Comparison Between Color and Texture Features for Image Retrieval

Asadollah Shahbahrami[1, 2] Demid Borodin[1] Ben Juurlink[1]
[1]Computer Engineering Laboratory
Faculty of Electrical Engineering, Mathematics, and Computer Science
Delft University of Technology, The Netherlands
Phone: +31 15 2787362. Fax: +31 15 2784898.
E-mail: {A.Shahbahrami,D.Borodin,B.H.H.Juurlink}@TUDelft.nl
[2]Department of Computer Engineering, Faculty of Engineering, University of Guilan, Rasht, Iran.

Abstract—

Content-Based Image Retrieval (CBIR) systems are used in order to automatically index, search, retrieve, and browse image databases. Color and texture features are important properties in content-based image retrieval systems. We use color histogram, color moments, and Color Coherence Vector (CCV) features as color descriptors and co-occurrence matrices and discrete wavelet transform features as texture descriptors in this paper. Both color and texture features of images are extracted and stored as feature vectors in a database. During the retrieval process, the color and texture feature vector of the query image is computed and matched against those features in the database. We use sum-of-squared-differences for similarity measurement. The goal of this paper is to determine which color or texture features are the most efficient to represent similarity of color images. Our initial results show that the color moment and color histogram descriptors are not effective features because they do not consider spatial information of image pixels. Therefore, different images may have similar color distributions. In addition, our results show that the CCV and co-occurrence matrix features retrieve much more relevant images than other color and texture features. Additionally, in order to increase precision, the combination of color and texture features should be used in CBIR systems.

Keywords: Feature Extraction, Image Retrieval, Similarity Measurements.

I. Introduction

There exist two approaches to search, to browse, and to retrieve images. The first one is based on textual information attributed to the images manually by a human. This is called *concept-based* or *text-based* image indexing [3]. A human describes the images according to the image content, the caption, or the background information. However, the representation of an image with text requires significant effort and can be expensive, tedious, time consuming, subjective, incomplete, and inconsistent. To overcome the limitations of the text-based approach, the second approach, *Content-Based Image Retrieval* (*CBIR*) techniques are used [3, 4]. In a CBIR system, images are automatically indexed by summarizing their visual features such as color, texture, and shape. These features are automatically extracted from the images.

A key function in the CBIR system is feature extraction. A feature is a characteristic that can capture a certain visual property of an image either globally for the whole image, or locally for objects or regions. Some key issues related to CBIR systems are the following. First, how the extracted features can present image contents. Second, how to determine the similarity between images based on their extracted features. One technique for these issues is using a vector model. This model represents an image as a vector of features and the difference between two images is measured via the distance between their feature vectors.

In this paper, we represent images by different descriptors and we compare their effectiveness and efficiency in a CBIR system. We focus on color images in the RGB color space model. Color and texture features [7], which are important properties, are extracted from images. We use the following color descriptors: color histogram, color moments, and Color Coherence Vector (CCV). Texture features are extracted from co-occurrence matrices and wavelet transform coefficients. The features are extracted and stored as feature vectors. During the retrieval process, the feature vector of the query image is computed and matched against those features in the database. We use sum-of-squared-differences for similarity measurements. The goal of this paper is to determine which color or texture features are the most efficient to rep-

This research was supported in part by the Netherlands Organization for Scientific Research (NWO) and by the European Commission in the context of the SARC integrated project #27648 (FP6).

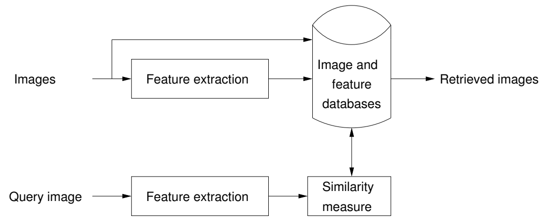

Fig. 1. A typical block diagram for a content-based image retrieval system.

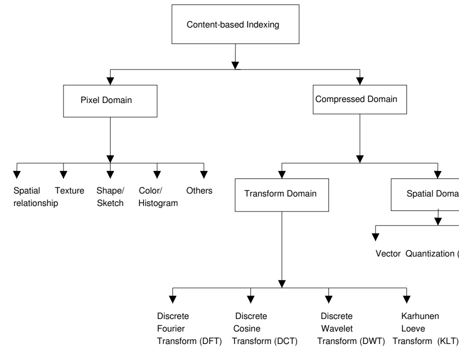

Fig. 2. Various methods in content-based image indexing.

resent similarity of color images.

