A Flexible Hierarchical Classification Algorithm for Content Based Image Retrieval

Qiao Liu, Jiangfeng Chen, and Hui Zhang

***Abstract*—The goal of paper is to describe a flexible hierarchical classification algorithm and a new image similarity computing model based on mixing several image features for promoting the performance and efficiency of speed for content-based image retrieval. With an experimental comparison of a large number of different representative point selection approach, we are trying to seek for a method of uniform division of image space, eventually design a novel approach enlightening by high-dimensional indexing and social networking, that introduces the directivity to image classification that is used to explain the convergence of images to edge points of the high-dimension feature space in this paper. Meanwhile we find the laws of parameter setting of this algorithm through experiments and these laws acquires satisfied effects in different dataset. In addition to that algorithm, we also find some features assembling with reasonable formula to represent images better in color, texture and shape. Experimental results based on a database of about 50,000 person images demonstrate improved performance, as compare with other combinations in our descriptor set consisting of several general features mentioned below.**

***Index Terms*—CBIR, hierarchical classification, feature combination.**

I. Introduction

With the continuous development of the search engine technology, people are no longer satisfied with simple text retrieval. Multimedia Retrieval has become an indispensable part of the search field, especially image. Most of the search engines have put text-based image retrieval into their services, even a few outstanding leaders of them just like google have commercialized content-based image retrieval successfully. But although google and other experimental image search projects of the famous university and research institutions have made great progress in this field, either performance or speed has a huge room for improvement. The main reasons for this dilemma involves two aspects: (i) lack of a uncomplicated but accurate computing model and (ii) the high computation complexity. According to this situation, research interest in this field focuses on solving those problems.

In a typical content-based image retrieval system, the visual contents of images in the dataset are converted to multi-dimensional feature vectors. The distance between two feature vectors computing by corresponding method is considered as the similarity of two original images, and we select the Top N minimum distance images as output from calculating results between the feature vectors of the query image and target images in the dataset [1]-[5]. The images that exist in a high-dimensional space formed various sizes clusters based on the distance between them. The approach proposed by this paper narrows greatly the scope of calculation through selecting uniformly representative points using density and directivity of image space in cluster.

The paper is organized as follows. Section 2 describes related work. In Section 3, we elaborate our hierarchical classification algorithm and in which the role of density and directivity of image space. Section 4 shows experimental performance and section 5 contains conclusions.

II. Related Work

In the CBIR (content-based image retrieval) field, most researches focused on descriptor extracting [2], [8], [9] and relationship analysis between images[5], [7]. After decades of development, the descriptors that can be grouped into the (i)color representation, (iii)texture representation[2], and (iii) shape representation[8] contains dozens of different kinds and the models generated by the combination of these characteristics also have shown a trend of diversification. Relationship analysis has been made on the impact of image retrieval on merging interests among different fields of study, such as multimedia (MM), machine learning (ML), information retrieval (IR), computer vision (CV), and human-computer interaction (HCI) [10].

Because of inevitably need to find the optimal results by using the traverse method in the retrieval framework, researchers have attempted to select the subset of image database as small and representative as possible for reducing the amount of computation, involving mainly classification[7] and clustering[5], [11]. But clustering is not effective for query in several high-density classes.

In addition to that, high-dimensional indexing as an effective method to quickly find the target in the high-dimensional space has also been applied to this area, and distance based methods of that, such as SR-tree [4] and VP-tree[12], are suitable for kinds of complex and changeable image descriptor vectors.

A research achievement on social networking interpersonal patterns also brings enlightenment to other similar researches. For example, the phenomenon that people always have bias towards a specific topic to express their opinions could be used for simulating the relationship between images [3].

III. Image Similarity Computing with Combined Features

In this section we give an overview of the features tested

Manuscript received September 4, 2012; revised November 18, 2012.

The authors are with the Department of Computer Science and Engineering, Beijing University of Aeronautics and Astronautics, Beijing, CHN (e-mail: lq@ nlsde.buaa.edu.cn).

and try to make the selection of feature combinations as representative and at the state-of-the-art as possible. The experimental descriptors that come from general features[6] and lire (An Content Based Image Retrieval Library) include following kinds : (I). Scalable color (II). Color layout (III). Edge Histogram (IV). CEDD (V). FCTH (VI). JCD (VII). Tamura (VIII). Gabor (IX). Simple Color Histogram.

Group a Group b

Fig. 1. The group two of test case.

