



Region-based image retrieval using color-size features of watershed regions

Cheng-Chieh Chiang^{a,*}, Yi-Ping Hung^b, Hsuan Yang^c, Greg C. Lee^d

^a Department of Information Technology, Takming University of Science and Technology, Taipei, Taiwan

^b Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan

^c Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan

^d Department of Computer Science and Information Engineering, National Taiwan Normal University, Taipei, Taiwan

ARTICLE INFO

Article history:

Received 13 February 2007

Accepted 13 January 2009

Available online 27 January 2009

Keywords:

Content-based image retrieval

Region-based image retrieval

Visual feature

Color-size feature

Region filtering

Earth mover's distance

ABSTRACT

This paper presents a region-based image retrieval system that provides a user interface for helping to specify the watershed regions of interest within a query image. We first propose a new type of visual features, called color-size feature, which includes color-size histogram and moments, to integrate color and region-size information of watershed regions. Next, we design a scheme of region filtering that is based on color-size histogram to fast screen out some of most irrelevant regions and images for the preprocessing of the image retrieval. Our region-based image retrieval system applies the Earth Mover's Distance in the design of the similarity measure for image ranking and matching. Finally, we present some experiments for the color-size feature, region filtering, and retrieval results that demonstrate the efficiency of our proposed system.

© 2009 Elsevier Inc. All rights reserved.

1. Introduction

Content-based image retrieval (CBIR) [3,6,7,9,20,30] has become a very active research area since the 1990's due to the rapid increase in the use of digital images. The goal of CBIR is to retrieve desired images from a large image database based on their contents. Many techniques have been proposed for yielding efficient and effective CBIR systems in the past decade. This paper focuses on region-based approach to CBIR.

Region-based image retrieval (RBIR) is a special type of CBIR by use of regions that are parts of an image with relatively homogeneous subjects or features. Regions are used to represent and index images in RBIR. The contents in an image or region are represented using extracted visual features, e.g., color [18,25,26], texture [13,17,18], and then the corresponding similarity measure is computed. Therefore, RBIR system returns images having regions that are similar to the query regions. In general, RBIR system can be categorized as two types according to the chosen query format: whole-image-as-query (WIQ) and image-region-as-query (IRQ). In a WIQ RBIR, the user provides the example image, and the system extracts feature information from the whole image for performing the query [2,15,24,30]. In an IRQ RBIR, the user performs a query by choosing regions of interest

from the example image according to their requirements [3,5,28,31].

In this paper, we design an IRQ RBIR system that focuses on three tasks: visual feature extraction, image/region representation, and image/region similarity measure. We propose a new type of visual features, called color-size feature, which embeds region-size information in color features for representing both color and texture of images. In the image/region representation, we adopt a set of watershed regions instead of a whole image. The user can specify region combinations on the subject of interest, rather than the whole image, as the query. This approach allows the retrieval system to focus more precisely on the user's requests, but using regions as the retrieval units increases the computational cost since the retrieved database contains a huge number of regions. We, therefore, design a two-phase scheme for computing the similarity measure: (1) region filtering based on color-size histogram and (2) calculating the similarity measure based on Earth Mover's Distance (EMD) [21]. The region filtering removes the most irrelevant regions/images, with the system only needing to apply the EMD-based similarity to matching of the resulting subset of candidate images.

The rest of this paper is organized as follows. Section 2 provides an overview of our RBIR system. In Section 3, we present the proposed color-size histogram and moments, and use them in region filtering in Section 4. The design of the image similarity measure, based on EMD, is presented in Section 5. Some experiments and the results that demonstrate the efficiency of our method are presented in Section 6, and finally conclusions are drawn and future work is described in Section 7.

* Corresponding author.

E-mail addresses: kevin@csie.ntnu.edu.tw (C.-C. Chiang), hung@csie.ntnu.edu.tw (Y.-P. Hung), edmond.td@gmail.com (H. Yang), leeg@csie.ntnu.edu.tw (G.C. Lee).

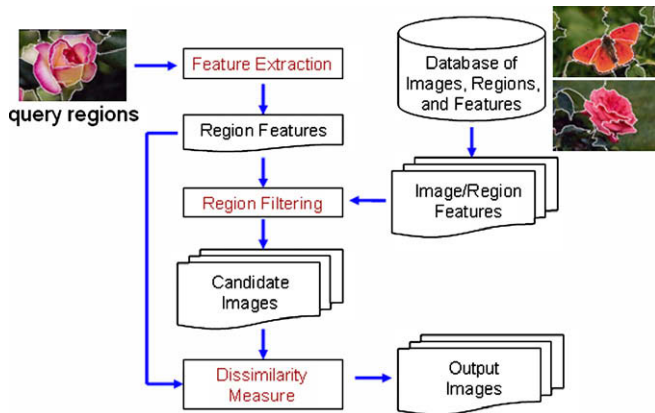


Fig. 1. The flowchart of our system.

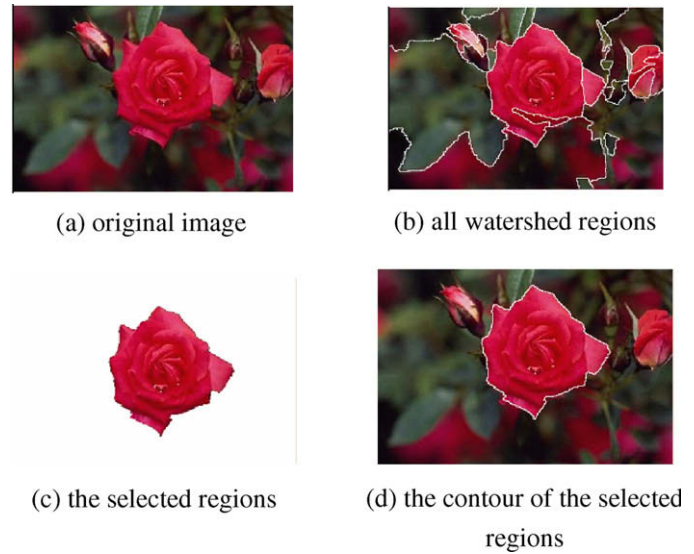


Fig. 3. Four display modes for the user specifying query.

can choose an image file and select regions of interest as the query regions.

