 2002. https://doi.org/10.1109/TPAMI.2002.1017623
11. Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma. **A survey of content-based image retrieval with high-level semantics**, *Pattern Recogn*., vol. 40, no. 1, pp. 262–282, Jan. 2007. https://doi.org/10.1016/j.patcog.2006.04.045
12. C. Kavitha, B. Prabhakara, and A. Govardhan CH. **Image Retrieval Based on Color and Texture Features of the Image Sub-Blocks**. *International Journal of Computer Applications (0975 – 8887)*, Vol. 15– No.7, February 2011.
13. E. Hadjidemetriou, M. Grossberg, S. Nayar. **Multiresolution Histograms and Their Use for Texture Classification**, *3rd International Workshop on Texture Analysis and Synthesis*, Oct 2003.
14. N. Yang, W. Chang, C. Kuo and T. Li. **A fast MPEG-7 dominant color extraction with new similarity measure for image retrieval**, *J. Vis. Commun. Image R.*, vol. 19, p. 19, 2008. https://doi.org/10.1016/j.jvcir.2007.05.003
15. J. Ma. **Content-Based Image Retrieval with HSV Color Space and Texture Features**, *2009 International Conference on Web Information Systems and Mining*, 2009. https://doi.org/10.1109/WISM.2009.20
16. H. Atlam, G. Attiya and N. El-Fishawy. **Integration of Color and Texture Features in CBIR System**, *International Journal of Computer Applications (0975 – 8887)*, vol. 164, no. 3, pp. 23-29, 2017.