Efficient Content-Based Image Retrieval Methods Using Color and Texture

Sang-Mi Lee, Hee-Jung Bae, and Sung-Hwan Jung

CONTENTS

ABSTRACT

In this paper, we propose efficient content-based image retrieval methods using the automatic extraction of the low-level visual features as image content. Two new feature extraction methods are presented. The first one is an advanced color feature extraction derived from the modification of Stricker's method. The second one is a texture feature extraction using some DCT coefficients which represent some dominant directions and gray level variations of the image. In the experiment with an image database of 200 natural images, the proposed methods show higher performance than other methods. They can be combined into an efficient hierarchical retrieval method.

I. INTRODUCTION

Due to the rapid advance in computer technology for multimedia systems, the amount of audiovisual information is increasing rapidly. Currently, however, searching for multimedia content is not possible even though text-based search is available. Therefore, the new MPEG project, MPEG-7, is working to provide solutions for identifying multimedia content [1]. Recently, along with that activity, a content-based approach emerged to retrieve images efficiently [2]-[4]. It is an image retrieval methodology using automatically generated content feature information, instead of any additional captions or text information. The content-based methods have various application areas, for example, digital library, tourist information, investigation services, medical applications, and verification of a trade mark or logo, and so on [1], [5].

Generally, the low-level characteristic values, shape, color, and texture have been used as main content features for content-based image retrieval. The use of shape features has largely been limited to specialized retrieval systems, because its domain-specific database has to be set up manually beforehand for effective retrieval [6]. The existing approaches using color features mostly use histograms [7]-[10]. The histogram-based methods provide general image characteristics. However, it is hard to represent localized features in detail. Most existing approaches using texture features also require large storage space and a lot of computation time to calculate the matrix of features such as SGLDM (Spatial Gray Level Dependence Matrix) [11].

Therefore, in this paper, we propose efficient content-based image retrieval methods using color and texture to improve the above mentioned problems. First, a modified Stricker's color feature extraction is proposed for considering localized features. Second, some DCT coefficients from transform domain are directly used for extracting texture features efficiently. It does not need additional complex computation for feature extraction. The proposed method can be directly applied to image data in the compressed domain. This may be a way to solve the large storage space problem and the computational complexity of the existing methods.

The remaining of the paper is organized as follows: In Section II, a general content-based image retrieval system is introduced. In Section III, the proposed image retrieval system is presented. In Section IV, experimental results are shown. Finally, in Section V, we conclude with some remarks.

II. GENERAL IMAGE RETRIEVAL SYSTEM

The general image retrieval system is shown in Fig. 1. The system consists of three main modules: the input module, query module, and retrieval module.

In the input module, the feature vector

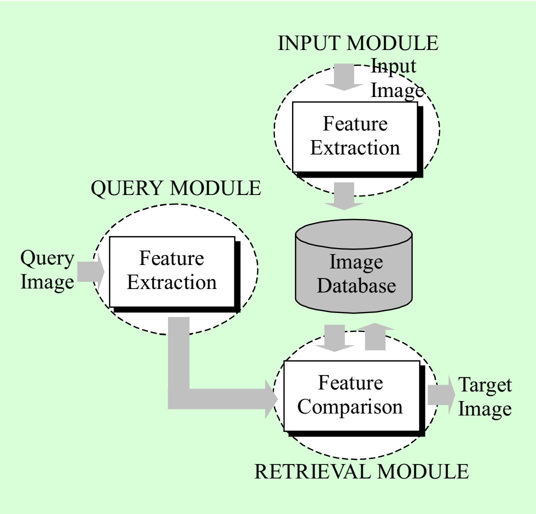

Fig. 1. The block diagram of a general image retrieval system.

is extracted from each input image. It is then stored in an image database with its input image. On the other hand, when a query image enters the query module, the feature vector of the query image is extracted. In the retrieval module, the extracted feature vector of the query image is compared with the feature vectors of stored images in the image database. As a result of the query, similar images are retrieved according to their similarity with the query image. Finally, a target image will be obtained from the retrieved images. In our system, we considered color and texture for feature extraction. Therefore, in the feature comparison, a hierarchical retrieval is possible by applying the two features one after the other. A selective retrieval is also possible by using just one of the two feaures.