The interface provides four display modes that aim to help the user to easily specify the query regions in the bottom-left area of Fig. 2. These four models displays the original image, all watershed regions, only user-selected regions, and the contour of user-selected regions, which are illustrated in Fig. 3(a)–(d), respectively. The user can choose the regions of interest by clicking on the image in order to best represent a collection of focused subjects using any of the four modes. This approach does not only reduce the effort of delineating the subject boundary, but also enhances the accuracy of the query. Therefore, the retrieval task can focus on the specified regions of interest. If R_i denotes a watershed region selected by the user, then a query Q , either a part of an image of or an entire image, can be expressed in general terms as

$$Q = \bigcup_i R_i. \quad (1)$$

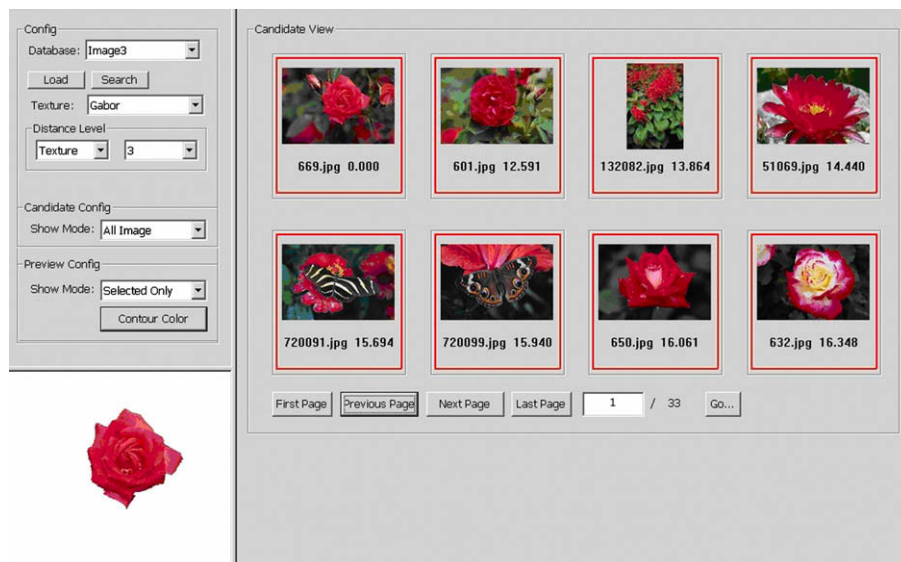


Fig. 2. The user interface of our system.

2. System overview

Fig. 1 shows the flowchart of the proposed system. All images in the database are initially segmented into regions using the watershed segmentation algorithm [27,29]. A set of features, comprising the color-size feature and Gabor texture [13,17,18], is then extracted from each watershed region. All images, regions, and features are stored in the database. Each query involves the acquisition of a set of regions of interest specified by the user, and visual features are extracted from the query regions. A set of candidate images is then produced by region filtering. In this way, the system computes the similarity measure between the query and each candidate image, which is used for image matching and ranking.

Similar to most CBIR systems, we employ “query by example” for the interface in our system, in which the user specifies an image as the query and then the system retrieves similar images from the database. Fig. 2 shows the user interface including three parts in our system. The top-left area provides the parameter configuration for the retrieval. The right area lists the retrieval results containing eight similar images on each page, where the image captions contain the file names and similarity scores of the images. The bottom-left area shows the query image. The user

3. Color-size feature

Extracting visual features for image representation is one of the fundamental tasks in image retrieval. Many kinds of visual features have been proposed for color, texture, or shape representation [4]. For example, MPEG-7 contains several color descriptors including dominant color, scalable color, color structure, and color layout [18]. These color descriptors are designed for different goals to extract the characteristics of image contents.

In this section, we present a new type of visual feature, called color-size feature, which contains the distribution information of both color and region-size. On one hand color features are widely used for characterizing image contents [4], and on the other region-size information is often used for weighting regions in image matching and ranking in region-based image retrieval [15,30,31]. Moreover, region-size information can somehow reflect the “structure” in images, which is described in Section 3.2, according to the segmentation results. Hence, we design the color-size feature by embedding region-size information into color features in this section.

3.1. Image segmentation

The goal of image segmentation is to partition an image into a set of regions. Different methods for image segmentation have been applied to region-based tasks for different goals, e.g., image retrieval, image annotation, and object recognition. The most intuitive method for image segmentation is to segment objects (or foreground subjects) from an image for region-based image matching [1,3,14,15,30], even though this is very difficult. However, the segmentation results greatly affect the performances of region-based tasks. Hence, some researchers divided an image into rectangular grids [11,19] or a large number of overlapping circular regions [12,23].

Our opinion is not to generate the best or perfect regions with segmentation, but rather to make useful ones. In this work, we use the watershed segmentation [27,29], which is an efficient, automatic, and unsupervised segmentation method for gray-level images, to partition an image into non-overlapping regions. We first convert the color images to the grey images and then partition them by watershed segmentation. Pixels in a watershed region are homogeneous in the intensity space, and hence we take the watershed regions to represent units of an image in our RBIR system. In addition, the system allows the user for specifying the compound of watershed regions within the query regions.

Because the basic watershed algorithm is highly sensitive to gradient noise, it usually results in over-segmentation. To overcome this problem, small local minima in the gradient image should be eliminated [29]. These minima are defined as local minima consisting of a small number of pixels or having low contrast with their neighbors, and are eliminated by assigned two *scaling parameters*: r and h . Parameter r is the size of the structuring element of dilation operators, whose application eliminates local minima of size less than r pixels, and parameter h is the height of elevation used to remove the local minima with low contrast. These two parameters can be used to control the coarseness of the segmentation results: as r or h increases, the number of regions generated decreases. Fig. 4 illustrates how the number of regions changes associated with different scaling parameters r and h . In the evaluation of the proposed color-size feature, we set these parameters as $r = 1$ and $h = 3$, which results in 75,000 regions for the 5000 images that are used in our retrieval experiments in Section 6.2.

3.2. Region-size feature

The region-size is defined as the number of pixels in a region, or the size percentage of a region in an image when normalized.







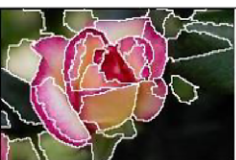


	$r=1$	$r=3$	$r=5$
$h=1$	 # of regions = 62	 # of regions = 20	 # of regions = 13
$h=3$	 # of regions = 44	 # of regions = 18	 # of regions = 12
$h=5$	 # of regions = 40	 # of regions = 17	 # of regions = 11

Fig. 4. Watershed segmentation controlled by the scaling parameters r and h .