This paper is organized as follows. The background information is discussed in Section II. The color and texture features are explained in Section III and Section IV, respectively. Section V describes the similarity measurement functions. Section VI presents the experimental results and finally, conclusions and future work are given in Section VII.

II. Bakground

In CBIR systems, images are indexed by some extracted features. Image pixels are mapped into a reduced representation set of features. Mapping the image pixels into the feature space is called features extraction. Features try to represent relevant information from the image pixels in order to perform the required functions such as searching, indexing, and browsing using this reduced representation instead of the full image size. Figure 1 depicts a typical CBIR system.

A feature is a characteristic that can capture a certain visual property of an image either globally for the whole image, or locally for objects or regions [5]. CBIR systems use features for searching, indexing, and browsing in a database of images. This is because represinting image from pixel space to feature space is more efficient in terms of storage and computation. An image is represented as a vector of features and dimension of vector features is less than the image dimension. In addition, the CBIR systems use different similarity measurements to determine the similarity between images based on their feature vectors.

In retrieval stage of a CBIR system, features of the given query image is also extracted. After that the similarity between the features of the query image and the stored feature vectors is determined. That means that computing the similarity between two images can be transformed into the problem of computing the similarity between two feature vectors [8]. This similarity measure is used to give a distance between the query image and a candidate match from the feature data database. It is a feature match function. The final images are retrieved based on some ranking such as minimum distance measurements. Consequently, the large computational cost associated with CBIVR systems is related to matching algorithms for feature vectors, because there are many feature vectors from different images in the feature database.

Typically, visual indexing techniques can be divided into two group, namely, pixel domain and compressed domain techniques, as depicted in Figure 2. The pixel domain indexing is based on features such as color/histogram, texture, shape, etc. Compressed domain indexing techniques can be generally classified into transform domain and spatial domain techniques. The transform domain techniques are generally based on discrete Fourier transform, Discrete Cosine Transform (DCT), Karhunen-Loeve transform, and Discrete Wavelet Transform (DWT). Spatial domain techniques include vector quantization (VQ) and fractals [9].

In this paper we have considered both color and texture features, which are extracted in both pixel and transform domains.

III. Color Features

Color is an important feature for image representation which is widely used in image retrieval. This is due to the fact that color is invariance with respect to image scaling, translation, and rotation. The key items in color feature extraction consist of color space, color quantization, and the kind of similarity measurements. In this paper three color descriptors, namely, color moment, color histogram, and Color Coherence Vector (CCV) are used.

A. Color Moment and Color Histogram

Mean, variance, and standard deviation are defined for an image of size $N \times M$ in the following equations, respectively.

$$\bar{x} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} x_{ij}}{NM}. \quad (1)$$

$$\delta^2 = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}(x_{ij} - \bar{x})^2. \quad (2)$$

$$\delta = \sqrt{\frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}(x_{ij} - \bar{x})^2}. \quad (3)$$

where x_{ij} is the value of the pixel in row i and column j.

A histogram is the distribution of the number of pixels for an image. The color histogram represents the color content of an image [1]. It is robust to translation and rotation [14]. Color histogram is a global property of an image. The number of elements in a histogram depends on the number of bits in each pixel in an image. For example, if we suppose a pixel depth of n bit, the pixel values will be between 0 and $2^n - 1$, and the histogram will have 2^n elements.

The color histogram can be used to efficiently calculate the mean and standard deviation of very large data sets. This is especially important for images, which can contain millions of pixels. The sum of all elements in the histogram must be equal to the number of pixels in the image.

$$NumberofPixles = \sum_{i=0}^{255} h[i]. \quad (4)$$

where h is the histogram of the image. Therefore, the mean and standard deviation are calculated using the color histogram by the following equations.

$$\bar{x} = \frac{\sum_{i=0}^{255} i * h[i]}{NumberofPixles}. \quad (5)$$

$$\delta = \sqrt{\frac{1}{NumberofPixles}\sum_{i=0}^{255} h[i] * (i - \bar{x})^2}. \quad (6)$$

Color histogram does not consider the spatial information of pixels. Therefore, different images may have similar color distributions. In order to avoid this, we have used color coherence vector feature.