We conducted to find the feature combination from descriptors above through reasonable experiments to satisfy the requirement of computing resource and accuracy. The best answer should consist of two or three kinds of descriptors so that the indexing generated will not be too large. And the combination also should be able to weaken the impact of single feature caused by slight change of image background and shape.

The experiment extract 50,000 person images according to their text labels from 1,170,000 general-purpose images in our image database to make up experimental dataset. Meanwhile we selected two groups of person images from that including :(i). same portrait but different background and (ii) same background but different portrait is shown in Fig. 1. The combination could be regarded as one of candidates if the result of experiments on two group all satisfy that the Top 10 results retrieved from dataset using it include one image while other image is query in the same group. Through building indexings consisting of different kinds of descriptors and analyzing searching result, there exists four kinds of combinations meeting the above requirements:

- Edge Histogram + JCD + Simple Color Histogram
- Edge Histogram + CEDD + Simple Color Histogram
- Edge Histogram + JCD
- Edge Histogram + CEDD

Because of overlapping between different combinations and indexing size (for example, the size of the feature vector of JCD is almost three times more than CEDD but has less improvement), the last one, edge histogram and CEDD in good proportion, is selected as the combination of first step of our computing model and the performance is shown in Fig. 2b while the query is shown in Fig. 2a. Through continuously experiments, the further improvement method is proposed that making use of two other features (FCTH and Color Layout) to sort the results generated by the first step. On the basis of less impacting to better results, a formula that sum up the old distance D_{old} computed by the first step and new distance D_{new} computed by new feature in accordance with a suitable proportion is

$$D_{Last} = \frac{D_{Max} - D_{Old}}{D_{Max}} * D_{Old} + \frac{D_{Old}}{D_{Max}} * D_{New}$$

The Maximum Distance D_{Max} in the above formula means the maximum value that may appear in the first step, new performance is shown in Fig. 2c.

Fig. 2a . The query image.

Fig. 2b . The result of computing after first step.

Fig. 2c . The result of computing after second step.

IV. Hierarchical Classification Algorithm Based on Density and Directivity

According to the distribution situation of images presented in the high-dimensional space, we hope to select the representative images with appropriate quantity and relatively uniform distance by layer-by-layer recursion. The distance of two picture P_i and P_j is denoted by $D(P_i, P_j)$. And the most similar image in dataset H with image Q is called $candidate(Q, H)$ .The specific steps of our algorithm are represented as follows :

- Step 1. Initialize the classification tree T
- Step 2. Select the N least similar images(LSI) P_1, P_2, ..., P_n as L_1 in the whole dataset as first-level child nodes of T, the meaning of least similar images are explained below. And each image P_i selected represents one class C_i.
- Step 3. Each picture Pm of class represented by root (the whole dataset) is classified into different class C_i represented by first-level child nodes Pi of T if $P_i = candidate(P_m, L_1)$
- Step 4. Every child node follow the example of root repeat Step 2 and Step 3 until the number of images of class is less than L_{max} or the level of Node is larger than the maximum level K_{max}
- Step 5. Collect every leaf node of T to make up representative points set S
- Step 6. Each picture is classified into classes represented by each points of S using the same method as above.

The least similar images(LSI) based on the rule of directivity of inter-image relationship in the Step 1 can be

mathematically illustrated as follows. Given a set representative of images selected *S* with distance matrix *W* is

$$\{ D(P_i, P_j) \mid P_i S \ I \ P_j S \}$$

let *S* is

$$\{ P_i \mid " P_j S \ D(P_i, P_j) > R_k \}$$

R_k means the min-distance between LSIs of *K* level of classification tree. Nonetheless three questions still remain: (i) how to select these the least similar images (LSI)? (ii) how to set parameters mentioned above? and (iii) how to retrieve images using these class? For the first question, a iterative approach is introduced as follow steps :

- Step 1. Get any one image *A* from dataset *H*, find the image $P_1 = candidate(A, H)$. .
- Step 2. Get subset *H1* of original dataset *H* containing every images that has distance with P_1 more than R_k
- Step 3. Add P_1 in *S* and Select $P_2 = candidate(P_1, H_1)$
- Step 4. Repeat Step 2 and Step 3 until the size of *S* is larger than *N*

For the second question, the basic parameters mentioned above are described in Table I. The size and density of dataset is mainly considered in these variables setting. For instance, the larger amount of dataset always matches the rise of K_{max} and L_{max}, because more images need to classify more times but every class have better involved more to avoid the reduction of the accuracy. The changing trend in general is concluded in Table II.

