A Ranking Algorithm Using Dynamic Clustering for Content-Based Image Retrieval*

Gunhan Park, Yunju Baek, and Heung-Kyu Lee

Division of Computer Science
Department of Electrical Engineering & Computer Science,
Korea Advanced Institute of Science and Technology
373-1 Kusung-Dong Yusong-Gu Taejon, 305-701, Republic of Korea
{gunhan,yunju,hklee}@rtlab.kaist.ac.kr

Abstract. In this paper, we propose a ranking algorithm using dynamic clustering for content-based image retrieval(CBIR). In conventional CBIR systems, it is often observed that visually dissimilar images to the query image are located at high ranking. To remedy this problem, we utilize similarity relationship of retrieved results via dynamic clustering. In the first step of our method, images are retrieved using visual feature such as color histogram, etc. Next, the retrieved images are analyzed using a HACM(Hierarchical Agglomerative Clustering Method) and the ranking of results is adjusted according to distance from a cluster representative to a query. We show the experimental results based on MPEG-7 color test images. According to our experiments, the proposed method achieves more than 10 % improvements of retrieval effectiveness in ANMRR(Average Normalized Modified Retrieval Rank) performance measure.

1 Introduction

According as multimedia data increases in recent years, effective and efficient methods for storing and retrieving multimedia data have been required. In particular, images are used as important information representation in a many variety of areas such as medicine, entertainment, education, trademark, fashion design, manufacturing, etc. Over the previous years, techniques for content-based image retrieval(CBIR) from image collection have been studied.

In conventional content-based image retrieval systems, images are represented by visual features, and the retrieval process is performed as calculating similarity between visual features of the query image and images from database. The retrieved results are shown as orders by similarity ranking algorithm in the practical CBIR system. Users evaluate the performance of the system by the ranked results. A ranking algorithm is an important component for CBIR systems. Unfortunately, it is often observed that visually dissimilar images have

* This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Advanced Information Technology Research Center(AITrc)

M. S. Lew, N. Sebe, and J. P. Eakins (Eds.): CIVR 2002, LNCS 2383, pp. 328–337, 2002.

higher ranking than visually similar images in the conventional CBIR systems[1]; we call it *ranking inversion*.

To remedy the ranking inversion problem, we re-calculate the similarity distance by grouping and analyzing retrieved results. Retrieval results can be classified into some sub-groups via dynamic clustering. The formed groups should have a high degree of association between members of the same group and a low degree between members of different groups.

In this paper, we use 2-step methods for improving retrieval performance. In the first step of our method, images are retrieved using visual features. Next, the retrieved images are analyzed using clustering, and adjusted similarity according to distance from a cluster representative to a query. In experiments, we show that the application of dynamic clustering over retrieved results can significantly improve retrieval performance in CBIR systems.

The rest of the paper is organized as follows. In section 2, we explain about an image retrieval model. In section 3, we define the clustering method for CBIR systems and describe steps of a ranking algorithm using dynamic clustering and its advantages in CBIR system. In section 4, we explain experimental environments, results, and performance evaluations. Finally, we conclude in section 5.

2 Image Retrieval Model

There are many visual features such as color, texture, shape, etc. for CBIR systems. Generally, visual features can be represented as a vector in a n-dimensional vector space. We denote the images as a feature vector as follows,

$$I = (f_1, f_2, f_3, \cdots, f_n) \tag{1}$$

where f_n is the element of visual features.

We can define similarity functions as one of vector space distance models, and the definition is as follows,

$$D(I, I') = \sum_{k=1}^{n} d(f_k, f'_k) \tag{2}$$

$d()$ function is one of that similarity measures such as absolute difference($L1$ norm)[2], square root($L2$ norm), quadratic distance($L2$-related norm)[3], and so on.

Using these definitions, the typical ranking method in image retrieval is defined as follows,

$$\text{Sorting}\{D(I, I')\} \text{ where } I = \text{ query image }, I' \in \text{ image DB}. \tag{3}$$

Our goal is to improve the $D()$ function in equation (2) via clustering analysis of retrieved results.

