Available online at www.sciencedirect.com
ScienceDirect

Pattern Recognition 40 (2007) 262–282

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

A survey of content-based image retrieval with high-level semantics

Ying Liu[a,*], Dengsheng Zhang[a], Guojun Lu[a], Wei-Ying Ma[b]

[a] *Gippsland School of Computing and Information Technology, Monash University, Vic 3842, Australia*
[b] *Microsoft Research Asia, No. 49 ZhiChun Road, Beijing 100080, China*

Received 19 November 2005; received in revised form 20 March 2006; accepted 28 April 2006

Abstract

In order to improve the retrieval accuracy of content-based image retrieval systems, research focus has been shifted from designing sophisticated low-level feature extraction algorithms to reducing the 'semantic gap' between the visual features and the richness of human semantics. This paper attempts to provide a comprehensive survey of the recent technical achievements in high-level semantic-based image retrieval. Major recent publications are included in this survey covering different aspects of the research in this area, including low-level image feature extraction, similarity measurement, and deriving high-level semantic features. We identify five major categories of the state-of-the-art techniques in narrowing down the 'semantic gap': (1) using object ontology to define high-level concepts; (2) using machine learning methods to associate low-level features with query concepts; (3) using relevance feedback to learn users' intention; (4) generating semantic template to support high-level image retrieval; (5) fusing the evidences from HTML text and the visual content of images for WWW image retrieval. In addition, some other related issues such as image test bed and retrieval performance evaluation are also discussed. Finally, based on existing technology and the demand from real-world applications, a few promising future research directions are suggested.

Keywords: Content-based image retrieval; Semantic gap; High-level semantics; Survey

1. Introduction

With the development of the Internet, and the availability of image capturing devices such as digital cameras, image scanners, the size of digital image collection is increasing rapidly. Efficient image searching, browsing and retrieval tools are required by users from various domains, including remote sensing, fashion, crime prevention, publishing, medicine, architecture, etc. For this purpose, many general-purpose image retrieval systems have been developed. There are two frameworks: text-based and content-based. The text-based approach can be tracked back to 1970s. In such systems, the images are manually annotated by text descriptors, which are then used by a database management system (DBMS) to perform image retrieval. There are two disadvantages with this approach. The first is that a considerable level of human labour is required for manual annotation. The second is the annotation inaccuracy due to the subjectivity of human perception [1,2]. To overcome the above disadvantages in text-based retrieval system, content-based image retrieval (CBIR) was introduced in the early 1980s. In CBIR, images are indexed by their visual content, such as color, texture, shapes. A pioneering work was published by Chang in 1984, in which the author presented a picture indexing and abstraction approach for pictorial database retrieval [3]. The pictorial database consists of picture objects and picture relations. To construct picture indexes, abstraction operations are formulated to perform picture object clustering and classification. In the past decade, a few commercial products and experimental prototype systems have been developed, such as QBIC [4], Photobook [5], Virage [6], VisualSEEK [7], Netra [8], SIMPLIcity [9]. Comprehensive surveys in CBIR can be found in Refs. [10,11].

[*] Corresponding author. Tel.: +61 3 99027101; fax: +61 3 99026842.
E-mail addresses: ying.liu@infotech.monash.edu.au (Y. Liu), dengsheng.zhang@infotech.monash.edu.au (D. Zhang), guojun.lu@infotech.monash.edu.au (G. Lu), wyma@microsoft.com (W.-Y. Ma).

0031-3203/$30.00 © 2006 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2006.04.045

1.1. The semantic gap

The fundamental difference between content-based and text-based retrieval systems is that the human interaction is an indispensable part of the latter system. Humans tend to use high-level features (concepts), such as keywords, text descriptors, to interpret images and measure their similarity. While the features automatically extracted using computer vision techniques are mostly low-level features (color, texture, shape, spatial layout, etc.). In general, there is no direct link between the high-level concepts and the low-level features [2].

Though many sophisticated algorithms have been designed to describe color, shape, and texture features, these algorithms cannot adequately model image semantics and have many limitations when dealing with broad content image databases [12]. Extensive experiments on CBIR systems show that low-level contents often fail to describe the high-level semantic concepts in user's mind [13]. Therefore, the performance of CBIR is still far from user's expectations.