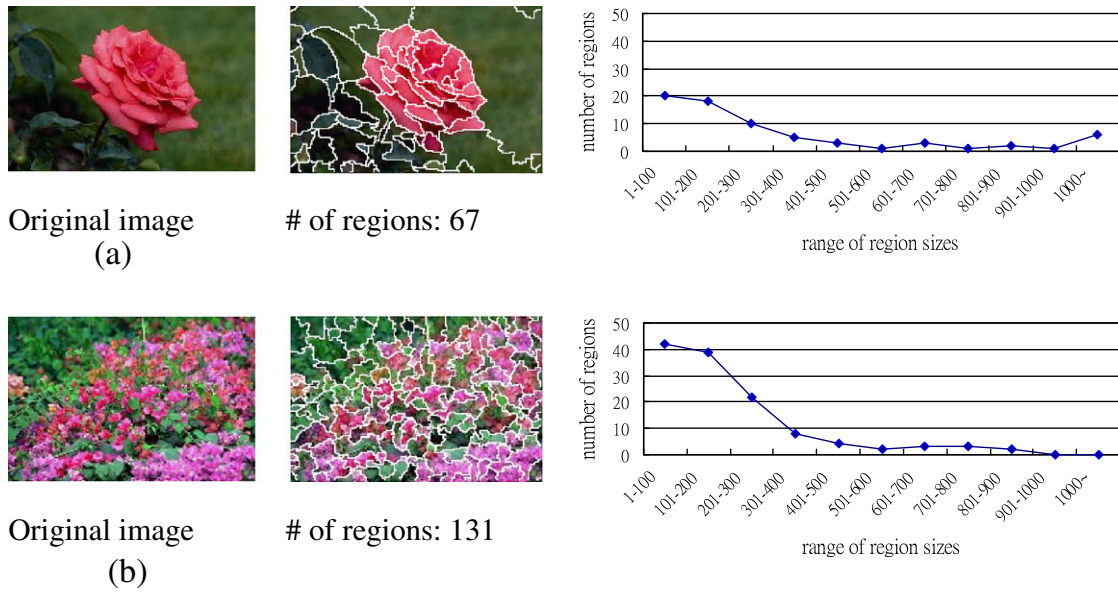


Fig. 5. Distributions of the region-size in the two examples, where both (a) and (b) contain the original image, the watershed results, and the distributions. Note that the scaling parameters of the watershed segmentation are $r = 1$ and $h = 3$.

While segmenting an image, different images can yield different numbers of regions with the same segmentation parameters; that presents some “structure” information involved in images. In addition, the region-size information is often used for weighting regions in images. These reasons motivate us to extract the region-size information, that is simple but necessary in RBIR, from regions of images and embed it in the color feature in order to improve the color feature.

Fig. 5 illustrates the “structure” information of images, that the structure of the image in Fig. 5(b) is more fragmental than that in Fig. 5(a), by demonstrating the distributions of the region-size of two images with the same size (both 192×128 pixels). The region-size distribution is based on the results of image segmentation. Images with different structures will exhibit different segmentation results with the same scaling parameters, and hence the region-size distribution contains the structural information of an image. We, therefore, do not only consider the well-known color feature but also include the region-size information to involve the “structure” of the image. We expect that considering both the region-size information and color features will yield more representative and discriminable features.

Given a pixel p in a region R of an image I , we define that the region-size attribute for p equals the percentage size of I represented by R , and hence all pixels in the same region have the same

region-size attribute. The range of the region-size attribute is from 0 to 1, and then we need to define the quantization in the extraction of the region-size histogram.

Fig. 6 shows the statistics of the region-size attribute. Here, we adopt the image data that is used in Section 6.2 to produce the region-size histogram. The dataset contains 5000 images and each image is divided by watershed segmentation with scaling parameters $r = 1$ and $h = 3$. Fig. 6(a) plots the region-size histogram of all regions, where we uniformly quantize the region-size percentage into 100 levels, i.e., 0.01 in each level. Fig. 6(a) shows that most regions are concentrated in the first two or three size levels, and the average region-size is 0.0681. Hence, using equal quantization throughout the region-size histogram is inappropriate because the region-size histogram is not uniform. We computed the cumulative distribution, which is shown in Fig. 6(b), and set the number of quantized bins in the region-size attribute, S , to four in our implementation, and then we can define the quantization boundaries of the region-size attribute at 0.001, 0.012, and 0.049 for 25%, 50%, and 75% of regions, respectively.

3.3. Color-size histogram and color-size moments

Embedding the region-size within the color feature results in each pixel of an image having four attributes: three color compo-

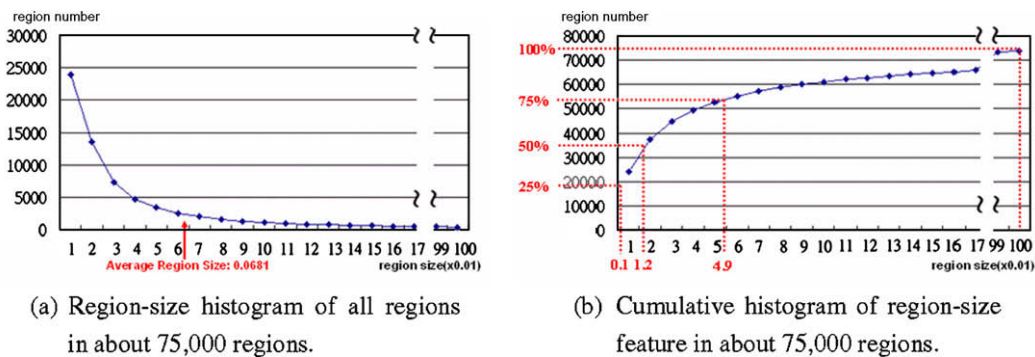


Fig. 6. The region-size histogram and the corresponding cumulative histogram.

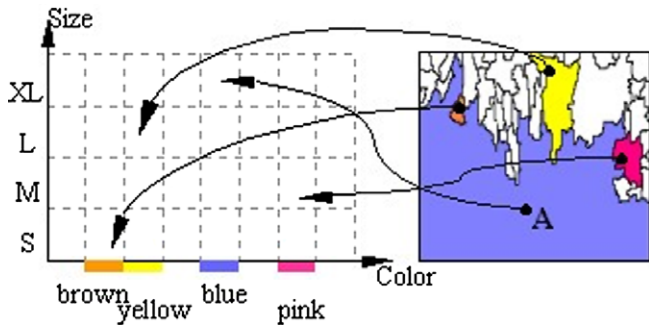


Fig. 7. Extraction of color-size histogram.

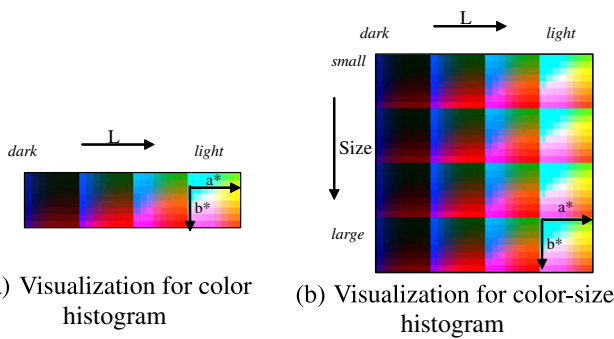


Fig. 8. The color panel for visualizing color histogram and color-size histogram, where L , $Size$, a , and b are quantized into 4, 4, 8, and 8 bins, respectively.

nents and one region-size component. Let K_1 , K_2 , and K_3 be the number of bins used to quantize the three color attributes, and K_4 be that to quantize the region-size attribute. Then, a *color-size histogram* (CSH) of an image is a $K_1 \times K_2 \times K_3 \times K_4$ -dimensional feature set,

$$CSH = \{\gamma_{ijkl} | 1 \leq i \leq K_1, 1 \leq j \leq K_2, 1 \leq k \leq K_3, 1 \leq l \leq K_4\} \quad (2)$$

where each γ_{ijkl} value in the histogram corresponds to the number of pixels having the values in color and region-size channels.