3 The Ranking Algorithm Using Dynamic Clustering

A typical CBIR system retrieves and ranks images according to a similarity function based on a feature vector distance model. In this paper, we define another properties in deciding ranking of results. As we use dynamic clustering methods about retrieved images, we make relevant groups that contain similar images. Using the groups, we analyze similarity relationship of retrieved results and the query image. The ranking and the similarity value of retrieved images are adjusted according to the cluster analysis.

The proposed method is depicted in Fig. 1. In the first step, we calculate difference value using vector distance between visual feature vectors. In the second step, we apply the dynamic clustering method to the retrieved results of the first step and the query image. We make a tree structure of hierarchical agglomerative cluster. We select a cluster representative to compute distance between query image and the cluster. After we investigate cluster analysis, we adjust the similarity distance from the query image.

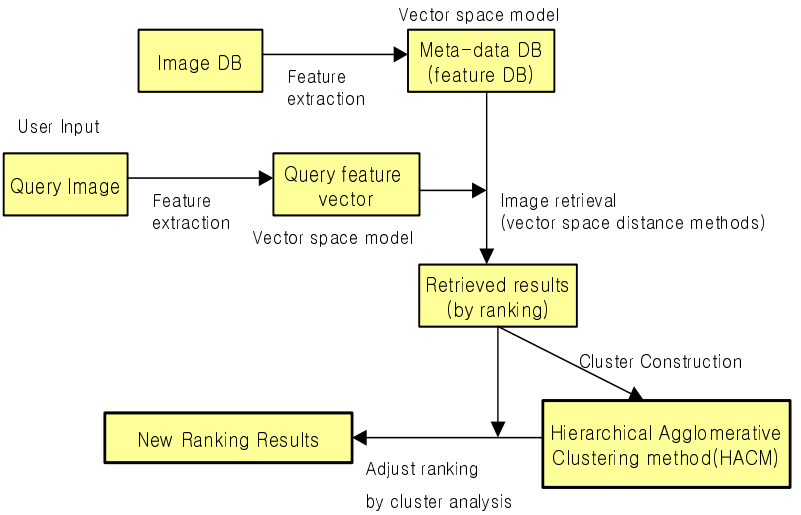


Fig. 1. Overview of the proposed method

There has been some research on how to employ clustering to improve retrieval performance in document information retrieval field. Hearst[6] shows that clustering method is effective in browsing retrieved results using document subgroups and summary text. Lee[7] shows that document clustering improve performance significantly. These results show that clustering is an effective method in document information retrieval. In case of CBIR systems, as images are represented as vectors, clustering can be an effective factor for retrieval performance improvement.

In general, there are two kinds of method in clustering methods. One is a hierarchical method, and the other is a non-hierarchical method. And the hierarchical method has two kinds of methods, agglomerative and divisive. In non-hierarchical clustering methods, the initial value and the number of clusters must be predefined. Non-hierarchical clustering method shows good performance for fixed number of clusters. But the flexible clustering method is needed for information retrieval. As these reasons, hierarchical agglomerative clustering methods(HACM) are frequently used in information retrieval field[6].

The cluster structure resulting from a HACM is made as a tree structure : *dendrogram*. There are several methods to make a dendrogram for HACM such as single link, complete link, Ward'd method, etc. In this paper, we use *Ward's method* for constructing dendrogram based on stored similarity matrix and Lance-Williams update formula[5]. In Ward's method, images join the cluster that minimizes the increase in the total within-group error sum of squares. It relatively makes an un-biased clustering tree.

After dynamic clustering using retrieved results, we modify the distance value of the results according to a equation (4).

$$D'(I, I') = \alpha D(I, I') + \beta Dc(I, I') \tag{4}$$

where $Dc(I, I')$ is distance from the query image to the cluster I'.

According to clustering hypothesis, more similar images are divided into same cluster and irrelevant images are divided into different cluster. Distance value is adjusted, as images in same cluster with query image have small distance value, otherwise images have large distance value as shown in equation (4).