TABLE I: THE DESCRIPTION OF PARAMETERS

Parameter	Description
N	The number of LSI in every layer
R_k	The minimum distance between any two LSIs in kth layer
L_{Max}	The minimum size of class allowed to classify
K_{max}	The maximum number of layer

TABLE II: THE CHANGING TREND OF PARAMETERS RELYING ON THE SIZE AND DENSITY OF DATASET

Changes	Size	Density	N	R_k	L_{max}	K_{max}
1	↑	↓	↓	↑	↑	↑
2	↑	↑	↑	↓	↑	↑
3	↓	↓	↓	↑	↓	↓
4	↓	↑	↑	↓	↓	↓

For the third question, we traverse the representative points set *S* at first and obtained the M nearest points, then searching among images belong to the class represented by these points and getting the final results. In addition to content based image retrieval, this kind of classification algorithm could be used for selecting subset in high-dimension space.

V. EXPERIMENT AND RESULTS

Considering the performance of this algorithm in real environment and the characteristic of this algorithm that is not limited by the similarity model, we conducted experiments with images of NIPIC database called dataset *A* in this paper consisting of 1,170,000 general-purposed images and draw 100,000 items from it in random as subset *B* to discover the laws of parameter setting of our classification algorithm. The feature combination mentioned in Section 3 is used as the similarity measure for computing the similarity between the query and target images in the database. Our CBIR system has been deployed on a DELL Inspiron 660 desktop with 4GB memory and Quad-Core 3.1GHz CPU, using the method of reading indexing into memory and multi-threaded parallel computation to speed up its computing.

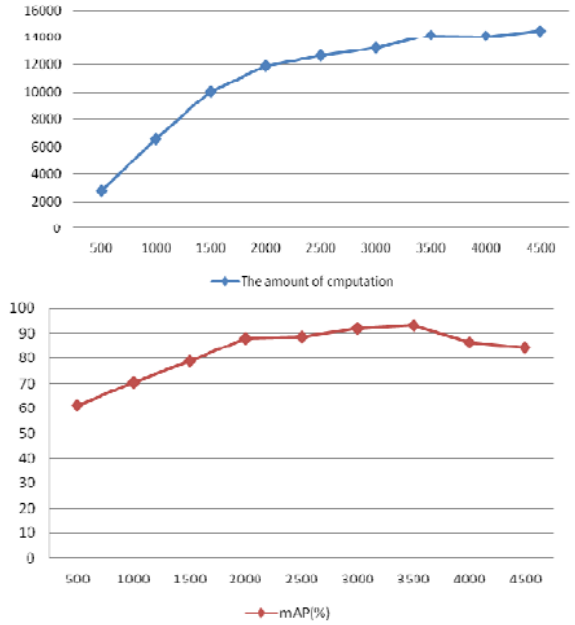

Fig. 3. The amount of computation and precision rate when *N* is 5.

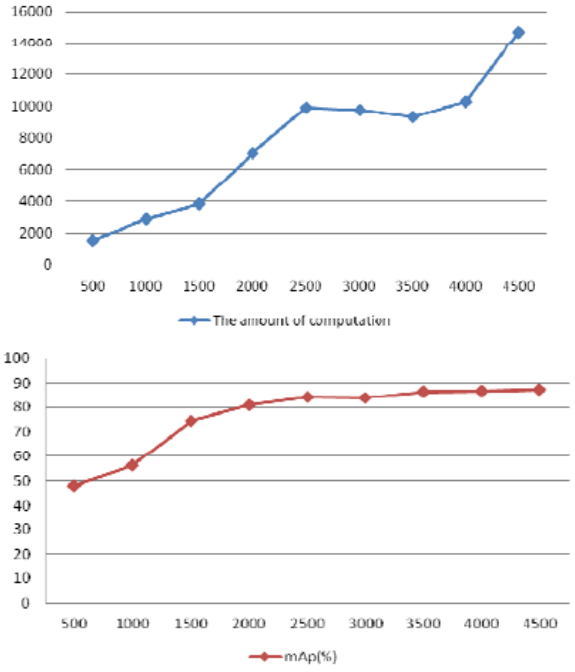

Fig. 4. The amount of computation and precision rate when *N* is 10.