III. IMAGE RETRIEVAL USING COLOR AND TEXTURE FEATURES

Most large image database systems require efficient feature comparison as well as feature extraction to provide a reasonable response to an image query. The exhaustive search of every image in a large image database can be very expensive. Therefore, in our system, by applying hierarchical retrieval using color and texture, it is possible to reduce the search space in a large image database. Figure 2 shows the block diagram of the proposed method. If a query image enters the system, image conversion for processing, block processing for considering local characteristics, and feature vector extraction for retrieval are sequentially performed. Using one of two feature vectors, color and texture, a selective retrieval can be performed as shown in Fig. 2. A hierarchical retrieval can also be performed. A color category code is generated from adaptive resonance theory (ART) 2 [13] neural network classifier using a color feature vector as input. At first, the global retrieval is done, using the color category code.

After global retrieval, the images which have the same color category code as that of the query image are selected. Then, the local retrieval within the selected images is done, using a texture feature vector. Finally, a target image can be obtained from the images with the same color cate-

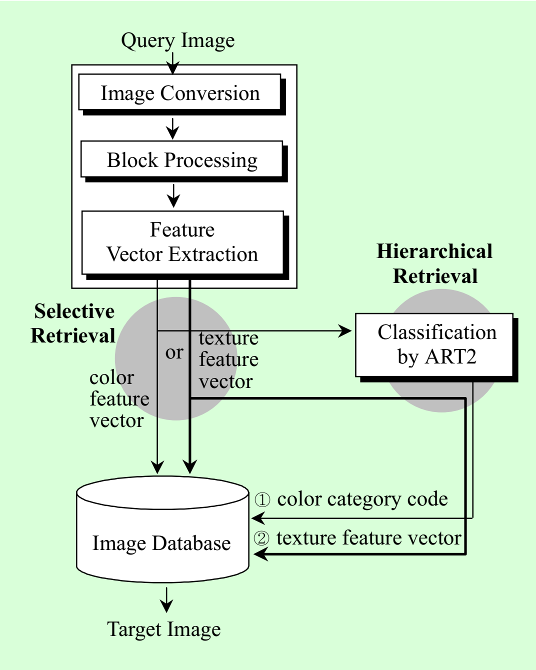

Fig. 2. The block diagram of the proposed retrieval method.

gory code, instead of from all the images in database.

1. Color Feature Extraction

There have been researches using color feature as a query key for image retrieval. In general, they use color histogram methods to obtain color feature information from an image. Color histogram indexing was introduced by Swain *et al.* [8]. They used the brightness values of RGB color space and stored coarsely quantized color histogram of images as indices. Funt *et al.* modified Swain's method to improve its performance by using the histogram of the RGB color ratio from neighboring locations [12]. In general, histogram-based methods provide overall image characteristics. However, they show high sensitivity to illumination and can not represent localized features well. We propose an advanced color extraction method to compensate the above drawbacks of histogram-based methods. The block diagram of the color feature extraction is shown in Fig. 3.

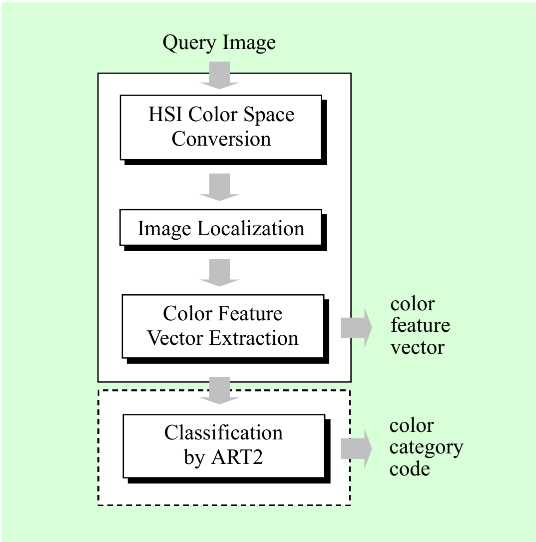

Fig. 3. The block diagram of the color feature extraction.

