A Content-based Parallel Image Retrieval System

Zhou Bing
Institute of Image Processing and Pattern Recognition
Henan University
Kai Feng, China
e-mail: zhoub1631@163.com

Yang Xin-xin
School of Computer and Information Engineering
Henan University
Kai Feng, China
e-mail: yangxinxin38@163.com

Abstract—**As we all know, the content-based image retrieval (CBIR) is very time-consuming due to the extraction and matching of high dimensional and complex features. The traditional CBIR systems could not respond to a very large number of retrieval requests at mean time, which are submitted from the Internet. In this paper, we propose a content-based parallel image retrieval system to achieve high responding ability. Our system is developed on cluster architectures. It has several retrieval servers to supply the service of content-based image retrieval. Our system adopts the Browser/Server (B/S) mode. The users could visit our system though web pages. Our system uses the Symmetrical Color-Spatial Features (SCSF)[25] to represent the content of an image. The SCSF is effective and efficient for image matching because it is independent of image distortion such as rotation and flip as well as it increases the matching accuracy. The SCSF was organized by M-tree[16], which could speedup the searching procedure. Our experiments show that the image matching is quickly and efficiently with the use of SCSF. And with the support of several retrieval servers, the system could respond to many users at mean time.** *(Abstract)*

Keywords-Content-based Image Retrieval; Cluster Architecture; Color-Spatial Feature; B/S mode; Task Parallel; Internet (key words)

I. INTRODUCTION

With the wide widespread of digital images on WWW, the need of finding an image from the Internet has been increasing rapidly. Though there have been some information searching systems, such as google, these systems use the text-based retrieval techniques. This method could not retrieval the image efficiently due to it is very hard to describe an image with some words. If considering with the different cultures and languages, it is unimaginable to find images with some key words. In the recent years, content-based image retrieval techniques have been proposed to overcome the limitations of the text-based retrieval techniques.

The content-based image retrieval systems (CBIR) have been studied widely. Some works focused on how to represent an image, which means how to extract the visual features of an image. Some other works focused on how to understand an image, which means how to extract the objects in an image and describe the relationship between objects. All these works emphasize on the accuracy of the retrieval, but pay very little attention on the ability of responding to a huge mount of requests. However, under the WWW environment, this responding ability is very important, because lots of users will submit their retrieval requests at same time through the Internet. If the responding ability is not enough, the users will wait a long time, even could not get service due to time-out.

There are two ways to increase the responding ability. The first is to speedup the procedure of content retrieval. However the content-based image retrieval is very time-consuming due to the extraction and matching of high dimensional and complex features. It is hard for the traditional system to increase the responding ability. The second way is to study new parallel architectures for CBIR systems.

In this paper, we propose a content-based parallel image retrieval system to achieve high responding ability. Our system is developed on cluster architectures. It has several retrieval servers to supply the service of content-based image retrieval. Our system adopts the Browser/Server (B/S) mode. The users could visit our system though web pages. Our system uses the Symmetrical Color-Spatial Features (SCSF) [25] to represent the content of an image. The SCSF is effective and efficient for image matching because it is independent of image distortion such as rotation and flip as well as it increases the matching accuracy. The SCSF was organized by M-tree[16], which could speedup the searching procedure. Our experiments show that the image matching is quickly and efficiently with the use of SCSF. And with the support of several retrieval servers, the system could respond to many users at mean time.

The rest of this paper is organized as follows. Section 2 reviews some related works. In section 3, our system is stated in details. Section 4 is the experiment results. Finally, we conclude in section 5 with directions for further works.

II. RELATED WORKS

There have been many content-based image retrieval systems include QBIC[1], Virage[2], VisualSEEk[3], Photobook[4], Chabot[5] and VIPER[6]. Some special CBIR systems have been used in many applications, such as medical imaging[7], multimedia[8], art history[9], geology[10] and satellite image databases[11]. With the development of Internet, some CBIR systems on the mode of network have been proposed[12, 13, 14, 15]. One of the main features of these systems is that they use the mode of B/S. Supported by middle-ware, these system could be visited though Internet. However they do not consider the responding ability to the huge mount of requests coming

though networks. [12] and [15] use key words to filter the images. Though this could reduce the searching space and speedup the retrieval procedure, the accuracy will decrease due to the deficiencies of text-based image retrieval.