In our method, clustering result is very important. There are some factors in considering. The factors are as follows,

- Cut-off size(N) : how much high rank images are clustered?
- Clustering construction threshold(T) : what is the best dividing value at agglomerative hierarchical clustering tree?
- Combining value : what is the more important element in distance computation?(α, β in equation (4))

Considering to these factors, we evaluate and investigate the effect of various methods and parameters in the next section.

4 Experimental Results

4.1 Experimental Environments

In our experiments, we use around 5,000 images from MPEG-7 experimental data set to form the image database. This test set includes a variety of still images which include stock photo galleries, screen shots of television programs, and animations etc. In our experiments, number of queries was about 1% of the

number of images in the database. A set of 50 common color queries(query sets), each with specified ground truth images(manually predefined truth images), is used.

We use color histogram as the visual features for first retrieval step; 128 color bins of HSV color space is used. The distance measure in our experiments is $L1$ norm[2].

4.2 Retrieval Effectiveness Evaluation Measure: ANMRR

For performance evaluation, there is no standard measure like PSNR in image processing. There are several evaluation measures such as precision/recall graph, simple ranking method, precision/recall with scope, etc. for CBIR systems. In our experiments, we use a kind of ranking measure, ANMRR(Average Normalized Modified Retrieval Rank) that is defined from MPEG-7 research group. The ANMRR value is a normalized ranking method. This value is defined as follows[8,9].

First, we denote NG(q), K(q), R(k) as follows,

- $NG(q)$: the number of the ground truth images for a query q.
- $K(q) = min(4 * NG(q), 2 * GTM)$, Where GTM is max$\{NG(q)\}$ for all q's.
- $R(k)$ = rank of an image k in retrieval results.

Rank(k) is defined as follows,

$$Rank(k) = \begin{cases} R(k) & if\ R(k) \leq K(q) \\ (K+1)\ if\ R(k) > K(q) \end{cases} \tag{5}$$

Using equation (5), AVR(Average Rank) for query q is defined as follows:

$$\text{AVR}(q) = \sum_{k=1}^{NG(q)} \frac{\text{Rank}(k)}{NG(q)} \tag{6}$$

However, with ground truth sets of different size, the AVR value depends on $NG(q)$. To minimize the influence of variations in $NG(q)$, MRR(Modified Retrieval Rank) is defined as follows,

$$\text{MRR}(q) = \text{AVR}(q) - 0.5 - \frac{NG(q)}{2} \tag{7}$$

The upper bound of MRR depends on $NG(q)$. To normalize this value, NMRR(Normalized Modified Retrieval Rank) is defines as follows,

$$\text{NMRR}(q) = \frac{\text{MRR}(q)}{K + 0.5 - 0.5 * NG(q)} \tag{8}$$

NMRR(q) has values between 0(perfect retrieval) and 1(nothing found). And evaluation measure value for whole set over query sets, ANMRR(Average Normalized Modified Retrieval Rank) is defined as follows,

$$\text{ANMRR}(q) = \frac{1}{Q} \sum_{q=1}^{Q} \text{NMRR}(q) \tag{9}$$

4.3 Results

The goal of the experiments is to validate the proposed method. In order to evaluate the performance of the proposed method, we change the parameters of clustering, clustering threshold(T), distance weight(β), and cut-off size(N). We use a centroid of cluster as a cluster representative for calculating the distance from a cluser to a query. The results are shown in Table 1 - Table 3. The ANMRR value of an initial method, that is, the method without clustering is 0.904. The experimental results show that the ANMRR value is improved by more than 10% comparing to that of an initial method.

In table 1, the T value represents the same meaning of the number of clusters. The smaller T value has the larger number of clusters. As shown in table 1, too small T value or too large T value cannot influence the improvement of the overall performance of the system. In other words, too small number of clusters or too large number of clusters show same results of the method without clustering. Table 2 shows the influence and the weights of cluster analysis and first step retrieval method. In experiments, we first fix the α value, and then adjust β value. The ANMRR value is similar for any β values in table 2. But in case of large β value, the result is not improved comparing to the method without clustering. These results show the only cluster analysis cannot improve performance of systems. In table 3, we change the cut-off size. In case of small cut-off size, the performance is not improved because the system cannot perform the cluster analysis using small retrieved results. Also, too large cut-off size cannot improve performance of systems because the clustering results contain many irrelevant images.