First, when a query image is presented, its RGB color space is converted into HSI (Hue, Saturation, and Intensity) for better retrieval performance. Next, each image is divided into subblocks for considering spatial location information. Then, we can get a color feature vector from each subblock. Using the color feature vectors and the measure of similarity mentioned in Section IV, we can retrieve similar images from an image database with color feature only. For hierarchical retrieval, on the other hand, the color feature vectors are fed into an ART2 neural network for classification [13], [14]. By classification, the color category code of a query image is generated from its color feature vector. Color category code will be used for global search in hierarchical retrieval.

A. The Color Model Conversion and Subblock Division

Each RGB input image is converted into a HSI image using the following:

$$\begin{aligned}
\mathrm{H} &= \cos^{-1}\left\{\frac{\frac{1}{2}[(\mathrm{R}-\mathrm{G})+(\mathrm{R}-\mathrm{B})]}{[(\mathrm{R}-\mathrm{G})^2+(\mathrm{R}-\mathrm{B})(\mathrm{G}-\mathrm{B})]^{\frac{1}{2}}}\right\}\\
\mathrm{S} &= 1-\frac{3}{(\mathrm{R}+\mathrm{G}+\mathrm{B})}[\min(\mathrm{R},\mathrm{G},\mathrm{B})]\\
\mathrm{I} &= \frac{1}{3}(\mathrm{R}+\mathrm{G}+\mathrm{B}). \qquad (36)
\end{aligned}$$

To obtain spatial localization information in the input image, it is divided into $m \times n$ size subblocks.

B. Color Feature Vector Extraction

Stricker proposed a color extraction method using the three moments of each color channel of an image: average, standard deviation and skewness [10]. For the feature vector extraction, its color features are computed by the following equations:

$$E_i = \frac{1}{m \cdot n}\sum_{j=1}^{m \cdot n} P_{ij} \qquad (37)$$

$$\sigma_i = \left[\frac{1}{m \cdot n}\sum_{j=1}^{m \cdot n}(P_{ij}-E_i)^2\right]^{\frac{1}{2}} \qquad (38)$$

$$\alpha_i = \left[\frac{1}{m \cdot n}\sum_{j=1}^{m \cdot n}(P_{ij}-E_i)^3\right]^{\frac{1}{3}}, \qquad (39)$$

where E_i is an average of each color channel ($i =$ H,S,I), σ_i is a standard deviation, and α_i is a skewness. P_{ij} is the value of each color channel at the *j-th* image pixel. $m \cdot n$ is a total number of pixels per image.

Stricker used the 9 feature vector which consists of three moments for each color channel, H, S, I, respectively. The histograms are quantized to 16 bins for H channel and 4 bins for S and I channels, respectively.

We propose a modification to the color extraction algorithm proposed by Stricker, to improve performance as follows: (a) to use the color feature vectors considering spatial location information, we apply Stricker's method to each subblock; (b) we reduce the number of bins for the index in S and I channel to 2 while keeping the 16 bins for index in H channel, because S and I do not much affect the retrieval performance directly, (c) we only use the first moment for S and I channels except for H, for which we calculate also the second and third moments. Therefore, 5 color features are used instead of Stricker's 9 features.

For the *k-th* $m \times n$ subblock, we use the 5 features vector: the average($E_{k,H}$), standard deviation($\sigma_{k,H}$), and skewness($\alpha_{k,H}$) for H channel, and the average($E_{k,S}, E_{k,I}$)

for S and I channels, respectively.

C. Classification Using ART2

The ART neural network model was introduced by Carpenter and Grossberg [13]. It has a stability-plasticity characteristic which remains adaptive in response to a significant input but remains stable in response to an irrelevant input. Especially, ART2, a class of the ART model, rapidly self-organizes pattern categories in response to arbitrary sequences of either analog or binary input patterns [14].

For hierarchical retrieval, a query image is classified into its corresponding category by ART2, using color feature vectors of subblocks. The global search can be done on an image database, using a color category code. Then, the local search is followed, using texture feature vectors. Therefore, we can obtain search space reduction in image retrieval.