Although parameter setting has strong relationship with specific dataset, we expect to find a regular pattern for setting in detail with experiments and general trend summed up above. R_k is impacted by similarity computation model and

the main influence of K_{max} is the size of dataset. So our experiments focus on the role of N and L_{max} in the classification. We select three representative test cases from 30 query with different complexity of color and shape. The experiments on the image subset B is divided into two groups. N is set to 5 and 10 in two groups and the number of nearest classes selected as computing set is five. Through comparing the amount of computation and precision rate in the different value of L_{max}, we choose ideal point of both shown in Fig. 3 and Fig. 4.

The statistics of experiments show that the first group has better performances but larger computation due to the larger number of images of every class. But the precision rate of the second group is near the first group with less calculation when L_{max} exceeds the threshold of 2000. We surmise that the influence of N will diminish but the rise of amount of computation is not obvious when L_{max} is larger than a threshold value because increasing L_{max} solves the problem of excessive partitioning caused by N. This conclusion has been verified in the dataset A.

Considering the practicality of our system, we hope to decrease the amount of calculation in an acceptable accuracy, so we choose a group of parameters that has more conductive to the former. And the final results of classification of two dataset based on the reasonable parameters setting is shown in Table III.

TABLE III: THE SITUATION OF PARAMETER SETTING: HERE *RK* SUCCESSIVELY INCLUDES THE DISTANCES BETWEEN LSIS IN EVERY LAYER FROM ROOT TO LEAF AND T15 THAT DENOTES THE NUMBER OF IMAGES IN 15 LARGEST CLASSES IS THE UPPER LIMIT OF AMOUNT OF COMPUTATION

Dataset	A	B
The number of images	1,170,000	100,000
N	5	10
Rk	400,350,300,250, 200,170,150,130	400,350,300, 250,200,170
Lmax	2000	2000
K	8	6
The number of class	2951	204
T15	113432	46205

In our system, we extract the 15 nearest classes as computing set. For each query, we select the top 100 results but only show the top 12 in the paper due to the space limitation. We tested it with comparing the change of calculation amount and accuracy of results before and after using this algorithm and select three query images mentioned above. The statistics of these test cases is shown in Table 4 and actual performance in Fig. 6, 7 and 8 while the query images are shown in Fig. 5.

Although the number of images in 15 largest class is 113432 for the dataset A, only about 20000 images in 30 query will be computed as usual because high-density classes always don't get together.

Compare to other main methods with similarity purpose, such as HC (hierarchical clustering)[11] and SVM[13], our hierarchical classification method has a great advantage in index building. According to algorithm mentioned above, the time of index building of our method is the sum of two parts: Tlsi and Tclassification. Here Tlsi is the time to search LSIs and Tclassification. is the time to classify images.Therefore the total time is:

$$T_{index} = T_{lsi} + T_{classification} = O(N\ n) + O(K_{max}\ n)$$

Here n is the number of images of dataset, N and Kmax has mentioned in Table 1. The time complexity of most of other algorithms including HC and SVM is

$$O(n^2) >> O(N\ n) + O(K_{max}\ n)$$

It's not hard to see that our methods is better suited for massive images than HC and SVM.

Query A Query B Query C

Fig. 5. Three query in test case.

Before classifying

After classifying

Fig. 6. The result of query A.

Before classifying

After classifying

Fig. 7. The result of query A.

It is not difficult to discover the accuracy of search results after classifying influenced by the complexity of query images and target images in test database. Query A is composed by much various colors and complex shape

compared to query B and query C following the accuracy less than two other query. And the solution of this problem has become the focus of our next step.

Before classifying

After classifying

Fig. 8. The result of query A.

TABLE Ⅳ: THE QUERY Q SHOWN ABOVE, AND THE AMOUNT OF COMPUTATION BEFORE AND AFTER USING THIS ALGORITHM MB AND MA, AND THE PRECISION RATIO OF RESULTS PR IN 100 RESULT IMAGES

q	mb	ma	pr
A	1170551	20689	60%
B	1170551	23496	100%
C	1170551	18200	81%

VI. CONCLUSION

In this paper, we proposed an algorithm for content based image retrieval by hierarchical classification that is used to reduce amount of calculation in ergodic search. Our systems makes use of the directivity of images in high-dimension space and select the least similar images (LSI) in every subset as the classification center points. Meanwhile high-dimension indexing is applied to solve the problem that the different density of image space needs to match different number of class. We experimented with a practical used database consisting of approximately 1,170,000 images and its subset consisting 100,000 images. In our experiments, we used a feature combination that we design for decreasing the impact of slight change of background and shape of images as the similarity measure for computing the similarity of images in the database with a query image. Compare to the results without classification, we found that the proposed algorithm give better efficiency for a acceptable retrieval accuracy range.