In Ref. [1], Eakins mentioned three levels of queries in CBIR.

Level 1: Retrieval by primitive features such as color, texture, shape or the spatial location of image elements. Typical query is query by example, 'find pictures like this'.

Level 2: Retrieval of objects of given type identified by derived features, with some degree of logical inference. For example, 'find a picture of a flower'.

Level 3: Retrieval by abstract attributes, involving a significant amount of high-level reasoning about the purpose of the objects or scenes depicted. This includes retrieval of named events, of pictures with emotional or religious significance, etc. Query example, 'find pictures of a joyful crowd'.

Levels 2 and 3 together are referred to as semantic image retrieval, and the gap between Levels 1 and 2 as the semantic gap [1].

More specifically, the discrepancy between the limited descriptive power of low-level image features and the richness of user semantics, is referred to as the 'semantic gap' [14,15].

Users in Level 1 retrieval are usually required to submit an example image or sketch as query. But what if the user does not have an example image at hand? Semantic image retrieval is more convenient for users as it supports query by keywords or by texture.

Therefore, to support query by high-level concepts, a CBIR systems should provide full support in bridging the 'semantic gap' between numerical image features and the richness of human semantics [13,15].

1.2. High-level semantic-based image retrieval

How do we relate low-level image features to high-level semantics? Our survey shows that the state-of-the-art techniques in reducing the 'semantic gap' include mainly five categories: (1) using object ontology to define high-level concepts, (2) using machine learning tools to associate low-level features with query concepts, (3) introducing relevance feedback (RF) into retrieval loop for continuous learning of users' intention, (4) generating semantic template (ST) to support high-level image retrieval, (5) making use of both the visual content of images and the textual information obtained from the Web for WWW (the Web) image retrieval.

Retrieval at Level 3 is difficult and less common. Possible Level 3 retrieval can be found in domain specific areas such as art museums or newspaper library. Current systems mostly perform retrieval at Level 2. There are three fundamental components in these systems: (1) low-level image feature extraction, (2) similarity measure, (3) 'semantic gap' reduction.

Excellent survey on low-level image feature extraction in CBIR system can be found in Ref. [11]. In this paper, we focus on CBIR with high-level semantics. The rest of the paper is organized as follows. In Section 2, we briefly review various low-level image features used in high-level semantic-based CBIR systems. Image similarity measure is also discussed in Section 2. Section 3 focuses on different methods in narrowing down the 'semantic gap'. In Section 4, test image dataset and performance evaluation (PE) are discussed. Section 5 includes a few other issues related with CBIR systems which are suggested as promising research directions. Finally, Section 6 concludes this paper.

2. Low-level image features

Low-level image feature extraction is the basis of CBIR systems. To performance CBIR, image features can be either extracted from the entire image or from regions. As it has been found that users are usually more interested in specific regions rather than the entire image, most current CBIR systems are region-based. Global feature based retrieval is comparatively simpler. Representation of images at region level is proved to be more close to human perception system [16]. In this paper, we focus on region-based image retrieval (RBIR).

To perform RBIR, the first step is to implement image segmentation. Then, low-level features such as color, texture, shape or spatial location can be extracted from the segmented regions. Similarity between two images is defined based on region features. This section includes a brief description of these three parts focusing on what are used in RBIR system with high-level semantics.

2.1. Image segmentation

Automatic image segmentation is a difficult task. A variety of techniques have been proposed in the past, such as curve evolution [17], energy diffusion [18], and graph partitioning [19]. Many existing segmentation techniques work well for