2. Texture Feature Extraction

Most existing approaches to texture feature extraction use statistical methods. For the analysis of a texture image, it requires large storage space and a lot of computation time to calculate the matrix of features such as SGLDM and NGLDM (Neighboring Gray Level Dependence Matrix) [11]. In spite of the large size of each matrix, a set of their scalar features calculated from the matrix is not efficient to represent the characteristics of image contents.

Therefore, we propose a new texture extraction method based on Discrete Cosine Transform (DCT) [15]. The block diagram of the proposed one is shown in Fig. 4.

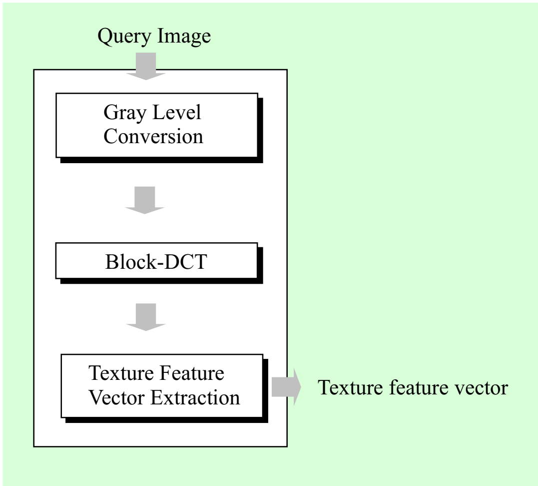

Fig. 4. The block diagram of the texture feature extraction.

When a query image is provided, its RGB version is converted into a gray level version for DCT transform. The DCT-based transform is used for considering spatial localization. The texture feature vector is obtained from some DCT coefficients. The property of our feature vector is easily computed directly from the DCT coefficients and the spatial localization using subblocks. Since we use the DCT coefficients directly from the transform domain, it does not need additional complex computation. It will then overcome some problems of existing methods such as computational complexity and storage space. Using the texture feature vectors and the measure of similarity mentioned in Section IV, we can retrieve similar images from an image database with texture feature only.

For the hierarchical retrieval, with the help of a texture feature vector, we can retrieve candidate images from only the image group with the same color category code generated by a color feature vector. The hierarchical method can reduce the search space in a large image database.

A. The Gray-Level Conversion and Block-Based DCT Transform

For the DCT transform, we convert an RGB query image into a gray level image, I component in (1). For spatial localization, we then use block-based DCT transformation. Each image is divided into $N \times N$ sized subblocks.

The two-dimensional DCT of a sequence $f(i,j)$ for $i,j = 0,1,\ldots,N-1$, can be defined as

$$\begin{aligned} F(u,v) &= \frac{4c(u)c(v)}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(i,j) \\ &\times \cos\left[\frac{(2i+1)u\pi}{2N}\right]\cdot\cos\left[\frac{(2j+1)v\pi}{2N}\right] \\ &\text{for} \quad u,v = 0,1,\ldots,N-1, \end{aligned} \tag{40}$$

where

$$c(\omega) = \begin{cases} \frac{1}{\sqrt{(2)}} & \text{for} \quad \omega = 0, \\ 1 & \text{for} \quad \omega = 1,2,\ldots,N-1. \end{cases}$$

The inverse DCT transform is given by

$$\begin{aligned} f(i,j) &= \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} c(u)c(v)F(u,v) \\ &\times \cos\left[\frac{(2i+1)u\pi}{2N}\right]\cdot\cos\left[\frac{(2j+1)v\pi}{2N}\right] \\ &\text{for} \quad i,j = 0,1,\ldots,N-1. \end{aligned} \tag{41}$$

Among the families of transforms, DCT is generally recognized as the most effective one to encode image information. It has been adopted the JPEG coding standard and the MPEG coding standards [16]-[18]. Our method uses DCT for feature extraction, and so, feature vectors may be extracted directly from a transform domain.