Let $p = \{p_1, p_2, p_3, p_4\}$ be the values of pixel p consisting of the three color components (p_1, p_2, p_3) and the region-size component

(p_4) , and N be the number of pixels in the image. The *color-size moments* (CSM), with first- and second-order moments, of an image are defined as:

$$CSM = \{\mu_1, \mu_2, \mu_3, \mu_4, \sigma_1, \sigma_2, \sigma_3, \sigma_4\} \quad (3)$$

where $\mu_i = \frac{1}{N} \sum_p p_i$, and $\sigma_i = \frac{1}{N} \sum_p (p_i - \mu_i)^2$, $i = 1, 2, 3, 4$.

Fig. 7 illustrates an example of extracting the color-size histogram. In this example, pixel A of the image is blue, and is contained in an extra-large (XL) region, and hence its presence will increment the bin corresponding to the color blue and region-size XL by one. Fig. 8 shows the color panel for visualizing both the color (in CIE-Lab space) and the region-size attribute, with L , $Size$, a , and b are quantized into 4, 4, 8, and 8 bins, respectively, where Fig. 8(a) is for color histogram and Fig. 8(b) is for color-size histogram. That is to say, the feature space of color histogram is divided into four ($K_4 = 4$) bins to form that of color-size histogram.

Therefore, we draw an example in Fig. 9 for showing the difference of color histogram and color-size histogram, which are based on the visualization panel of Figs. 8 and 9(a) shows two images and their corresponding color histograms, and Fig. 9(b) shows the segmentation results of the two images and their color-size histograms. In this example, the two images have similar color histograms, but they are actually different within the color-size histograms.

Note that other methods of image segmentation can be also employed to extract the region-size information for the color-size feature. Obviously, the color-size feature, either color-size histogram or moments, is rotation and shift invariant. However, the color-size feature is not scaling invariant because the region-size feature is sensitive to the segmentation. A simple solution for the scaling invariance is to tune the quantization scale K_4 of the region-size component. The bigger value K_4 , the more insensitive to the scaling change the color-size feature. Another possible method to overcome this problem is to design a multiscale representation for the color-size feature associated with the scales of image segmentation.

4. Region filtering

The goal of region filtering [15,31] is to rapidly determine whether an image may contain regions similar to the query regions and thereby speed up the computation for the retrieval process. We use the color-size histogram (in Lab color space) to build a

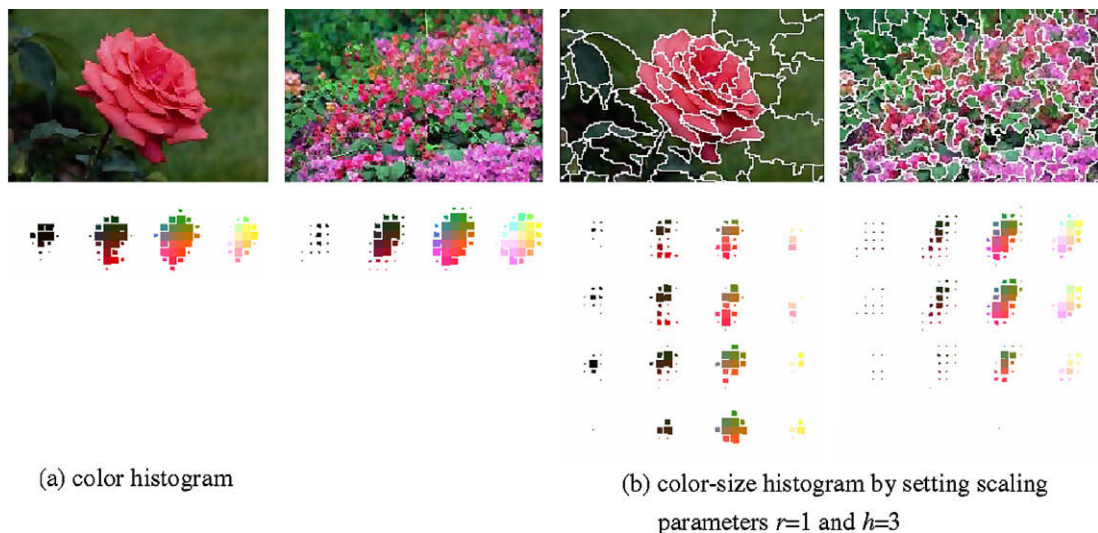


Fig. 9. An Example for color histogram and color-size histogram.

region filtering to screen out most of irrelevant regions and images. All regions of an image in the image database should be tested in the region filtering in order to decide whether the image is a candidate for image matching and ranking.

Since the color-size histogram has $K_1 \times K_2 \times K_3 \times K_4$ bins, as described in Section 3.3, the corresponding feature space can be divided into $K_1 \times K_2 \times K_3 \times K_4$ hypercubes. A straightforward method of region filtering is to mask hypercubes by use of query regions in the feature space, where regions that do not fall on masked hypercubes are filtered out. The main drawback of this method is its high sensitivity, in that regions with similar color and size may belong to neighboring hypercubes because of noises or a quantization effect. Hence, we have to loosen the constraint of region filtering to include these potential regions. The modified idea of our region filtering is based on dilation of the color-size hypercubes corresponding to the query regions. Given a color-size hypercube in which a query region falls, its neighboring hypercubes are appended to the masks of the region filtering. An illustration of hypercube dilation in three-dimensional feature space is given in Fig. 10.

Let R denote a region, and R^L , R^a , R^b , and R^S denote the quantized indices, associated with L , a , b , and region-size, respectively, of the hypercubes that contain this region in the feature space. The hypercube dilation of the region R based on color-size histogram, denoted as $D(CSH(R))$, is defined as follows:

$$\begin{aligned} D^L(CSH(R)) &= \left\{ \gamma_{ijkl} \mid \max(0, R^L - 1) \leq i \leq \min(K_1 - 1, R^L + 1), \right. \\ &\quad \left. j = R^a, k = R^b, l = R^S \right\}, \\ D^a(CSH(R)) &= \left\{ \gamma_{ijkl} \mid i = R^L, \max(0, R^a - 1) \leq j \leq \min(K_2 - 1, R^a + 1), \right. \\ &\quad \left. k = R^b, l = R^S \right\}, \\ D^b(CSH(R)) &= \left\{ \gamma_{ijkl} \mid i = R^L, j = R^a, \max(0, R^b - 1) \right. \\ &\quad \left. \leq k \leq \min(K_3 - 1, R^b + 1), l = R^S \right\}, \\ D^S(CSH(R)) &= \left\{ \gamma_{ijkl} \mid i = R^L, j = R^a, k = R^b, \max(0, R^S - 1) \right. \\ &\quad \left. \leq l \leq \min(K_4 - 1, R^S + 1) \right\} \end{aligned} \quad (4)$$

and

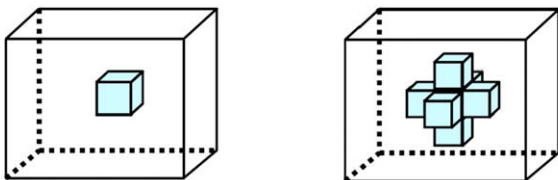
$$D(CSH(R)) = D^L(CSH(R)) \cup D^a(CSH(R)) \cup D^b(CSH(R)) \cup D^S(CSH(R)) \quad (5)$$

where $K_1 = K_4 = 4$, and $K_2 = K_3 = 8$ in our implementation.