B. Color Coherence Vector

In Color Coherence Vector (CCV) approach, each histogram bin is partitioned into two types, coherent and incoherent. If the pixel value belongs to a large uniformly-colored region then is referred to coherent otherwise it is called incoherent [10]. In other words, coherent pixels are a part of a contiguous region in an image, while incoherent pixels are not. A color coherence vector represents this classification for each color in the image.

IV. Texture Features

Texture refers to visual patterns with properties of homogeneity that do not result from the presence of only a single color such as clouds and water [11]. Texture features typically consist of contrast, uniformity, coarseness, and density. There are two basic classes of texture descriptors, namely, statistical model-based and transform-based. The former one explores the grey-level spatial dependence of textures and then extracts some statistical features as texture representation. One example of this group is co-occurrence matrix representation. The latter approach is based on some transform such as DWT. These texture features are discussed in the following section.

A. Gray-Level Co-occurrence Matrix

Gray-level co-occurrence approach uses Gray-Level Co-occurrence Matrices (GLCM) whose elements are the relative frequencies of occurrence of grey level combinations among pairs of image pixels. The GLCM can consider the relationship of image pixels in different directions such as horizontal, vertical, diagonal, and antidiagonal. The co-occurrence matrix includes second-order grey-level information, which is mostly related to human perception and the discrimination of textures [6]. Four statistical features of the GLCMs are computed. The features are energy, entropy, contrast, and homogeneity. $G \times G$ GLCM P_d for a displacement vector $d = (dx, dy)$ is defined as follows. The (i, j) of P_d is the number of occurrences of the pair of gray-level i and j which are a distance d apart. A number of texture features are listed as follows.

$$Energy = \sum_{i=1}^{N}\sum_{j=1}^{N} P_d^2(i, j). \quad (7)$$

$$Entropy = -\sum_{i=1}^{N}\sum_{j=1}^{N} P_d(i, j) log P_d(i, j). \quad (8)$$

$$Contrast = \sum_{i=1}^{N}\sum_{j=1}^{N}(i-j)^2 P_d(i,j). \quad (9)$$

$$Homogeneity = \sum_{i=1}^{N}\sum_{j=1}^{N}\frac{P_d(i,j)}{1+|i-j|}. \quad (10)$$

B. 2D Discrete Wavelet Transform

The wavelet representation of a discrete signal X consisting of N samples can be computed by convolving X with the lowpass and highpass filters and downsampling the output signal by 2, so that the two frequency bands each contains $N/2$ samples. With the correct choice of filters, this operation is reversible. This process decomposes the original image into two sub-bands: the lower and the higher band [12]. This transform can be extended to multiple dimensions by using separable filters. A 2D DWT can be performed by first performing a 1D DWT on each row (*horizontal filtering*) of the image followed by a 1D DWT on each column (*vertical filtering*).

Figure 3 illustrates the first decomposition level ($d = 1$). In this level the original image is decomposed into four sub-bands that carry the frequency information in both the horizontal and vertical directions. In order to form multiple decomposition levels, the algorithm is applied recursively to the LL sub-band. Figure 4 illustrates the second ($d = 2$) and third ($d = 3$) decomposition levels as well as the layout of the different bands.

The 2D DWT has been applied three times on all images. In other words, third decomposition level has been computed. In that level, there are 10 subbands. The mean and standard deviation of each sunband has been computed as texture features. This means that each image has 60 texture features, which have been obtained using wavelet coefficients.