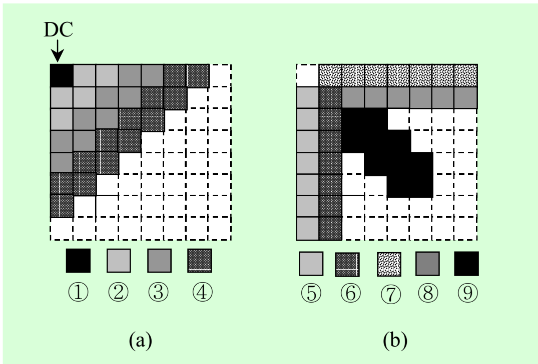

Fig. 5. DCT 9-feature vector: (a) vector elements from frequency components, (b) vector elements from directional information (each vector element is obtained by averaging same marked coefficients).

B. Texture Feature Vector Extraction

For efficient texture feature extraction, our method uses some DCT coefficients. For each subblock containing one DC coefficient and other AC coefficients, we extract a feature set of 9 vector components by averaging specific regions of coefficients as shown in Fig. 5, considering the following characteristics: i) the DC coefficient of each subblock represents the average energy of an image (vector component ①); ii) all the remaining coefficients within a subblock contain frequency information which represents a different pattern of image variation (vector components ② to ④); and iii) the coefficients of some regions within a subblock also represent some directional information (vector components ⑤ to ⑨); for example, the coefficients of the most upper region and those of the most left region in a DCT transform domain represent some vertical and horizontal edge information, respectively.

IV. EXPERIMENTAL RESULTS

The proposed methods were implemented on an IBM platform using GNU C language. Their effectiveness was evaluated by using 37 query images from a database of 200 images which were obtained by photo CD-ROM. There are three classes of images: mountains with sky, flowers, and forests. They are 256 levels color images of size 128×128. The subblock size is $32\times 32, m=n=32$.

For the similarity calculation, we use the similarity measure $D(Q,C)$. It computes the similarity between two images, $Q(x,y)$ and $C(x,y)$, as

$$D(Q,C)=\sum_{k=1}^{l}|F_{q,k}-F_{c,k}|, \qquad (42)$$

where $F_{q,k}$ is the feature vector of the k-th subblock in image Q and $F_{c,k}$ is that of image C, and l is the number of subblocks. Output images are retrieved, based on the similarity between a query image and images in the image database.

To evaluate the retrieval efficiency of the proposed methods, we use the objective measure, *Recall* and *Precision* [19] shown in (8). *Recall* is the relevant retrieval rate from all the relevant items in an image database and *Precision* represents the correct retrieval rate.

$$Recall=\frac{R_r}{T}, \quad Precision=\frac{R_r}{T_r}, \qquad (43)$$

where R_r is the number of relevant retrieved items, T is the total number of relevant items in the image database, and T_r is the number of all retrieved items.

Our proposed retrieval system supports selective or hierarchical retrieval using color, texture or one after the other.

Table 1 shows the selective retrieval results of our method using only the color feature. A comparison with other methods is also presented.

The proposed method presents about 36% and 53% performance improvement, com- pared with Stricker's method, in terms of *Re- call and Precision*, respectively. A sample query result of our method using the color feature is compared to that of Stricker in Fig. 6, which showes that the proposed method retrieved more similar images to the query image, compared to Stricker's method. Since the proposed method uses

Table 1. Comparison of the selective retrieval results using the color feature (query: 37 images, image database: 200 images).

Method / Class	Swain's method		Funt's method		Stricker's method		Proposed method	
	Recall	*Precision*	*Recall*	*Precision*	*Recall*	*Precision*	*Recall*	*Precision*
class 1	0.35	0.30	0.48	0.36	0.72	0.62	0.80	0.65
class 2	0.41	0.23	0.35	0.11	0.37	0.36	0.76	0.69
class 3	0.68	0.78	0.78	0.48	0.66	0.38	0.81	0.72
average	0.48	0.44	0.54	0.32	0.58	0.45	0.79	0.69