A CBIR system named CAIRO (Cluster Architecture for Image Retrieval and Organization) was proposed in [17]. As same as our system, CARIO is developed on cluster architectures, but this system does not support the B/S mode and has different parallel scheme from our system. CAIRO adopts computing parallel scheme. In CARIO, the images are distributed on each node, and each node only stores the features of its own images in a local database. For one image retrieval request, every node will try to find a set of similar images and return the results to a MASTER node, and after ranked by the MASTER node, the images are shown on the graphical user interface. On the contrary, our system adopts task parallel scheme. Though the images are distributed on each node, the features of all images are stored in one global database. The index structures of the features on each node are homogeneous. For one image retrieval request, the retrieval task is assigned to only one node, and other node could accept different tasks.

CARIO and our system have different characters. For the retrieval request of one user, since all the nodes take part into the computing, the user will get result very quickly. Within that time, if another user submits a request, the second one must wait until the first one finished. Compared to CARIO system, our system is slower for processing one request, but can respond to many users at the same time. One solution of combining these two systems is to divide all the nodes into several groups. Different tasks are assigned to different groups. And all the nodes in one group take part into the retrieval procedure together. This method is complex, and we will study it further.

III. THE PROPOSED SYSTEM

A. System architecture

Our system is a kind of the application of B/S mode, as depicted by Fig. 1. The image retrieval procedure has three steps. First step, user submits a sample image though the web pages of submission. Second step, the Feature Extraction module analyzes the sample image and extracts the features, which are utilized by the Image Matching module for searching of the similar images. Third step, the results are transformed to the web pages of result and shown on the browser.

Correspondingly, in our system, there are three types of basic server: Web Server, Retrieval Server and Feature Storage Server. And because we have several retrieval servers, a Task Dispatcher is needed. The feature storage server only stores the features of images, and the images are distributed on each node as JPEG format files. The location of an image file is stored with its features. The organization of the servers is shown in Fig. 2.

B. Image feature and its organization

The image feature could be color, texture and shape. Color is the most widely used feature for image retrieval. It is independent of image distortion, and it is simple, fast and all-purpose, which makes it suitable for Internet. However it is clear that color alone is not sufficient to characterize an image, because if two images have the same colors but different color distribution, these two images could not be similar. Adding some spatial information of colors will greatly increase the accuracy of identifying an image. One simple scheme of adding spatial information is dividing an image into some sub-images, and the color-spatial information is implied by the color-histograms of these sub-images.

In reference [19], the image is divided into 9 rectangular regions with 3*3 grid, and each region has same weights. In order to emphasize on the center part of the image, [20] partitions the image into five parts as Fig. 3, and the center part has a higher weight.

SCSF[25] uses circles to partition an image. Because the circle is symmetrical, SCSF is independent of symmetrical image distortion. And also, the circle-partition could give more emphasis on the center part of an image.

Fig. 4 illustrates that region-partition is sensitive to symmetrical image distortion, but circle-partition is insensitive to symmetrical image distortion.

The SCSF is obtained by: firstly, dividing the image into nine parts of a circle, seven cirques and the rest corners with 8 circles; secondly, computing the color histogram of the nine parts and the color bins are quantized to 16 with the algorithm proposed in [26]. In order to calculate the similarity of two images, we first define the distance of two histograms as:

$$D_H(H1,H2)=\sum_{i=1}^{16}(\min_{j=1}^{16}\{D(ci,cj)\}*|P(ci)-P(cj)|) \quad (1)$$

where the function P(c) means the pixel ratio of color c, function D(ci, cj) means the distance of color ci and cj in 3-dimensional color space.

The similarity of two images is defined as:

$$D_I(I1,I2)=\sum_{i=1}^{9} wi * D_H(H1,H2) \quad (2)$$

where wi is 2 for circle and cirques, and 1 for the rest.