V. Similarity Measurements

Among the different similarity measurements, the Euclidean distance or Sum-of-Squared-Differences (SSD) and the Sum-of-Absolute Differences (SAD) functions have been found to be the most useful [13, 15]. For example, in [18] eight similarity measurements for image retrieval have been evaluated. Based on the results presented there, in terms of retrieval effectiveness and retrieval efficiency, the SSD and SAD functions are more desirable than other functions. Additionally, the performance of four motion estimation algorithms using different distortion measures has been evaluated in [13]. The best results related to the quality of motion predicted frame have been obtained

```
float diff;
float sum = 0;
for (i = 0; i < n; i++) {
  if ((diff = Feature_Vector_Target[i] -
              Feature_Vector_Query[i]) < 0)
              diff = -diff;
  sum += diff;
}
```

Fig. 5. C code of the sum-of-absolute differences function.

```
float sum = 0;
for (i = 0; i < n; i++)
  sum += (Feature_Vector_Target[i] -
          Feature_Vector_Query[i]) *
         (Feature_Vector_Target[i] -
          Feature_Vector_Query[i]);
```

Fig. 6. C code of the sum-of-squared differences function.

using the SSD and SAD functions. Furthermore, according to [17], among all the image metrics, the Euclidean distance is the most commonly used in image recognition and computer vision.

Equations Equation (12) and Equation (12) define the SSD and SAD similarity measurements, respectively [2].

$$SSD(f_q, f_t) = \sum_{i=0}^{n-1}(f_q[i] - f_t[i])^2. \quad (11)$$

$$SAD(f_q, f_t) = \sum_{i=0}^{n-1}(|f_q[i] - f_t[i]|). \quad (12)$$

Here $f_q = f_q[0], f_q[1], ..., f_q[n]$ represents the query feature vector, $f_t = f_t[0], f_t[1], ..., f_t[n]$ denotes the target feature vector, and n is number of features in each feature vector.

Figures 5 and 6 depict the C codes of the SAD and SSD functions, respectively. These codes compute the distance between the query feature vector and a target feature vector.

The SSD is more accurate and more complex than the SAD, since the SSD function is used for measuring the similarity between feature vectors in this paper.

VI. Experimantal Results

In this section the selected benchmark and experimental evaluation are discussed.

A. Image Database

The experiment results have been implemented with a general-purpose image database including about 1000 pictures formed by ten image categories as shown in Table I [16]. Figure 7 depicts a sample of

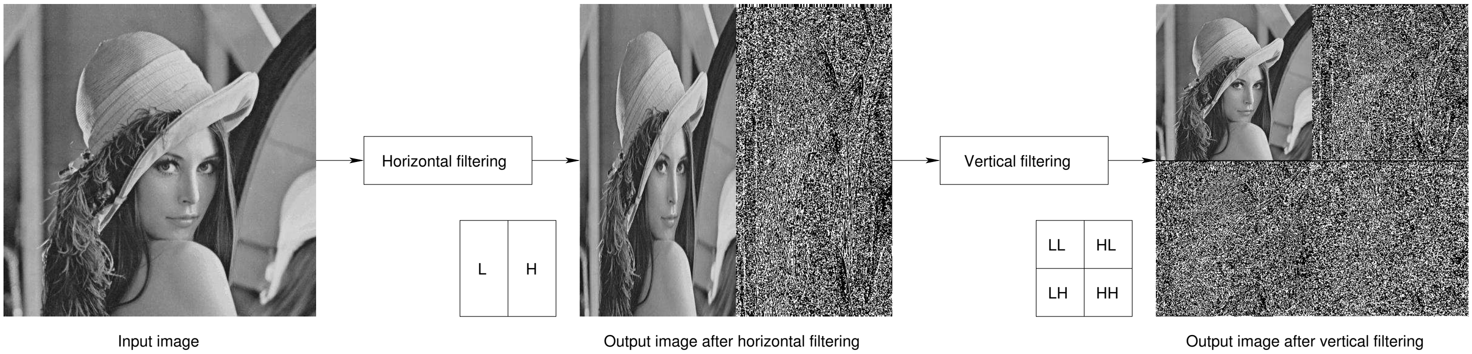

Fig. 3. Different sub-bands after first decomposition level.