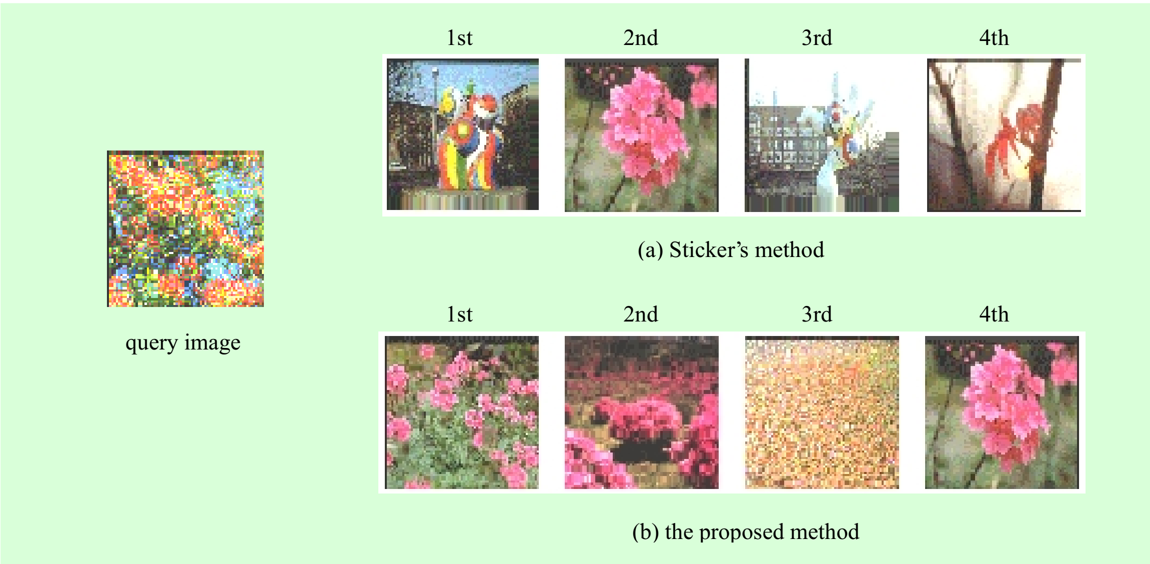

Fig. 6. A sample query result of Stricker's method and the proposed method using the color feature.

spatial location information, it yields better performance.

In Table 2, the selective retrieval results of our method using the texture feature only, are shown in comparison with Tamura's method [20], which is a typical texture feature extraction method using the properties of a texture, contrast, directionality, and coarseness.

The proposed method yields about 67% and 188% performance improvement respectively. A sample query result of our method using the texture feature, is compared with that of Tamura in Fig. 7, which shows that the proposed method can retrieve images that are more similar to the query image.