Table 1. Comparison of the performance of different T values, where N=100, β=0.5

T	0.6	0.8	1.0	1.2	1.4	1.6	1.8
ANMRR	0.0858	0.0795	0.0816	0.0818	0.0823	0.0793	0.0801
improvement	+5%	+12%	+9%	+9.5%	+8.9%	+12.2%	+11.3%

Table 2. Comparison of the performance of different β values, where $T = 1.6$

β	1.0	0.75	0.35	0.25
ANMRR	0.0826	0.0799	0.0777	0.0814
improvement	+8.6%	+11.6%	+14%	+9.9%

Table 3. Comparison of the performance of different N values, where $T = 1.6$, $\beta = 0.35$

N	80	120	140	160	180	200
ANMRR	0.0810	0.0788	0.0782	0.0784	0.0777	0.0826
improvement	+10.3%	+12.8%	+13.4%	+13.2%	+14%	+8.6%

The retrieval examples without clustering and our method are shown in Fig. 2, Fig. 3. It is clear that relevant images to the query are located at higher rank in the proposed method than the method without clustering. See the image at rank 3, rank 10 in Fig. 2(b). and rank 7, rank 8 in Fig. 3(b). It shows visually performance improvement of the proposed method.

The results indicate significant performance improvement using the dynamic clustering mechanism in CBIR systems. As analyzing experimental results, we show evidence validating our method is effective in CBIR systems.

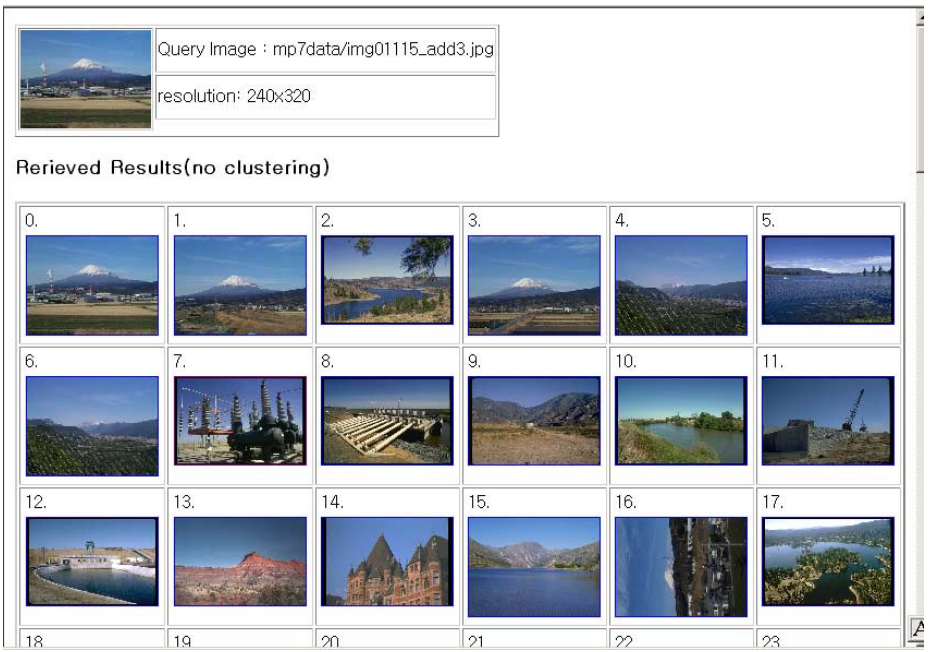
5 Conclusions

In this paper, we present an efficient ranking algorithm using dynamic clustering for image retrieval. Experimental results show that our method improves more than 10% retrieval performance in ANMRR measure. In the future work, we will use several visual features such as texture, shape, color layout features and different clustering strategy.