Therefore, we define the color-size matching matrix, corresponding to the query regions Q and an image I in the database, as a matrix whose row and column dimensions are the numbers of regions in Q and I , respectively, which is defined as

$$W_{Q,I}(R_i, R'_j) = \begin{cases} 1, & \text{if } CSH(R'_j) \subset D(CSH(R_i)) \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

where R_i is the i -th region of Q , and R'_j is the j -th region of I .



(a) A single hypercube (b) The dilation of hypercubes

Fig. 10. An illustration for hypercube dilation in the three-dimensional space.

We then extract information about whether image I contains similar regions to R_i . The counting matrix for image I is a matrix whose column dimension is one and its row dimension is the number of regions of Q . It is defined as

$$P_{Q,I}(R_i) = \begin{cases} 1, & \text{if } \sum_j W_{Q,I}(R_i, R'_j) \geq 1 \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

Therefore, we define the score of the candidate image as:

$$score(Q, I) = \frac{\sum_i P_{Q,I}(R_i)}{\text{of regions in } Q} \quad (8)$$

Thus we take images with the highest scores as the candidate set for query Q . Here, we define that the threshold value $Threshold_F$ equal to the number of images in the candidate set (see Section 6.2.2).

Region filtering is implemented using inverted indexing to build links between hypercubes and region features. Let M be the total number of regions in the image set, and c be the number of hypercubes in the feature space: M is about 75,000 in our experiments, and c is 1024 in the color-size histogram. On average, M/c regions will fall in a hypercube in the feature space. In region filtering, the system does not need to consider all hypercubes in the feature space, but instead check dilated hypercubes containing these query regions. Note that the computational loading of region filtering is proportioned to the number of query regions.

5. Image matching and ranking

Applying the region filtering presented in Section 4 yields a candidate set of retrieved images from the database. We then perform image matching and ranking to all members of the candidate set, and report the results to the user. In this section, we first describe the visual features, including color and texture, for image representation, and then we describe the image similarity measure based on EMD used in this work.

5.1. Visual features and image representation

Assume that an image I comprises n regions, written as $I = \{R_1, \dots, R_n\}$. We adopt two types of visual features for region and image representation: color-size moments, for the color feature and Gabor texture [13,17,18] for the texture feature, with 8 and 48 dimensions, respectively. Note that we adopt color-size moments instead of color-size histogram for its lower dimensions.

Let $RF(R_i)$ be the features extracted from the region R_i , then

$$RF(R_i) = \{CSM(R_i), G(R_i)\}. \quad (9)$$

where $CSM(R_i)$ and $G(R_i)$ are color-size moments and Gabor texture, respectively, of region R_i . Moreover, the feature of an image I is the collection of the features of its regions:

$$IF(I) = \bigcup_i RF(R_i), \text{ where } I = \bigcup_i R_i. \quad (10)$$

5.2. Image similarity measure

Since images often comprise different numbers of regions, the common distance measures, e.g., Euclidean distance, are not very feasible for region-based image matching and ranking. Earth Mover's Distance (EMD) was first introduced by Rubner et al. [21] for color and texture images. EMD is appropriate for measuring the distance between two variable-length distributions and allowing many-to-many relationship of regions [15], hence, we adopt EMD as the kernel of the similarity measure in our RBIR system.

Given a query image Q and a candidate image I , we define the similarity measure between them as the distance of EMD between their feature sets,

$$Sim(Q, I) = -EMD(IF(Q), IF(I)), \tag{11}$$

where similar images can have small distances of EMD. EMD measures the minimal cost that must be paid to transform one distribution to another [21], where the distribution is represented by a set of signatures. In order to compute EMD, we need to define (i) the representation of a signature in an image and (ii) the ground distance between two signatures. Both of them are described as the follows.

5.2.1. Signature

The matching unit in our RBIR system is the region, so it is appropriate to define a region feature as a signature in the EMD measure. Thus the signature set of an image I can be defined as the collection of ordered pairs consisting of a region feature and the corresponding region-size:

$$Signature(I) = \cup_i \{ (RF(R_i), w_i) \} \tag{12}$$

where $I = \cup_i R_i$ and $w_i = \frac{\text{\#of pixels in region } R_i}{\text{\#of pixels in image } I}$

5.2.2. Ground distance

The ground distance between two signatures in EMD can be intuitively defined as the distance between two region features. Thus the ground distance could be a mixture of distances on individual feature spaces. We define the ground distance between two region features $RF(R_i)$ and $RF(R_j)$ as:

$$d_{ij} = \sqrt{\lambda_C (d_{ij}^C)^2 + \lambda_G (d_{ij}^G)^2}, \tag{13}$$

Table 1

The illustration of the two datasets, containing the semantic names and the numbers of images in the categories, used in the experiments.



Table 2

The recognition rates (%) using color-size moments with changing r and k .

	$r = 1$	$r = 3$	$r = 5$
(a) Using dataset W10			
$k = 1$	0.831	0.783	0.767
$k = 5$	0.837	0.796	0.782
$k = 9$	0.836	0.801	0.782
(b) Using dataset CT10			
$k = 1$	0.67	0.632	0.629
$k = 5$	0.689	0.639	0.633
$k = 9$	0.688	0.642	0.635

where d_{ij}^C and d_{ij}^G are the L_2 -distance of the color-size moments and Gabor texture, respectively, between $RF(R_i)$ and $RF(R_j)$, and both λ_C and λ_G are weights for which $\lambda_C + \lambda_G = 1$. Note that the user can set the two weighting parameters by specifying the field “distance level” in the configuration of the user interface of the Fig. 2. For the example shown in Fig. 2, that the field “distance level” is set 3 for texture means λ_G is set 3/10 and λ_C is the other 7/10.

6. Experimental results

Our experiments contain two parts: (i) evaluation for the efficiency of color-size histogram and moments and (ii) evaluation for our RBIR system.

6.1. Evaluation for color-size feature

6.1.1. Dataset

Two public sets of image are adapted in the evaluation for color-size features. We took “Wang 1000” [16] for the first image set, denoted “W10” in this paper, that contains 10 categories from Corel

Photos, with each category consisting of 100 photo images, giving a total of 1000 images in the dataset. CalTech 101-object [10] is the second image set used in our experiments. The original dataset contains 101 object categories with a total of over 8000 images. In order to compare with dataset Wang10, we chose 10 categories that contain similar numbers of images, denoted “CT10” in this paper, with about 800 images totally. Table 1(a) and (b) illustrate each category in W10 and CT10, respectively, which indicates the diverse contents of the two datasets. We also present the category names and the numbers of images of each category in Table 2.