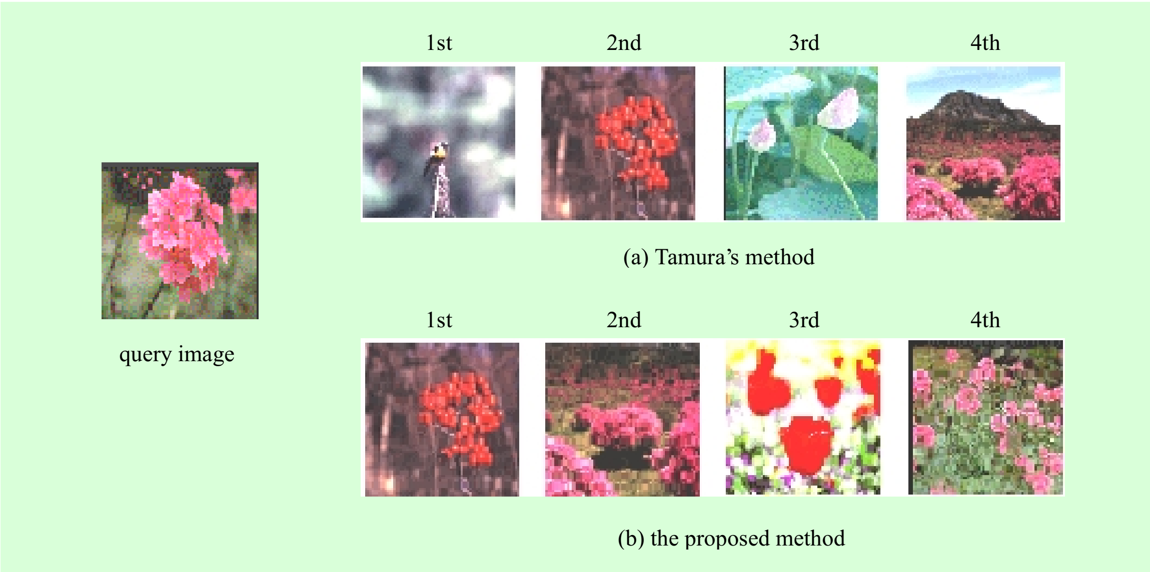

Fig. 7. A sample query result of Tamura's method and the proposed method using the texture feature.

Table 2. Comparison of the selective retrieval results using the texture feature (query: 37 images, image database: 200 images).

Method / Class	Tamura' s method		Proposed method (DCT)	
	Recall	*Precision*	*Recall*	*Precision*
Class 1	0.51	0.19	0.83	0.66
Class 2	0.37	0.34	0.75	0.79
Class 3	0.56	0.25	0.81	0.81
average	0.48	0.26	0.80	0.75

Table 3. Hierarchical retrieval results using the color and texture feature.

Measure / Class	exact matching rate	average matching rate	*Recall*	*Precision*
Class 1	0.89	0.91	1.00	0.96
Class 2	0.54	0.90	0.80	0.82
Class 3	0.89	0.93	0.87	0.88
average	0.77	0.91	0.89	0.89

In Table 3, the results of the hierarchical retrieval are presented using the color and texture feature hierarchically as shown in Fig. 2. In the table, the exact matching rate considers the retrieval of the first candidate only, and the average matching rate considers the first candidate and other candidates together.

The high average matching rate means that we can easily identify a target image among retrieved images. By the proposed hierarchical retrieval method using the color and texture together, we obtained about 91% average matching rate. We also conclude that the hierarchical retrieval in Table 3 shows better performance than the selective retrieval using one feature in Table 1 and Table 2, respectively.

V. CONCLUSION

Efficient content-based image retrieval methods using a color and texture feature, were presented. We proposed two new feature extraction methods. The first one is an advanced color feature extraction algorithm considering spatial location information. The second one is a texture feature extraction algorithm using some DCT coefficients which require no additional complex computation for feature extraction. A hierarchical retrieval and a selective retrieval is possible by applying the two features one after the other or only just one of the two.

In the experiments using an image database of 200 natural images, the proposed methods show higher performance than other methods. Hierarchical retrieval can offer a reduction of the search space in a large image database. Furthermore, the proposed texture feature extraction method allows for an easy computation of the feature vectors because they can be calculated directly from DCT coefficients in a transform domain. Therefore, the proposed method overcomes some problems of the existing methods, such as storage space problem and computation complexity.

For future work, we will consider the improvement of our retrieval rate through the efficient selection of features and the adoption of other features. Furthermore, we would like to compare our methods with other available methods using a larger image database.

REFERENCES

[1] *ISO/IEC JTC1/SC29/WG11 N1920, MPEG-7: Context and Objectives*, October 1997.

[2] Atsushi Ono, Masashi Amano, Mituhiro Hakaridani, Takashi Satou, and Masao Sakauchi, "A Flexible Content-Based Image Retrieval System with Combined Scene Description Keyword," *Proceedings of Multimedia '96*, IEEE, 1996, pp. 201–208.

[3] Tat-Seng Chua, Swee-Kiew Lim, and Hung-Keng Pung, "Content-Based Retrieval of Segmented Images," *Association for Computing Machinery (ACM) Multimedia '94*, 1994, pp. 211–218.

[4] Kozaburo Hachimura, "Retrieval of Paintings Using Principal Color Information," *Proceedings of ICPR '96 IEEE*, 1996, pp. 114–119.

[5] E.M. Rasmussen and S.A. McLean, "Image Query and Retrieval: A Case Study," *Proceedings of International Symposium on Digital Libraries 1995*, University of Library and Information Science, Tsukuba, Japan, 1995, pp. 5–9.