The index structure of the high dimensional features is another important issue. The most common used structures are R-tree family[1, 21] and k-d-tree family[22]. However, since the distance of two images is not always the distance of geometry, the index structure designed for metric space is more suitable for content retrieval systems[16, 23, 24].

In our system, the SCSF of all images are stored in the feature storage server. And when a retrieval server is starting, it groups the features into some clusters according to similarity, and builds a M-tree[16] as index for the clusters.

C. The task dispatcher

Since our system has many retrieval servers, a task dispatcher is needed to assign the tasks and balance the workload. The dispatcher has a list of the retrieval servers, which records the address of the retrieval servers and the summaries of the image size and number already assigned to a retrieval server. Because the time cost of extracting the features of an image is highly related to the size of the image, the scheme for task assignment is decided by the summary of

the image size. And if two retrieval servers have the same summary of image size, choosing the one that has a less image number. According to this scheme, we define a factor **F** for assignment as:

$$F_i = Summary_i + n_i / N \quad (3)$$

in which Fi means the value of F for the i-th retrieval server, Summaryi means the summary of image size for the i-th retrieval server, ni means the number of image for the i-th retrieval server, and N=Sum(n_i). For every task assignments, choosing the retrieval server, which has the smallest F, as the target server.

A retrieval server could "register to" or "logout from" the dispatcher flexibly, which support our system with great scalability. And the dispatcher also plays the role of troubleshooting. It broadcasts a "is alive" request to retrieval servers at a fixed interval. If one retrieval server dose not respond to this request for certain times, this retrieval server will be delete from the list of retrieval servers.

IV. EXPERIMENT RESULTS

The cluster platform used by our system is CPC-8, which was developed by the institute of Image Processing and Pattern Recognition at Henan University. The CPC-8 has 8 nodes and each node has a CPU of Core 2 Duo 2.67GHz, 1G main memories and a 80G hard disk. The operation system is Linux.

Among the 8 nodes, there is one WEB server, one Task Dispatcher and one DB server. The rest five nodes are Retrieval servers. The WEB server is Apache server. The DB system is MYSQL. The task dispatcher, retrieval server and web pages were developed by us. The task dispatcher is developed with JAVA language, and the retrieval server is developed with C language. The communication between them is realized with Sockets Programming. The web pages are developed with the combination of JAVA Servlet and JavaBean techniques.

Our system has 5 retrieval servers, and each server could generate some threads to accept and process tasks. Considering with the limitation of main memory resource, only two threads are allowed for each retrieval server. So our system could respond to 10 users at the mean time.

In order to test our system, we built an image database, which has 2,500 color images. We test the time cost of retrieval, which is defined as beginning at the moment that the whole sample image is received and ending at the moment that the searching procedure is finished. The time elapse between user submitting a sample image and receiving the result is hard to test, because it is also affected by the status of Internet and the client point.

We use some color images with different size to test the time cost of retrieval. The result is shown as the table 1.

TABLE I. THE RESULT OF THE TIME COST OF RETRIEVAL

	*360*270(Pixels)*	*640*427(Pixels)*	*640*800(Pixels)*	*1024*768(Pixels)*
Feature extraction(s)	5.59	19.38	34.99	56.63
mage searching(s)	0.34	0.33	0.32	0.34
Summary (s)	5.93	19.71	35.31	56.97

Fig. 5 shows an example of the retrieval result. The image at the top left corner is a sample image, and the left 8 images are return by our system.

V. CONCLUSION AND FURTHER WORK

In this paper, we propose a content-based parallel image retrieval system. Our system is developed on cluster architectures. It has several retrieval servers to supply the service of content-based image retrieval. Our system is one of the applications of the Browser/Server (B/S) mode. The users could visit our system though Internet. Our system uses the Symmetrical Color-Spatial Features (SCSF) to represent the content of an image. The SCSF is effective and efficient for image matching because it is independent of image distortion such as rotation and flip as well as it increases the matching accuracy.