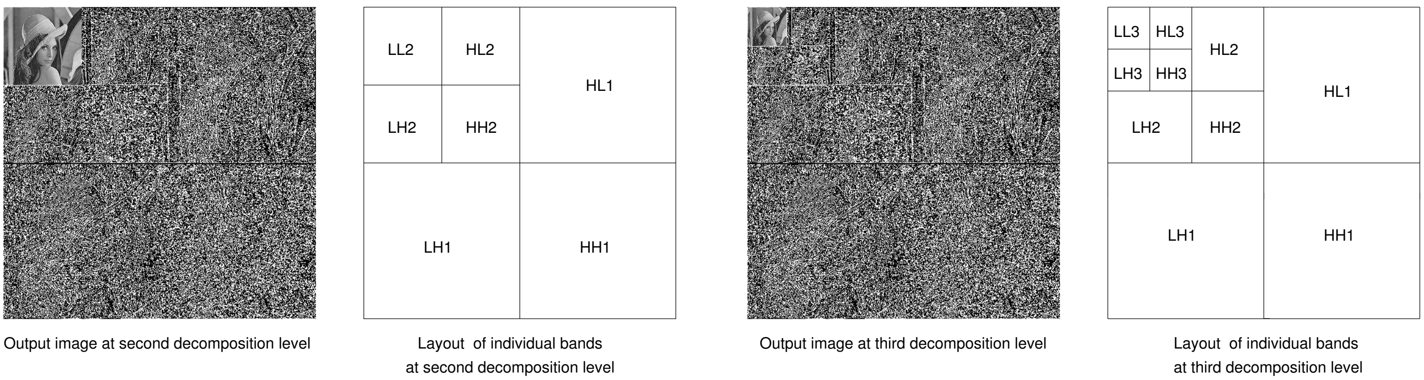

Fig. 4. Sub-bands after second and third decomposition levels.

Category number	**Category Name**
1	Africa people and villages
2	Beach
3	Historical building
4	Buses
5	Dinosaurs
6	Elephants
7	Flowers
8	Horses
9	Mountains and glaciers
10	Food

TABLE I
Different categories of image database [16].

each category. Each category depicts a distinc semantic topic. In addition, each group has 100 pictures. These pictures are stored in JPEG format with size 384×256. Each image has three component R, G, and B. For each RGB image, three different color and texture descriptors have separately been computed and stored in database.

B. Experimental Results

We have selected some sample images in order to evaluate different extracted features. In our evaluation, a retrieved image is considered a match if and only if it is in the same category as the query image. In addition, the effectiveness of the extracted features has been measured by precision and recall parameters. Precision is the ratio of relevant retrieved images to the total number of retrieved images. Recall is the ratio of retrieved relevant images to the total number of relevant images in the database.

The red bus picture depicted in Figure 7 was selected as a query image. Table II depicts different results for this query image. Color moment and color histogram features can retrieve 15 images 10 of which are bus images, while the other 5 are not. These retrieved images are shown in Figure 8. Therefore, its precision and recall are $10/15 = 0.67$ and $10/100 = 0.1$, respectively. The CCV feature retrieves 37 images. The 20 images are in the bus category, while 17 are in other category as depicted in Figure 9. Thus, the precision is $20/37 = 0.54$, and the recall is $20/100 = 0.2$. Its precision is lower than color moment and histogram features, while its recall is higher. The CCV features can retrieve more images than color

Fig. 7. One sample of each category in the image database.

histogram moment and histogram. The combination of all color features retrieve 7 pictures, all of them are in the same category as shown in Figure 10. This means that using combination of all color features is much better than using each color feature separately in terms of precision factor.

The number of retrieved images by co-occurrence matrix features is 49, which is the largest compared to other descriptors. These retrieved images are depicted in Figure 11. The 20, 11, and 29 images are buses, red buses and other images, respectively. The texture features extracted by DWT retrieve 36 images, while the combination of all texture features retrieves 9 images, 7 of them are in the bus category and 2 of them are not. Figure 12 depicts the retrieved images using DWT descriptors.

In addition, we have used all color and texture features together. In this case just one image, the same as the query image, has been retrieved.

VII. Conclusions

We have used different color and texture features in our Content-Based Image Retrieval (CBIR) system. These descriptors are low-level features and can be easily extracted in both pixel and transform domains. For color descriptors, we have used color histogram, color moments, and color coherence vector features. The co-occurrence matrices and discrete wavelet transform features have been used for texture descriptors. All extracted features have been stored in a database. While searching in the database, the color and texture feature vectors of the query images have been computed and matched against those features in the database. The sum-of-squared-differences has been used as a similarity measurement function. Our results showed that the color moment and color histogram descriptors show poorer effectiveness because they do not consider spatial relationship between image pixels. Different images may have similar color distributions. Additionally, our results showed that the CCV and co-occurrence matrix features retrieve much more relevant images than other color and texture features. Furthermore, in order to increase precision, the combination of color and texture features can be used in CBIR systems.