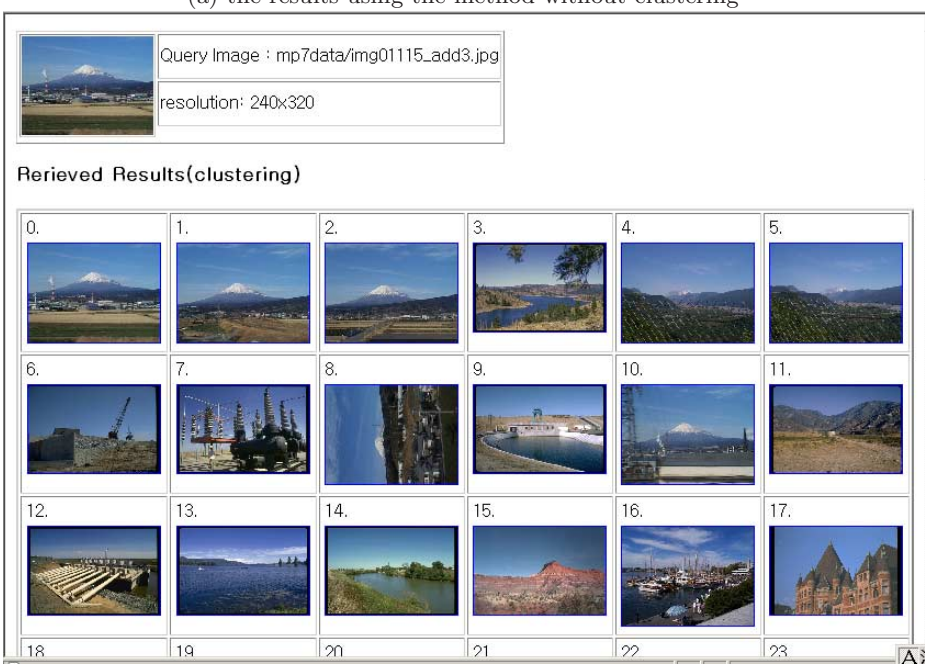
References

1. Yong Rui, Thomas S. Huang, and Shih-Fu Chang, "Image Retrieval : Current Techniques, Promising Directions, and Open Issues," *Journal of Visual Communication and Image Presentation*, Vol 10, pp. 39 62, March, 1999. 329
2. Michael J. Swain and Dana H. Ballad, "Color Indexing," *International journal of computer vision*, Vol. 7, No. 1, p.11 - p.32, 1991. 329, 332
3. Christos Faloutsos, Ron Barber, Myron Flickner, Jim Hafner, Wayne Niblack, Dragutin Petkovic and Will Equitz "Efficient and Effective Querying by Image Content," *Journal of Intelligent Information Systems*, 3, 3/4, pp. 231-262, July 1994. 329
4. J. R Smith, "Integrated Spatial and Feature Image Systems : Retrieval, Analysis and Compression," Doctoral Dissertations, Columbia University, 1997.
5. William B. Frakes, and Ricardo Baeza-Yates, "Information Retrieval : Data Structures & Algorithms," Prentice Hall, 1992. 331
6. Marti A. Hearst, and Jan O. Pederson, "Reexamining the Cluster Hypothesis : Scatter/Gather on Retrieval Results," *Proceedings of 19th ACM SIGIR International Conference on Research and Development in Information Retrieval*, pp. 76-84, 1996. 330, 331

7. Kyung-Soon Lee, Young-Chan Park, and Key-Sun Choi, "Re-ranking model based on document clusters," *Information Processing and Management*, vol. 37, pp.1-14, 2001. 330
8. B. S. Manjunath, Jens Rainer Ohm, Vinod V. Vasudevan, and Akio Yamada, "Color and texture descriptors," *IEEE Trans. On circuits and systems for video technology*, Vol. 11, No. 6, pp. 703 715, June 2001. 332
9. V. V. Vinod and B. S. Manjunath, *Report on AHG of color and texture*, ISO/IEC/JTC1/SC29/WG11, Doc. M5560, Maui, December 1999. 332