Next, we will do lots of work to increase the reliability and stability of our system. Moreover, we will add the texture and shape features as an optional choice for users to increase the accuracy.

ACKNOWLEDGMENT

This research is funded by a research foundation of Henan university named Sheng Bu Gong Jian(SBGJ090602)

REFERENCES

[1] Wayne Niblack, R. Barber, W. Equitz, Myron Flickner, E. Glasman, Dragutin Petkovic, Peter Yanker, Christos Faloutsos, and G. Taubin, "The QBIC project: querying images by content using color, texture and shape," Proc. SPIE, Vol. 1908, 1993, pp. 173-187, doi:10.1117/12.143648

[2] J.R. Bach et al, "The virage image search engine: an open framework for image management," SPIE Proceedings of the Storage and Retrieval for Still Images and Video Databases IV, February, 1996, pp. 76-87.

[3] J. Smith and S.-F. Chang, "VisualSEEk: A fully automated content-based image query system," Proceedings of the Fourth ACM Multimedia Conference, ACM Press, New York, USA, November 1996, pp. 87-98.

[4] A. Pentland, R.W. Picard, and S. Sclaro, "Photobook: Tools for Content-based manipulation of image databases," International Journal of Computer Vision, vol. 13, 1996, pp. 233-254.

[5] V.E. Ogle and M. Stonebraker, "Chabot: retrieval from a relational database of images," IEEE Computer, vol. 28, 1995, pp. 40-48.

[6] B.C. Ooi, K.L. Tan, and C.Y. Yee, "An evaluation of color-spatial retrieval techniques for large image databases," Multimedia Tools and Applications, vol. 14, 2001, pp. 55-78.

[7] F. Korn, C. Faloutsos, N. Sidiropoulos, E. Siegel, and Z. Protopapas, "Fast nearest neighbor search in medical image databases," Proceedings of the 22th VLDB Conference, Mumbai, India, September 1996, pp. 215-226.

[8] S.W. Smoliar and H.J. Zhang, "Content-based video indexing and retrieval," IEEE Multimedia, vo. 1, 1994, pp. 62-72.

[9] T. Kato, "Database Architecture for Content-based image retrieval," SPIE Proceedings of the International Society for Optical Engineering, San Jose, CA, 1992, pp. 112-123.

[10] R. Shann, D. Davis, J. Oakley, and F. White, "Detection and Characterization of Carboniferous foraminifera for content-based retrieval from an image database," SPIE Proceedings of the Storage and Retrieval for Still Images and Video Databases I, February 1993, pp. 188-197.

[11] A. Kitamoto, C. Zhou, and M. Takagi, "Similarity Retrieval of noaa satellite imagery by graph matching," SPIE Proceedings of the Storage and Retrieval for Still Images and Video Databases I, February 1993, pp. 60-73.

[12] Hongmei Tang, Ming Yu, Zhitao Xiao, and Yingchun Guo, "A Content-based Image Retrieval System on the Mode of Network," IEEE Asia Pacific Conference on Circuits and Systems, 2000, pp. 422-425.

[13] B. Verma, P. Sharma, and S. Kulkarni, "An Intelligent On-line System for Content Based Image Retrieval," International Conference on Computational Intelligence and Multinedia Applications, 1999, pp. 273-277.

[14] Y. Alp Aslandogan and Clement T. Yu, "Automatic Feedback For Content Based Image Retrieval On The Web," IEEE International Conference on Multimedia and Expo., 2002, pp. 221-224.

[15] Wu Yi, Zhuang Yue-Ting, and Pan Yun-He, "Image Retrieval System for Web: Webscope-CBIR," IEEE Proceedings of the 11th International Workshop on Database and Expert Systems Applications, 2000, pp. 620-624.

[16] Paolo Ciaccia, Marco Patella, and Pavel Zezula, "M-tree: An Efficient Access Method for Similarity Search in Metric Spaces," in Proceedings of the 23rd VLDB Conference, Athens Greece, 1997, pp. 426-435.