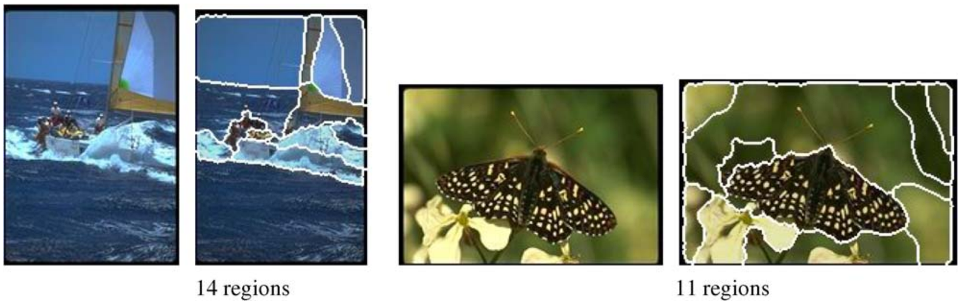


Fig. 1. JSEG segmentation results.

images that contain only homogeneous color regions, such as direct clustering methods in color space [20]. These apply to retrieval systems working only with colors [21,22].

However, natural scenes are rich in both color and texture, and a wide range of natural images can be considered as a mosaic of regions with different colors and textures. Texture is an important feature in defining high-level concepts. As stated in Ref. [23], texture is the main difficulty in a segmentation method. Many texture segmentation algorithms require the estimation of texture model parameters which is a very difficult task [23]. 'JSEG' segmentation [23] overcomes these problems. Instead of trying to estimate a specific model for texture region, it tests for the homogeneity of a given color-texture pattern. 'JSEG' consists of two steps. In the first step, image colors are quantized to several classes. Replacing the image pixels by their corresponding color class labels, we can obtain a class-map of the image. Spatial segmentation is then performed on this class-map which can be viewed as a special type of texture composition. The algorithm produces homogeneous color-texture regions and is used in many systems [16,24,25]. Fig. 1 gives two examples.

Blobworld segmentation [26] is another widely used segmentation algorithm [24,27]. It is obtained by clustering pixels in a joint color-texture-position feature space. Firstly, the joint distribution of color, texture, and position features is modelled with a mixture of Gaussians. Then expectation maximization (EM) algorithm is used to estimate the parameters of the model. The resulting pixel-cluster membership provides a segmentation of the image. The resulted regions correspond roughly to objects.

Some systems design their own segmentations in order to obtain the desired region features during segmentation, be it color, texture, or both [9,28–31]. These algorithms are usually based on k-means clustering of pixel/block features. In Ref. [9], firstly, an image is segmented into small blocks of size $4*4$ from which color and texture feature are extracted. Then k-means clustering is applied to cluster the feature vectors into several classes with each class corresponding to one region. Blocks in same class are classified into same region. A so-called KMCC (k-means with connectivity constraint) is proposed in Ref. [31] to segment objects from images. It is extended from the k-means algorithm. In this algorithm, the spatial proximity of each region is taken into account by defining a new center for the k-means algorithm and by integrating the k-means with a component labelling procedure.

The use of segmentation algorithm depends on the requirements of the system and the data set used. It is hard to judge which algorithm is the best. For example, JSEG provides color-texture homogeneous regions, while KMCC intends to obtain objects which are usually not homogeneous. Compared with JSEG, KMCC is computationally more intensive. JSEG and Blobworld segmentations seem to be the most widely used so far.

2.2. *Low-level image features*

Many sophisticated feature extraction algorithms have been designed and good surveys are available. Here we focus on the features used in RBIR systems with high-level semantics.

2.2.1. *Color feature*

Color feature is one of the most widely used features in image retrieval. Colors are defined on a selected color space. Variety of color spaces are available, they often serve for different applications. Description of different color spaces can be found in Ref. [32]. Color spaces shown to be closer to human perception and used widely in RBIR include, RGB, LAB, LUV, HSV (HSL), YCrCb and the hue-min-max-difference (HMMD) [21,25,27,31,33]. Common color features or descriptors in RBIR systems include, color-covariance matrix, color histogram, color moments, and color coherence vector [16,28,34–36]. MPEG-7 has included dominant color, color structure, scalable color, and color layout as color features [37]. In Ref. [38], the authors are interested in objects taken from different point of view and illumination. As the result, a set of viewpoint invariant color features have been computed. The color invariants are constructed on the basis of hue, hue-hue pair and three color features computed from reflection model.