Table 3
The recognition rates (%) using color-size histogram with different dimensions d , where $r = 1$, $h = 3$, and $k = 5$.

	$d = 5$	$d = 20$	$d = 50$	$d = 100$	$d = 1024$
W10	0.791	0.825	0.81	0.799	0.802
CT10	0.623	0.654	0.641	0.629	0.639

Table 4
The recognition rates (%) using Size: region-size feature, CH, color histogram; CSH, color-size histogram; CM, color moments; and CSM, color-size moments; where $r = 1$, $h = 3$ (for Size, CH, CSM), and $d = 20$ (for CH and CSH).

	Size	CH	CSH	CM	CSM
<i>(a) Using dataset W10</i>					
$k = 1$	0.472	0.763	0.82	0.775	0.831
$k = 5$	0.51	0.771	0.825	0.791	0.837
$k = 9$	0.507	0.779	0.825	0.8	0.836
<i>(a) Using dataset CT10</i>					
$k = 1$	0.43	0.607	0.639	0.627	0.67
$k = 5$	0.431	0.62	0.654	0.642	0.689
$k = 9$	0.43	0.621	0.653	0.645	0.688

6.1.2. Results and discussion

To precisely evaluate the efficiency of using the color-size feature, we need to design a compact experiment to avoid the influence of other factors. This is achieved by performing image classification (instead of image retrieval) using k -NN, which is simply performed with the leave-one-out strategy [8]. The experiments for color-size histogram and moments are based on the two image sets W10 and CT10.

We first consider the influence of color-size features by changing the scaling parameter r and h in watershed segmentation and the value k in k -NN. Fig. 4 shows the different numbers of regions with different scaling parameters, and it presents that r is more important than h in controlling the number of regions. Hence, in the subsequent experiments, we fix h at 3 and only change r and k for simplifying the comparison in the experiments. Table 2 lists the recognition rates using color-size moments with different values of r and k , and that shows the recognition rates are stable with different k . Hence, we set k fixed at 5 in our experiments. Also, the results of Table 2 show that the accuracies are worse while the parameter r increases ($r = 5$ is the worst). The main reason is that the additional region-size attributes are not discriminative in image classification if most regions are larger in segmentation. Hence, we fixed r at 1 in the rests of the experiments.

Another important issue is to verify the influence of the dimension for color-size histogram. The original dimension of color-size histogram is 1024 in our design, and we reduced it to 5, 20, 50, and 100 by PCA (principal component analysis) [8]. Table 3 lists the accuracies with different dimensions, denoted d , on the two datasets. We do not need to employ a large dimension of color-size histogram in classification because principal components of the features have been preserved in the first several dimensions.

Table 4 lists the recognition accuracies using five types of features with different values of k in the k -NN classifier. This table indicates that the recognition rate is bad when only using the region-size feature. But using the color-size features is better than

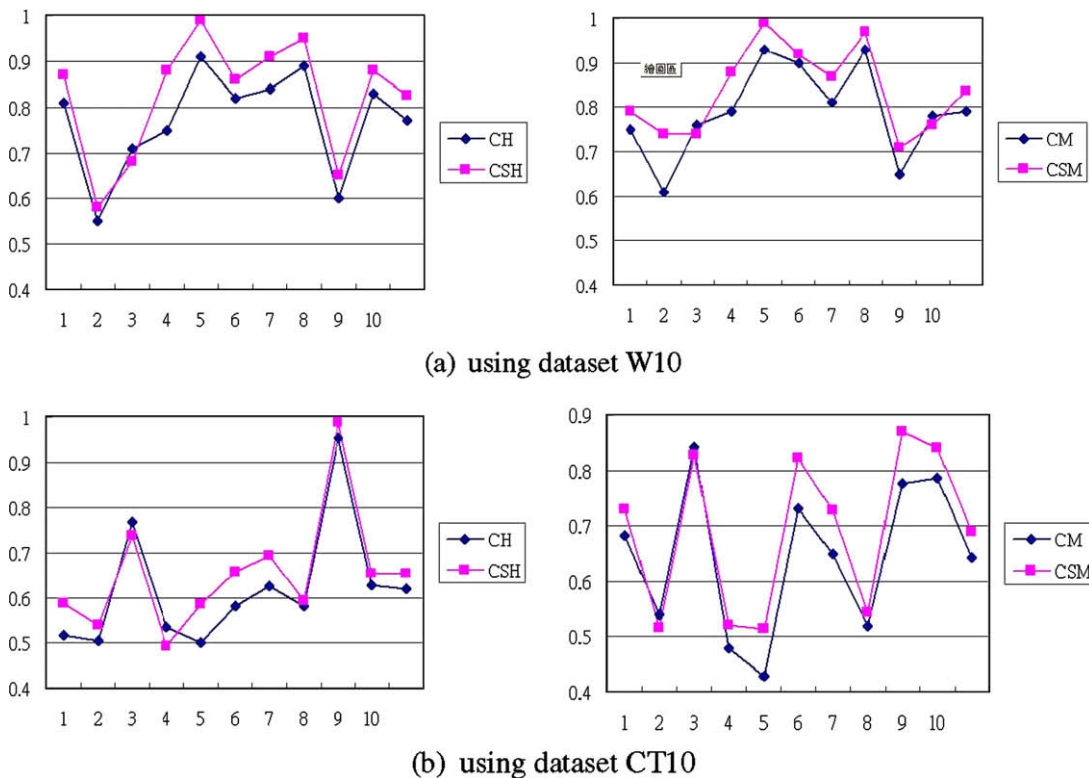


Fig. 11. The recognition rate of each category using different features with $r = 1$, $h = 3$, $k = 5$, and $d = 20$.

using the color feature, when comparing either color-size histogram vs. color histogram or color-size moments vs. color moments. Note that recognition rates using dataset W10 are between 84.1% and 37.5% described in [19], and our tests by trying different parameters achieve error rates between 76.7% and 83.6%. Moreover, Fig. 11 shows the detailed classification rates of categories with $k = 5$, which indicates that the color-size feature is better than the color feature in most cases.

6.2. Evaluation for region filtering and image retrieval

6.2.1. Dataset

For the retrieval task, we extend the dataset W10 used in the previous experiments of Section 6.1 to 50 categories by arbitrarily choosing extra 40 categories from Corel Photos and list their IDs and semantic names in Table 5. Each of these 50 categories contains 100 images, giving a dataset with 5000 images. Two images are viewed as relevant in our experiments if they are in the same category. The data categories are classified according to human concepts such as “Buses” or “Elephants”. Hence, images in the same category have very variable contents. We random select 20 images as the query, where each query is set as an entire image because we need to automatically perform the experiment on the large query set: $20 \times 50 = 1000$ query images.