ACKNOWLEDGMENT

F. A. Author is grateful for support from the subject" Recognition and Development Trend Study based on Network Community Hot Issue "of NLSDE, BUAA.

REFERENCE

[1] 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.

[2] S. A. Chatzichristofis and Y. S. Boutalis, “CEDD: Color and edge directivity descriptor, a compact descriptor for image indexing and retrieval,” in *Proc. of ICVS'08*, Santorini, Greece, vol. 5008, pp. 312 – 322, 2008.

[3] P. H. C. Guerra, A. Veloso, W. Meira, and V. Almeida, “From bias to opinion: A transfer-learning approach to real-time sentiment analysis,” in *Proc. of KDD'11*, pp. 150-158, New York, USA, 2011.

[4] N. Katayama and S. Satoh, “The SR-tree: an structure for high-dimensional nearest neighbor queries,” in *Proc. of SIGMOD' 97*, pp. 369-380, New York, USA, 2009.

[5] S. M. Zakariya, R. Ali, and N. Ahmad, “Unsupervised content based image retrieval by combining visual features of an image with a threshold,” in *Proc. of ICCT'10*, pp. 204-209, Bhubaneswar, India, 2010.

[6] T. Deselaers, D. Keysers, and H. Ney, “Features for image retrieval: An experimental comparison,” *Journal Information Retrieval*, vol. 11 no. 2, pp. 77-107, 2008.

[7] A. Vailaya, M. A. T. Figueiredo, A. K. Jain, and H.-J. Zhang, “Image classification for content-based indexing,” *IEEE Trans. Image Processing*, vol. 10, no. 1, pp. 117 – 130, 2001.

[8] C. S. Won, D. K. Park, S. J. Park, “Effcient use of MPEG-7 edge histogram descriptor,” *ETRI Journal*, vol. 24, no. 1, pp. 23 – 30, Feb. 2002.

[9] M. Bober, “MPEG-7 visual shape descriptors,” *Journal IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 716-719, 2001.

[10] R. Datta, D. Joshi, J. Li, and J. Z. Wang, “Image retrieval: Ideas, influences, and trends of the new age,” *Journal ACM Computing Surveys*, vol. 40, no. 2, 2008.

[11] S. Krishnamachari and M. A. Mottaleb, “Hierarchical clustering algorithm for fast image retrieval,” in *Proc. SPIE Conf. Storage and Retrieval for Image and Video Databases VII*, San Jose, CA, pp. 427 – 435, January, 1999.

[12] I. Markov, “VP-tree: Content-based image indexing,” in *Proc. of IJCNN 2004*.

[13] K. K. Seo, “An application of one-class support vector machines in content-based image retrieval,” Expert System wit.

Qiao Liu received the E.E. degree in software engineering, from Nankai University, Tianjin, China, in 2011, respectively. Now he is a graduate student with State Key Laboratory of Software Development Environment, the Department of Computer Science and Engineering, Beijing University of Aeronautics and Astronautics, Beijing, China. His research interests are in the fields of information retrieval and Distribution System.

Jiangfeng Chen received the E.E. degree in Flight Vehicle Design and Applied Mechanics, and the Ph. D. degrees in computer science and technology, all from Beijing University of Aeronautics and Astronautics (BUAA), Beijing, China, in 1998, 2008, respectively. He held a visiting position with the Department of Computing Science, the University of Illinois and the University of Washington. Now he is an Assistant Professor with State Key Laboratory of Software Development Environment, the Department of Computer Science and Engineering , BUAA. His scientific interests are in the fields of information retrieval and computer network.

Hui Zhang received the E. E., M.S. and Ph.D. degrees in computer science and technology, all from Beijing University of Aeronautics and Astronautics (BUAA), Beijing, China, in 1989, 1991 and 1994, respectively. Since 1994, he has been with the Department of Computer Science and Engineering, BUAA. Since 1998, he has been deputy director of Network Center of BUAA. Now he is the deputy director of State Key Laboratory of Software Development Environment, the Department of Computer Science and Engineering, BUAA. His scientific interests are in the fields of computer network, Internet information retrieval and mass data mining.