Most of those color features though efficient in describing colors, are not directly related to high-level semantics. For convenient mapping of region color to high-level

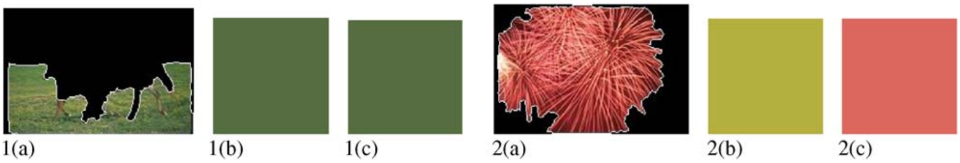


Fig. 2. Average color and dominant color: (a) original region; (b) average color; (c) dominant color.

semantic color names, some systems use the average color of all pixels in a region as its color feature [22,31,39]. Although most segmentation tends to provide homogeneous color regions, due to the inaccuracy of segmentation, average color could be visually different from that of the original region. In Ref. [25], a dominant color in HSV space is defined as the perceptual color of a region. To obtain dominant color, the authors first calculate the HSV space color histogram ($10 * 4 * 4$ bins) of a region and select the bin with maximum size. Then the average HSV value of all the pixels in the selected bin is defined as the dominant color. It is observed that in most cases, average color and dominant color are very similar, as in Fig. 2(1). However, in some cases, they can be visually very different as in Fig. 2(2).

The selection of color features depends on the segmentation results. For instance, if the segmentation provides objects which do not have homogeneous color, obviously average color is not a good choice. It is stated that for more specific applications such as human face database, domain-knowledge can be explored to assign a weight to each pixel in computing the region colors [22].

It should be noted that in most of the CBIR works, the color images are not pre-processed. Since color images are often corrupted with noise due to capturing devices or sensors, it will improve retrieval accuracy significantly if effective filter is applied to remove the color noise. The pre-process can be essential especially when the retrieval results are used for human interpretation. A number of such color filters are available for this purpose [32,40,41].

2.2.2. Texture feature

Texture is not so well-defined as color features, some systems do not use texture features [2,21,22,31,42]. However, texture provides important information in image classification as it describes the content of many real-world images such as fruit skin, clouds, trees, bricks, and fabric. Hence, texture is an important feature in defining high-level semantics for image retrieval purpose.

Texture features commonly used in image retrieval systems include spectral features, such as features obtained using Gabor filtering [8] or wavelet transform [9], statistical features characterizing texture in terms of local statistical measures, such as the six Tamura texture features [43], and wold features proposed by Liu et al. [44]. Among the six Tamura features: coarseness, directionality, regularity, contrast, line-likeness, contrast and roughness, the first three are more significant [43]. The other three are related to the first three and do not add much to the effectiveness of texture description. MPEG-7 has employed the regularity, directionality and coarseness as the texture browsing descriptor [33,37]. The wold features of periodicity, randomness and directionality have been proved to work well on Brodatz textures [45].

The limitation of Tamura features is that there was no work at multiple resolutions to account for scale. Wold feature is also affected by image distortions such as scale and orientation variations due to perspective distortion [30]. Though working well on Brodatz textures, these features are proved to be less effective when applied to natural scene image retrieval as texture regions in such images are not so structured and homogeneous [30].

Among the various texture features, Gabor features and wavelet features are widely used for image retrieval and have been reported to well match the results of human vision study [8,9,37]. Gabor filtering and wavelet transform are originally designed for rectangular images. However, regions in RBIR systems are of arbitrary-shapes. How to extract texture features from arbitrary-shaped regions in RBIR systems? In some systems, texture features are obtained based on the texture property of pixels or small blocks contained in the region [8,31]. For example, in Ref. [8], for each region, the mean of the texture features of all the $4*4$ blocks it contains is used as the region feature. The problem of such feature is that they cannot sufficiently describe the texture property of the entire region. An intuitive way to solve this problem is to extend the arbitrary-shaped region into a rectangular area by padding some values outside the boundary and then apply block transforms. However, as regions in real-world images are usually not homogeneous texture, such initial padding will introduce spurious components which do not describe the original region and hence degrade the performance of the texture feature obtained. Still another possible solution is to obtain an inner rectangle (IR) from a region onto which block transforms can be performed to generate coefficients from which texture feature can be calculated. This works well when the region texture is homogeneous and the IR carries enough information to describe the region's texture property. However, image regions in real-world images are usually not homogeneous. In addition, in many cases, we can only obtain an IR covering a small area of the original region. Hence, the texture feature obtained from IR cannot well represent the property of the entire region. To solve this problem, an efficient texture feature extraction algorithm for arbitrary-shaped regions is presented in Ref. [46]. This algorithm extends an arbitrary-shaped region into a rectangle area by initial padding. Then a projection onto convex sets