In general, the effective implementation of a CBIR system is difficult. This is due to the fact that the implementation depends mainly on image and machine vision algorithms. Experimental results show that the use of a single class of descriptors, either color or texture in CBIR systems is not sufficient. An approach that can improve image retrieval in terms of precision and recall is to combine multiple heterogeneous descriptors. As a future work, we try to compare our CBIR system with other existing systems.

References

[1] Y. A. Aslandogan and C. T. Yu. Techniques and Systems for Image and Video Retrieval. *IEEE Trans. on Knowledge and Data Engineering*, 11(1):56–63, January-February 1999.

[2] S. Deb. *Video Data Management and Information Retrieval*. IRM Press, 2005.

[3] P. Enser. Visual Image Retrieval: Seeking the Alliance of Concept-based and Content-based Paradigms. *Journal of Information Science*, 26(4):199–210, August 2000.

Name of descriptors	# Retrieved Images	# Buses	# Red Buses	# Other Images	Precision	Recall
Color moment and his.	15	10	8	5	0.67	0.1
Color coherence vector	37	20	20	17	0.54	0.2
Combination of all color features	7	7	7	0	1	0.07
Co-occurrence matrix	49	20	11	29	0.41	0.2
DWT	36	15	7	21	0.42	0.15
Combination of all texture features	9	7	3	2	0.78	0.07
Combination of all features	1	1	1	0	1	0.01

TABLE II

THE NUMBER OF RETRIEVED IMAGES USING DIFFERENT DESCRIPTORS FOR RED BUS QUERY PICTURE [16].

Fig. 8. Different retrieved images by color moments and histogram when the red bus was used as a query image.

[4] V. N. Gudivada and V. V. Raghavan. Modeling and Retrieving Images by Content. *Information Processing and Management*, 33(4):427–452, 1997.

[5] I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh. *Feature Extraction, Foundations and Applications*. Springer, 2006.

[6] D. K. Iakovidis, D. E. Maroulis, and D. G. Bariamis. FPGA Architecture for Fast Parallel Computation of Co-occurrence Matrices. *Microprocessors and Microsystems*, 31:160–165, 2007.

[7] L. Kotoulas and I. Andreadis. Parallel Local Histogram Comparison Hardware Architecture for Content-based Image Retrieval. *Journal of Intelligent and Robotic Systems*, 39(3):333–343, March 2004.

[8] A. J. T. Lee, R. W. Hong, and M. F. Chang. An Approach to Content-based Video Retrieval. In *Proc. IEEE Int. Conf. on Multimedia and Expo*, pages 273–276, June 2004.

[9] M. K. Mandal, F. Idris, and S. Panchanathan. A Critical Evaluation of Image and Video Indexing Techniques in the Compressed Domain. *Image and Vision Computing Journal*, 17(7):513–529, May 1997.

[10] G. Pass, R. Zabih, and J. Miller. Comparing Images Using Color Coherence Vectors. In *Proc. 4th ACM Int. Conf. on Multimedia*, pages 65–73, 1997.

[11] J. R. Smith and S. F. Chang. Automated Binary Texture Feature Sets for Image Retrieval. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 2241–2246, 1996.

[12] E. J. Stollnitz, T. D. Derose, and D. H. Salesin. *"Wavelets for Computer Graphics: Theory and Applications"*. Morgan Kaufmann, 1996.

Fig. 9. Different retrieved images by color coherence vector feature when the red bus was used as a query image.

Fig. 10. Different retrieved images by combination of all color features when the red bus was used as a query image.

[13] S. R. Subramanya, H. Patel, and I. Ersoy. Performance Evaluation of Block-Based Motion Estimation Algorithms and Distortion Measures. In *Proc. IEEE Int. Conf. on Information Technology: Coding and Computing*, pages 2–7, 2004.

[14] M. J. Swain. Interactive Indexing into Image Databases. In *Proc. SPIE: Storage Retrieval Image Video Databases*, volume 1908, pages 95–103, February 1993.