6.2.2. Results and discussion for region filtering

The main goal of region filtering is to screen out as many of irrelevant images as possible, so we use the recall of candidate images to measure the performance of the proposed filtering method. There is a threshold $Threshold_F$, also described in Section 4, in region filtering, which corresponds to the number of candidate images available for a query. Obviously, the recall increases with the number of candidate images. We test the recall values for four numbers of candidate images: 500, 1000, 1500, and 2000. The average recalls of 50 categories are plotted in Fig. 12(a).

Fig. 12(a) indicates that the recall is about 45% in 1000 of the 5000 images (filtering out 80% of the dataset, and denoted as 1000/5000 for simplicity), and about 60% in 1500/5000 (filtering out 70% of the dataset). We show the best-10 and the worst-five categories in Table 6. The scheme of region filtering is shown to be stable since most categories are duplicated in different numbers of candidate images in the top-10 ranks. Most of these categories with better recalls contain obvious objects in images, e.g., “Bus”, “Cuisine”, and “Doors of Paris”. On the other hand, most categories with bottom-five ranks are also duplicated because of non-object involved in images (e.g., “Cloud” and “Waves”) or the foreground camouflages in the background (e.g., “Weasels and Hares” and “Reptilia”). Because we use the whole image instead of a part of image as the query for the automatic evaluation, the background in images hugely influences the results.

6.2.3. Results and discussion for image retrieval

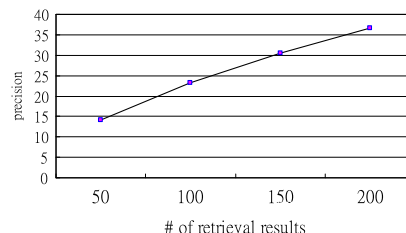
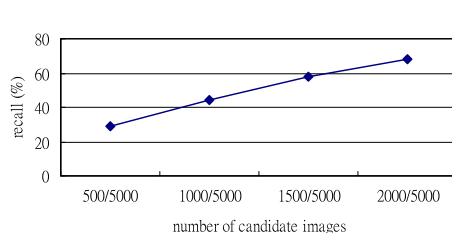
We first illustrate two experimental examples in this section. Fig. 13 shows the query, “rose”, and its retrieval results. In this example, the rose can be retrieved well because most regions in the rose are discriminated in the color-size histogram to other red objects. Fig. 14 shows another type of experimental example for our system. In this example, the query subject is the white horse, but the results are not as good when query regions only depict the horse. This is due to white being widely represented in images, which consequently results in the region filtering not judging correctly. The retrieval results are better if the query regions contain some background. Combining the backgrounds with different contents of query subjects is helpful in eliminating irrelevant regions.

The two examples presented in Figs. 13 and 14 indicate that it is difficult to quantitatively evaluate the performance of an IRQ RBIR system because the retrieval results vary with different selections of query regions. Following our experiments of region filtering presented in Section 6.2.2, Fig. 12(b) shows the average precisions for

Table 5

The category IDs and the semantic names of the extended dataset.

ID	11	12	13	14	15	16	17	18	19	20
Name	Glaciers and Mountains	Monument Valley	Autumn	Cavems	Fireworks	Doors of Paris	Dolphins and Whales	Owls	Fitness	Prehistoric World
ID	21	22	23	24	25	26	27	28	29	30
Name	Bonsai and Penj ing	Tropical Plants	Beautiful Roses	Museum Duck Decoys	Museum Easter Eggs	Plants and Animals in Desert	Chimpanzee	Close-up	Moths	Hawks and Falcons
ID	31	32	33	34	35	36	37	38	39	40
Name	Bears	Lions	Orchids of the World	Penguins	Weasels and Hares	Garden	Leopards	Models	Cloud	Insects
ID	41	42	43	44	45	46	47	48	49	50
Name	Waves	Reptilia	Poker	Moths and Butterflie	Office Interiors	Pedigree Cates	Heads of Animals	Bird Illustrations	Dinosaur Illustrations	Wild Birds



(a) Average recalls of region filtering (b) Average precisions of retrieval results

Fig. 12. Quantitative results of region filtering and retrieval.

Table 6
Details of averages recalls for the best-10 and worst-5 categories.

Rank	500/5000		1000/5000		1500/5000		2000/5000	
	Category Name	Recall(%)	Category Name	Recall(%)	Category Name	Recall(%)	Category Name	Recall(%)
1	Buses	93.25	Buses	98.65	Buses	99.65	Buses	99.95
2	Cuisine	66.85	Doors of Paris	83.45	Doors of Paris	92.8	Doors of Paris	96.7
3	African	60.6	Cuisine	80.8	Cuisine	88.7	Cuisine	92.7
4	Doors of Paris	59.4	African	77.45	African	86.65	African	91.75
5	Poker	51.5	Historical Remains	70.8	Bird Illustrations	81.05	Bird Illustrations	89.65
6	Historical Remains	48.4	Poker	65.9	Historical Remains	80.9	Historical Remains	88.1
7	Beautiful Rose	42.55	Penguins	62.15	Office Interiors	79.3	Office Interiors	87.1
8	Tropical Plants	42.2	Tropical Plants	61.85	Poker	76.75	Dinosaur Decoys	86.6
9	Orchids of the World	41.75	Orchids of the World	60.5	Tropical Plants	75	Poker	83.4
10	Museum Duck Decoys	40	Bird Illustrations	56.6	Penguins	72.55	Tropical Plants	82.95
46	Hawks and Falcons	15.45	Waves	26.15	Waves	39.15	Lions	50.85
47	Heads of Animals	14.45	Heads of Animals	25.15	Heads of Animals	36.7	Leopards	49.85
48	Reptilia	12.9	Reptilia	24.85	Reptilia	32.6	Weasels and Hares	44.75
49	Weasels and Hares	12.3	Weasels and Hares	22.1	Weasels and Hares	30.1	Reptilia	44.75
50	Cloud	11.35	Cloud	20.9	Cloud	28.2	Cloud	41.2

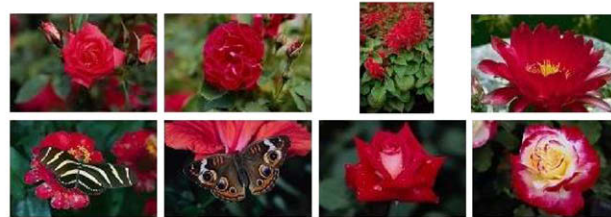
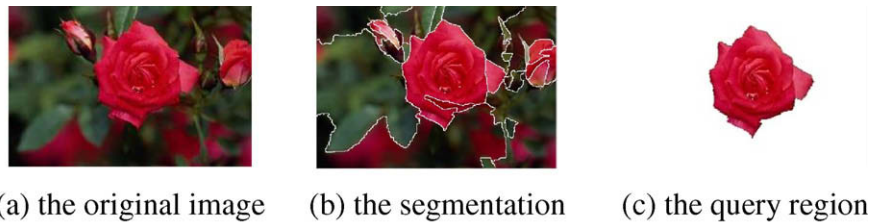


Fig. 13. Retrieval example with query by the subject of interest.