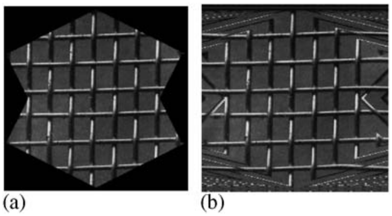


Fig. 3. Arbitrary-shaped region and padded results: (a) original region; (b) mirroring padded result.

(POCS) loop is applied to find a set of coefficients best describing the region by iterative projection between the image domain and its transform domain. Finally, texture features can be extracted from the coefficients obtained. Fig. 3 gives an example of initial padding.

The edge histogram descriptor (EHD) is found to be quite effective for representing natural images [37]. It captures the spatial distribution of edges, somewhat in the same idea as the color layout descriptor. To compute the EHD, a given image is first sub-divided into 4×4 sub-images, and local edge histograms for each of these sub-images is computed. Edges are broadly grouped into five categories: vertical, horizontal, 45°, 135° and neutral. Thus, each local histogram has five bins corresponding to the above five categories. The image partitioned into 16 sub-images results in 80 bins. These bins are non-uniformly quantized using 3 bits/bin, resulting in a descriptor of size 240 bits. But the EHD can be very sensitive to objects or scene distortions.

Huang and Dai have computed the gradient vector from the subband images of a wavelet decomposition as texture feature [47]. The gradient vector is a similar approach to EHD.

2.2.3. *Shape*

Shape is a fairly well-defined concept. Shape features of general applicability include aspect ratio, circularity, Fourier descriptors, moment invariants, consecutive boundary segments [48], etc.

Shape features are important image features though they have not been widely used in RBIR as color and texture features. Shape features have shown to be useful in many domain specific images such as man-made objects. For color images used in most papers, however, it is difficult to apply shape features compared to color and texture due to the inaccuracy of segmentation. Despite the difficulty, shape features are used in some systems and has shown potential benefit for RBIR. For example, in Ref. [31], simple shape features such as eccentricity and orientation are used. The system in Ref. [34] uses normalized inertia of order 1–3 to describe region shape. In Ref. [28], gross region shape descriptors based on area and second-order moments are used. MPEG-7 has included three shape descriptors for object-based image retrieval, one is the 3-D shape descriptor derived from 3-D meshes of shape surface, one is for region-based shape derived from Zernik moments and the other is for contour-based shape derived from curvature scale space (CSS) [37]. Although the CSS descriptor is invariant to translation, scaling and rotation, it is sensitive to general distortions which can be resulted from objects taken from different point of view. Mokhtarian and Abbasi have extended the CSS descriptor to be robust to affine transform which is a common way to approximate general shape distortions [49].

2.2.4. *Spatial location*

Besides color and texture, spatial location is also useful in region classification. For example, 'sky' and 'sea' could have similar color and texture features, but their spatial locations are different with sky usually appears at the top of an image, while sea at the bottom.

Spatial location usually are simply defined as 'upper, bottom, top' according to the location of the region in an image [50,51]. In Ref. [8], region centroid and its minimum bounding rectangle are used to provide spatial location information. In Ref. [31], spatial center of a region is used to represent its spatial location.