[15] S. M. Vajdic and A. R. Downing. Similarity Measures for Image Matching Architectures a Review with Classification. In *Proc. IEEE Symp. on Data Fusion*, pages 165–170, November 1996.

[16] J. Z. Wang, J. Li, and G. Wiederhold. SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):947–963, 2001.

[17] L. Wang, Y. Zhang, and J. Feng. On the Euclidean Distance of Images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(8):1334–1339, August 2005.

[18] D. Zhang and G. Lu. Evaluation of Similarity Measurement for Image Retrieval. In *Proc. IEEE Int. Conf. on Neural Networks and Signal Processing*, pages 928–931, December 2003.

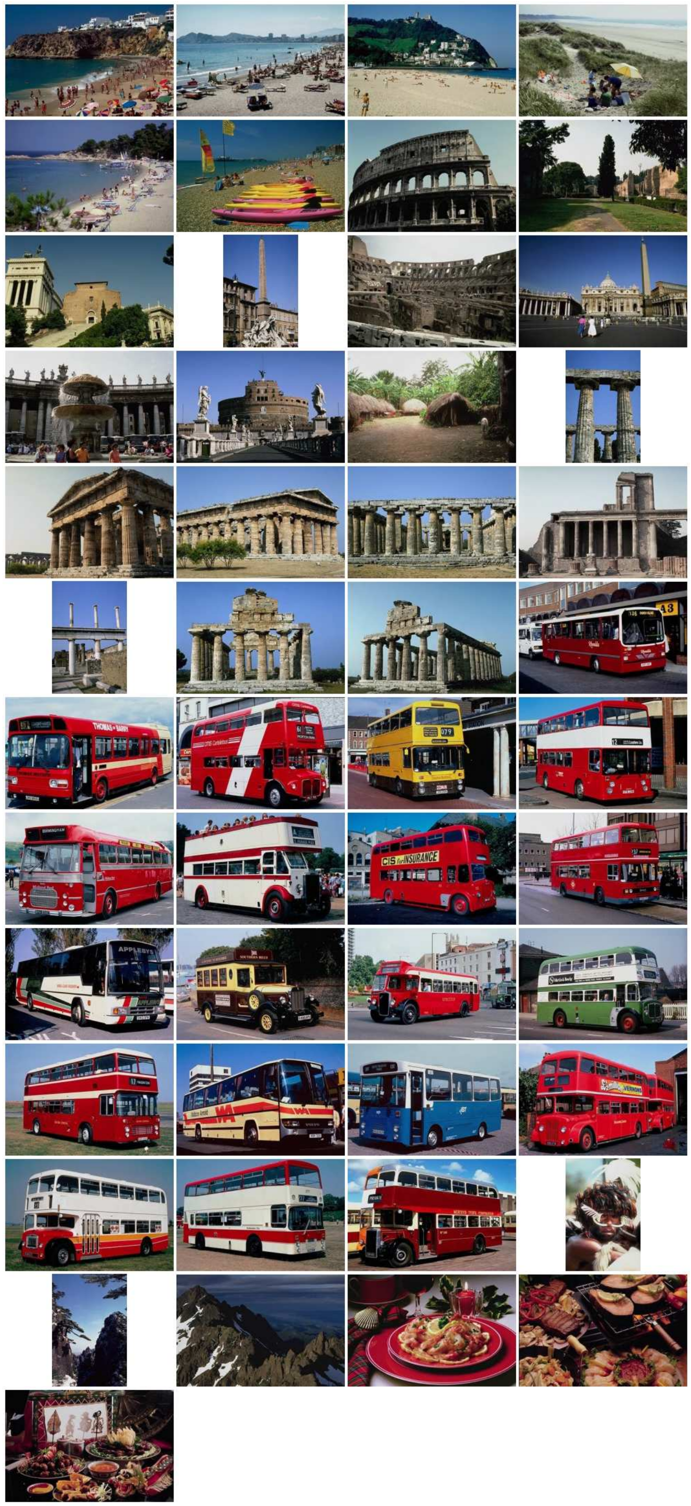

Fig. 11. Different retrieved images using co-occurrence matrix features when the red bus was used as a query image.

Fig. 12. Different retrieved images using DWT features when the red bus was used as a query image.