Fig. 14. Retrieval example associated with query regions with backgrounds and without backgrounds.

different numbers of retrieval results when we adopt the same query set with the query being a whole image.

7. Conclusion and future work

This paper presents the color-size feature for integrating color and region-size information in an image. We explore the use of color and region-size features as the visual representation of image regions. We have also designed an IRQ RBIR system to allow the user to specify regions of interest as a query. The proposed region filtering method can screen out most irrelevant images based on the color-size histogram, with only the resulting candidate images being ranked using the EMD-based similarity measure. The results from our experiments demonstrate the efficiency of using the proposed color-size feature and our RBIR system.

Some future tasks are needed to extend this work. The first is to design a scheme of relevance feedbacks [6,22] to learn what the user wants to retrieve, based on positive and negative examples specified by the user. Because modeling human perception is very difficult both in terms of visual features and the similarity measure, relevance feedbacks is a good way to interactively estimate and learn the concepts including in the user query. The second task is to design a multiscale representation for the color-size feature to overcome the problem of the scaling invariance. Besides, we need to design an evaluation model for an IRQ RBIR system. In our system, the query is dependent on the regions selected by the user, and it is challenging to design automatic evaluation for an IRQ RBIR system.

Acknowledgment

This work was in part supported by National Science Council, Taiwan, under Grant No. NSC 97-2218-E-147-002 and by Ministry of Economic Affairs, Taiwan, under Grant No. 97-EC-17-A-02-S1-032.

References

- [1] K. Barnard, D. Forsyth, Learning the semantics of words and pictures, Proceedings of International Conference on Computer Vision, 2 (2001) 408–415.
- [2] K. Barnard, N.V. Shirahatti, A method for comparing content based image retrieval methods, Internet Imaging IX, Electronic Imaging, 2003.
- [3] C. Carson, S. Belongie, H. Greenspan, J. Malik, Blobworld: image segmentation using expectation-maximization and its application to image querying, IEEE Transaction on Pattern Analysis and Machine Intelligence 24 (8) (2002) 1026–1038.
- [4] V. Castelli, L.D. Bergman, Image Databases: Search and Retrieval of Digital Imagery, John Wiley & Sons, Inc, 2002.
- [5] C.-C. Chiang, M.-H. Hsieh, Y.-P. Hung, G.C. Lee, Region Filtering Using Color and Texture Features for Image Retrieval, Proceedings of International Conference on Image and Video Retrieval, Singapore, 2005, pp. 487–496.
- [6] I.J. Cox, M.L. Miller, T.P. Minka, T.V. Papathomas, P.N. Yianilos, The Bayesian image retrieval system, PicHunter: theory, implementation, and psychophysical experiments, IEEE Transaction on Image Processing 9 (1) (2000) 20–37.
- [7] R. Datta, J. Li, J.Z. Wang, Content-based image retrieval—approaches and trends of the new age, Proceedings of ACM SIGMM International Workshop on Multimedia information retrieval, 2005.
- [8] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, second ed., John Wiley & Sons, Inc, 2001.
- [9] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, W. Equitz, Efficient and effective querying by image content, Journal of Intelligent Information Systems 3 (3–4) (1994) 231–262.
- [10] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Workshop on Generative-Model Based Vision, 2004.
- [11] S.L. Feng, R. Manmatha, V. Lavrenko, Multiple bernoulli relevance models for image and video annotation, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, 2004.
- [12] R. Fergus, L. Fei-Fei, P. Perona, A. Zisserman, Learning object categories from Google's image search, Proceedings of International Conference on Computer Vision, 2005.
- [13] P. Howarth, S. Ruger, Evaluation of texture features for content-based image retrieval, Proceedings of International Conference on Image and Video Retrieval, 2004.
- [14] J. Jeon, V. Lavrenko, R. Manmatha, Automatic image annotation and retrieval using cross-media relevance models, Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval, 2003.
- [15] F. Jing, M. Li, H.-J. Zhang, B. Zhang, An efficient and effective region-based image retrieval framework, IEEE Transaction on Image Processing, 13(5) (2004).
- [16] J. Li, J.Z. Wang, Automatic linguistic indexing of pictures by a statistical modeling approach, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (9) (2003) 1075–1088.
- [17] B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Transaction on Pattern Analysis and Machine Intelligence (1996) 837–842.
- [18] B.S. Manjunath, J.-R. Ohm, V.V. Vasudevan, A. Yamada, Color and texture descriptors, IEEE Transaction Circuits Systems Video Technologies (Special Issue on MPEG-7) 11 (6) (2001) 703–715.
- [19] R. Maree, P. Geurts, J. Piater, L. Wehenkel, Random Subwindows for Robust Image Classification, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2005.
- [20] S. Mehrotra, Y. Rui, M. Ortega-Binderberger, T.S. Huang, Supporting content-based queries over images in MARS, Proceedings of IEEE International Conference on Multimedia Computing and Systems, (1997) 632–633.
- [21] Y. Rubner, C. Tomasi, L.J. Guibas, The Earth Mover's Distance as a metric for image retrieval, International Journal of Computer Vision 40 (2) (2000) 99–121.
- [22] Y. Rui, T.S. Huang, M. Ortega, S. Mehrotra, Relevance feedback: a power tool for interactive content-based image retrieval, IEEE Transactions on Circuits and Systems for Video Technology 8 (5) (1998) 644–655.
- [23] J. Sivic, B.C. Russell, A.A. Efros, A. Zisserman, W.T. Freeman, Discovering objects and their location in images, Proceedings of International Conference on Computer Vision, 2005.
- [24] J.R. Smith, C.S. Li, Image classification and querying using composite region templates, Computer Vision and Image Understanding (1999) 165–174.
- [25] M. Stricker, M. Orengo, Similarity of color images, Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases (1995) 381–392.
- [26] M.J. Swain, D.H. Ballard, Color indexing, International Journal of Computer Vision 7 (1) (1991) 11–32.
- [27] L. Vincent, P. Soille, Watersheds in digital spaces: an efficient algorithm based on immersion simulations, IEEE Transaction on Pattern Analysis and Machine Intelligence 13 (6) (1991) 583–598.
- [28] K. Vu, A. Hua, J.H. Oh, A noise-free similarity model for image retrieval systems, Proceedings of SPIE Conference on Storage and Retrieval Media Databases, San Jose, CA, 2001, pp. 1–11.
- [29] D. Wang, A multiscale gradient algorithm for image segmentation using watersheds, Pattern Recognition 30 (12) (1997) 2043–2052.
- [30] J.Z. Wang, J. Li, G. Wiederhold, SIMPLcity: semantics-sensitive integrated matching for picture libraries, IEEE Transaction on Pattern Analysis and Machine Intelligence (2001) 947–963.
- [31] R. Weber, M. Mlivonic, Efficient region based image retrieval, Proceedings of ACM International Conference on Information and Knowledge Management, New Orleans, Louisiana, USA, 2003.