Relative spatial relationship is more important than absolute spatial location in deriving semantic features. 2D-string [52] and its variants are the most common structure used to represent directional relationships between objects such as 'left/right', 'below/above'. However, such directional relationships alone are not sufficient to represent the semantic content of images ignoring the topological relationships. To better support semantic-based image retrieval, a spatial context modelling algorithm is presented in Ref. [53] which considers six spatial relationships between region pairs: left, right, up, down, touch and front. An interesting method was proposed by Smith et al. [29]. The system uses a composite region template (CRT) to define the spatial arrangement of regions and each semantic class is characterized by the CRTs obtained from a collection of sample images [29].

2.3. *Similarity measure*

In RBIR systems, image similarity is measured at two levels. The first is region-level. That is to measure the distance between two regions based on their low-level features. The second is at image level. That is to measure the overall similarity of two images which might contain different number of regions.

Most researchers employ the Minkowski-type metric to define region distance. Suppose we have two regions represented by two p dimensional vectors $(x_1, x_2, \ldots, x_p)$, $(y_1, y_2, \ldots, y_p)$, respectively. The Minkowski metric is defined as

$$d(X, Y) = \left(\sum_{i=1}^{p} |x_i - y_i|^r \right)^{1/r}. \tag{1}$$

Particularly, when $r = 2$, it is the well-known Euclidean distance (L_2 distance). When r is 1, it is the Manhattan distance (L_1 distance).

An often-used variant version is the weighted Minkowski distance function which introduces weighting to identify important features

$$d(X,Y)=\left(\sum_{i=1}^{p} w_i|x_i-y_i|^r\right)^{1/r}, \tag{2}$$

where w_i, $i=1,\ldots,p$ is the weight applied to different features.

Other distances are also used in image retrieval, such as the Canberra distance, angular distance, Czekanowski coefficient [54], inner product, dice coefficient, cosine coefficient and Jaccard coefficient [55].

The overall similarity of two images is more difficult to measure. Basically, there are two ways.

(1) *One-One match*: This means each region in the query image is only allowed to match one region in the target image and vice versa. As in Ref. [56], each query region of the query image is associated to a single 'best match' region in the target image. Then the overall similarity is defined as the weighted sum of the similarity between each query region in the query image and its 'best match' in the target image, while the weight is related to region size.

(2) *Many-Many match*: This means each region in the query image is allowed to match more than one region in the target image and vise versa. A widely used method is the Earth Mover' Distance (EMD) [57]. EMD is a general and flexible metric. It measures the minimal cost required to transform one distribution into another based on a traditional transportation problem from linear optimization, for which efficient algorithms are available. EMD matches perceptual similarity well and can be applied to variable-length representations of distributions, hence it is suitable for image similarity measure in RBIR system [16,57].

Li et al. propose an integrated region matching (IRM) scheme which allows for matching a region of one image to several regions of another image and thus decreases the impact of inaccurate segmentation [34]. In this definition, a matching between any two regions is assigned with a significance credit. This forms a significance matrix between two sets of regions (one set is of the query image, another set is of the target image). The overall similarity of two images is defined based on the significance matrix in a way similar to EMD.

Though Minkowski metric is widely used in current systems to measure region distance, intensive experiments show that it is not very effective in modelling perceptual similarity [58]. How to measure perceptual similarity is still a largely unanswered question. There are some works done in trying to solve this problem. For example, in Ref. [58], a dynamic partial distance function (DPF) is defined, which reduces the dimension of feature vectors by dynamically choosing a smaller amount of dimensions. Let $\delta_i=|x_i-y_i|, i=1,\ldots,p$, the authors define $\Delta_m=\{m \text{ smallest } \delta\text{' s of } (\delta_1,\ldots,\delta_p)\}$. Then DPF is defined as

$$d(m,r)=\left(\sum_{\delta_i\in\Delta_m}\delta_i^r\right)^{1/r}. \tag{3}$$

There are two parameters to be adjusted m and r. Initial experimental results demonstrate that DPF can provide more accurate retrieval results than Minkowski metrics. However, the value of m is data-dependent, this makes the algorithm inflexible. In addition, to be broadly used in image retrieval systems, further study is required to confirm its performance in various applications.