[17] Odej Kao, "Parallel and Distributed Methods for Image Retrieval with Dynamic Feature Extraction on Cluster Architectures," Proceedings of 12th International Workshop on Database and Expert Systems Applications, 2001, pp. 110-114.

[18] Bo Xiaochen and Liu Jianping, "Research on Some Key Problems in Color Based Image Retrieval," Mini-Micro Systems. Vol. 19, Oct. 1998, pp. 42-47.

[19] Gong, Y., Zhang, H., Chuan, H.C., and Sakauchi, M, "An image database system with content capturing and fast image indexing abilities," in Proc. IEEE International Conference on Multimedia Computing and Systems, 1994, pp. 121-130.

[20] M.Stricker and A. Dimai, "Color Indexing With Weak Spatial Constraints," in SPIE Proceedings, vol. 2670, 1996, pp. 29-40.

[21] M. Nappi, G. Polese, and G. Tortora, "FIRST: Fractal Indexing and Retrieval SysTem for Image Databases," Image and Vision Computing, vol. 16, 1998, pp. 1019-1031.

[22] Chabane Djeraba and Marinette Bouet, "Digital Information Retrieval," in Proc. ACM CIKM 97, LasVegas USA, 1997, pp. 185-192.

[23] Uhlmann J. K, "Satisfying general proximity/similarity queries with metric trees," Inf. Process. Lett. Vol. 40, 1991, pp. 175-179.

[24] Tolga Bozkaya and Meral Ozsoyoglu, "Indexing Large Metric Spaces for Similarity Search Queries," ACM Tran. On Datanase Systems, Vol. 24, 1999, pp. 361-404.

[25] Bing Zhou and Shou-zhi Wei, "Use Symmetrical Color-Spatial Feature for Image Comparison," in Proceedings of the Third International Conference on Information Technology and Applications, Sydney, July, 2005, Vol. 2, pp. 357-360.

[26] Zhou Bing, Shen Jun-yi and Peng Qin-ke, "An Adjustable Algorithm for Color Quantization," Pattern Recognition Letters, vol. 25, 2004, pp. 1787-1797.