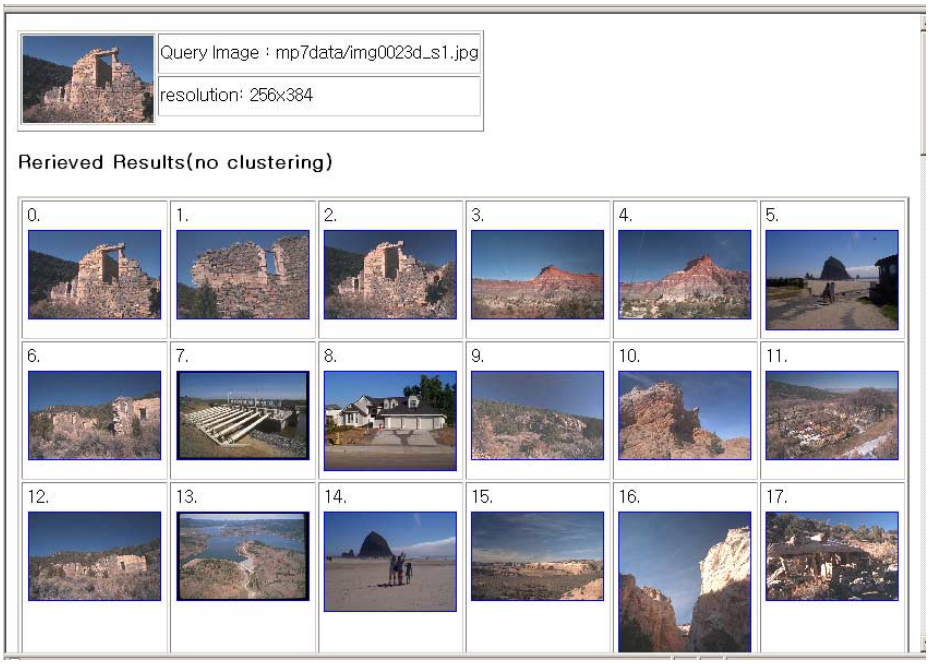


(a) the results using the method without clustering

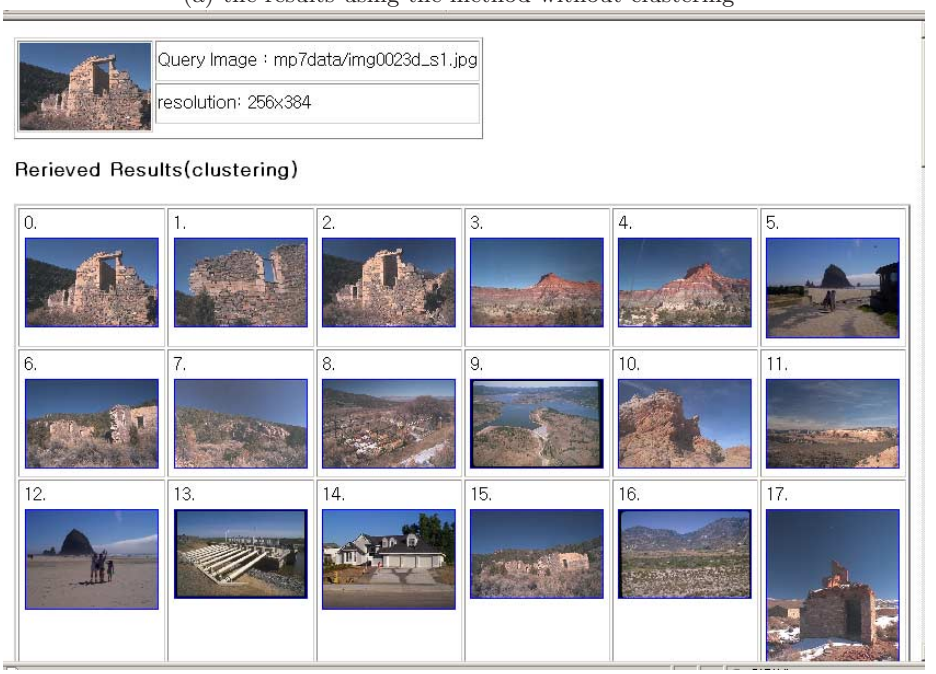


(b) the results using the proposed method

Fig. 2. The retrieval example using query set no. 35



(a) the results using the method without clustering



(b) the results using the proposed method

Fig. 3. The retrieval example using query set no. 50