[6] A. Del Bimbo and P. Pala, "Image Indexing Using Shape-Based Visual Features," *Proceedings of ICPR '96 IEEE*, Vol. 3, 1996, pp. 351–355.

[7] Rajiv Mehrotra and James E. Gray, "Feature-Based Retrieval of Similar Shapes," *9th International Conference on Data Engineering*, 1993, pp. 108–115.

[8] M.J. Swain and D.H. Ballard, "Color Indexing," *International Journal of Computer Vision*. Vol. 7, No. 1, 1991, pp. 11–32.

[9] Greg Pass, Ramin Zabih, and Justin Miller, "Comparing Images Using Color Coherence Vectors," *Association for Computing Machinery (ACM) Multimedia '96*, 1996, pp. 65–73.

[10] M. Stricker and M. Orengo, “Similarity of Color Images,” *Proceedings of The International Society for Optical Engineering (SPIE)*, Vol. 2220, No. 1, 1995, pp. 381−392.

[11] Borko Furht, *Video and Image Processing in Multimedia Systems*, Kluwer Academic Publishers, 1995, pp. 225−249.

[12] B. Funt and G. Finlayson,“Color Constant Color Indexing,” *IEEE Trans. Pattern Ananysis and Machine Intelligence*, Vol. 17, No. 5, May 1995, pp. 522−529.

[13] Gail A. Carpenter and Stephen Grossberg, “ART 2: Self-Organization of Stable Category Recognition Code for Analog Input Pattern,” *Optics*, Vol. 26, No. 23, 1987, pp. 4919−4930.

[14] J.A. Freeman and D.M. Skapura, *Neural Networks Algorithms, Applications, and Programming Techniques*, Addison-Wesley Publishing Company, New York, 1991, pp. 291−339.

[15] Hee-Jung Bae and Sung-Hwan Jung, “Image Retrieval Using Texture Based on DCT,” *Proceedings of ICICS ’97*, Singapore, Vol. 2, 1997, pp. 1065−1068.

[16] G.K. Wallace, “Overview of the JPEG Still Image Compression Standard,” *SPIE*, Vol. 1244, 1990, pp. 220−233.

[17] D.J. Le Gall, “The MPEG Video Compression Algorithm: A Review,” *SPIE*, Vol. 1452, 1991, pp. 444−457.

[18] Sung-Hwan Jung, S.K. Mitra, and D. Mukherjee, “Subband DCT: Definition, Analysis, and Applications,” *IEEE Trans. on Circuit and Systems for Video Tech.*, Vol. 6, No. 3, June 1996, pp. 273−286.

[19] G. Salton and C. Buckley, *Improving Retrieval Performance by Relevance Feedback,* the Research Report of Cornell University, Feb. 1988, pp. 1−24.

[20] H. Tamura, S. Mori, and T. Yamawaki, “Textures Corresponding to Visual Perception,” *IEEE Trans. Syst. Man Cybern*, SMC-8(6), 1978, pp. 460−473.

Sang-Mi Lee received the B.S., M.S., and Ph.D. degrees from Kyungpook National University, Korea, in 1983,1985, and 1990, respectively. She has worked for ETRI since 1991 as a Senior Member of Technical Staff. Her interests are video coding, segmentation, computer human interfaces and multimedia.

Hee-Jung Bae received the B.S. and M.S. degrees from Changwon National University, Korea in 1992 and 1997, respectively. In 1997, she joined the Department of Computer Engineering at Kyungpook National University as a student of Ph.D. course. Her current research interests include parallel algorithm and multimedia database.

Sung-Hwan Jung received the B.S., M.S., and Ph.D. degrees from Kyungpook National University, Korea, in 1979,1983, and 1988, respectively. From 1983 to 1985 he had worked as a Member of Research Staff at ETRI. In 1988, he joined the Department of Computer Science at Changwon National University as an Associate Professor. From 1992 to 1994, He was a Post-Doc in the Department of Electrical and Computer Engineering, University of California, Santa Barbara. His current research interests include image processing, computer vision, multimedia database, and neural network applications.