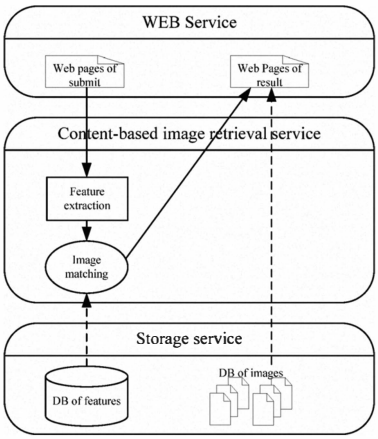

Figure 1. Content-based image retrieval of B/S mode

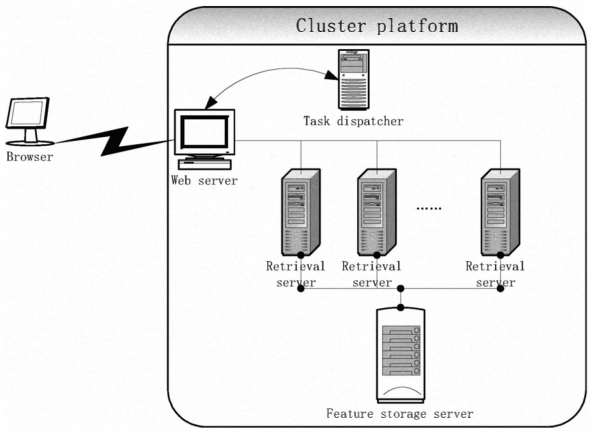

Figure 2. The organization of the servers

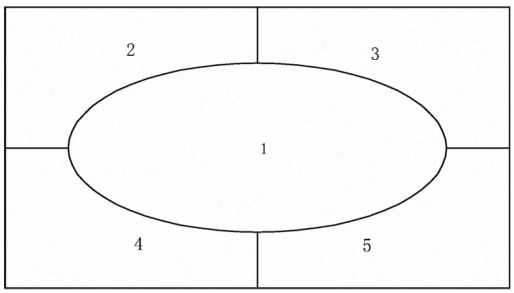

Figure 3. An image partition method

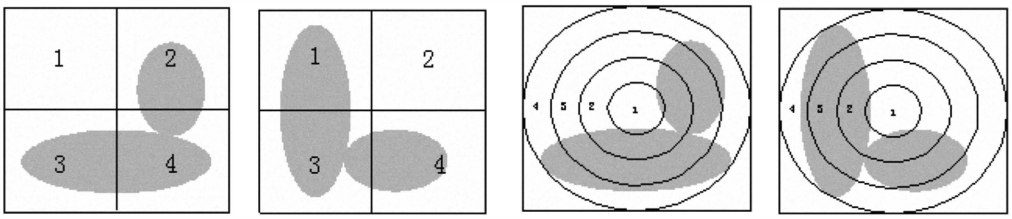

Figure 4. Illustrations of the difference of two partition methods

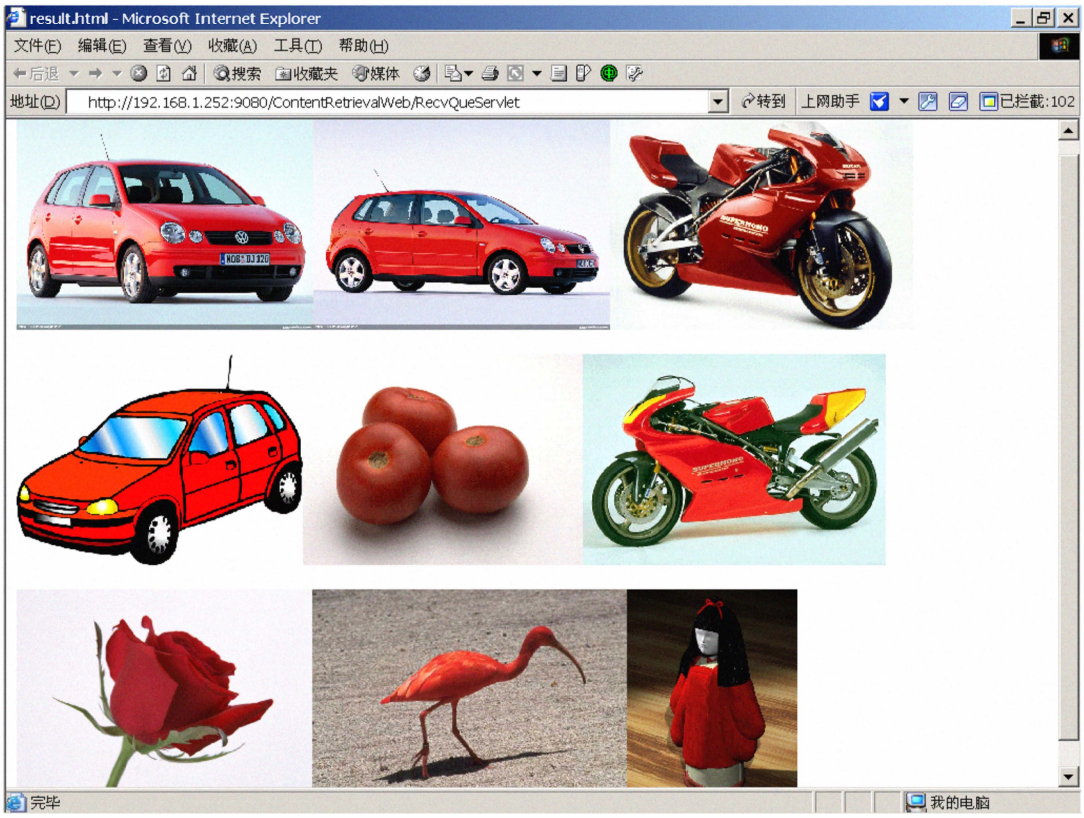

Figure 5. An example of retrieval result