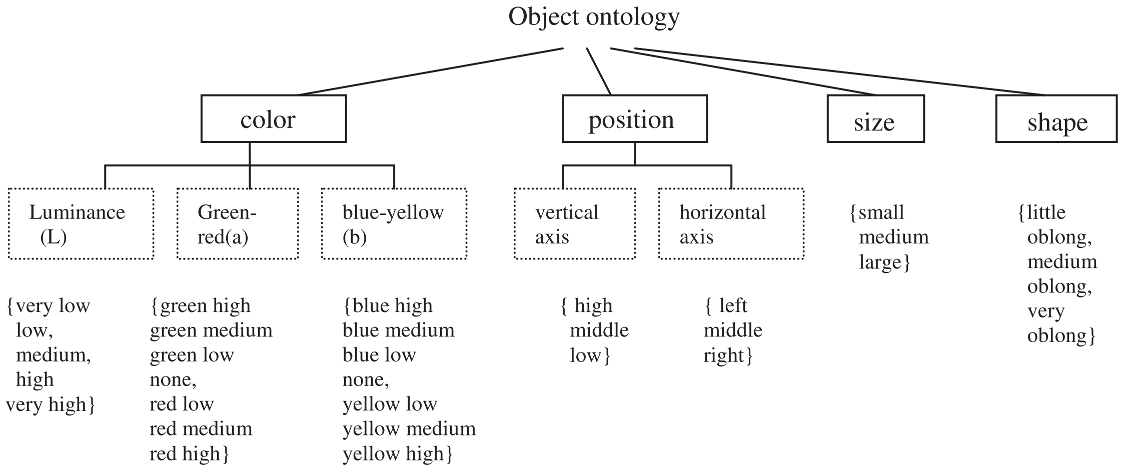
In Ref. [59], a perceptual distance for shape similarity measure is presented. Each shape is characterized with a set of tokens. A metric distance between tokens is first defined then a non-metric distance is defined as the collection of token distance to measure shape similarity. The method can be extended into RBIR by treating image regions as the tokens.

Vasconcelos and Lippman proposed a multiresolution manifold distance (MRMD) for face recognition. In the MRMD, two images to be matched are treated as manifolds, and the distance between the two images are the one which minimizes the error of transforming one manifold into the other. In order to reduce the computation, the images are put into multiresolution analysis. The distance measure is suitable for image alignment applications like face recognition and video scene detection [60].

In Ref. [61], similarity measure between different types of image features is taken as a multilevel decision making process. Images in the database are represented by a number of MPEG-7 color and texture descriptors, these descriptors are put into a hierarchical decision fusion framework using fuzzy logic. The advantage of this similarity measurement is that different types of image features can be combined into an integrated feature. In their later work, the authors have extended the decision fusion framework into a supervised learning framework with RF from users [62].

3. Reducing the 'semantic gap'

The state-of-the-art techniques in reducing the semantic gap can be classified in different ways from different point of view. For example, by considering the application domain, they can be classified as those targeting at artwork retrieval [21], scenery image retrieval [27,28,31], WWW images retrieval [63,64], etc. In this paper, we focus on the techniques used to derive high-level semantics and identify five categories as follows. (1) Using object ontology to define high-level concepts [21,31,65–67]. (2) Using supervised or unsupervised learning methods to associate low-level features with query concepts [2,24,27,28,68]. (3) Introducing RF into retrieval loop for continuous learning of users' intention [16,31,69]. (4) Generating ST to support high-level image retrieval [29,70,71]. (5) Making use of both the textual



Object ontology
color
position
size
shape
Luminance (L)
Green-red(a)
blue-yellow (b)
vertical axis
horizontal axis
{small medium large}
{little oblong, medium oblong, very oblong}
{very low low, medium, high very high}
{green high green medium green low none, red low red medium red high}
{blue high blue medium blue low none, yellow low yellow medium yellow high}
{ high middle low}
{ left middle right}

Fig. 4. Object ontology used in Ref. [32].

information obtained from the Web and the visual content of images for Web image retrieval [63,70,72]. Many systems exploit one or more of the above techniques to implement high-level semantic-based image retrieval. For example, (3) is often combined with (1), (2) or (5) [16,31,70,72], (5) is usually combined with the other four techniques [27,63,70].

3.1. Object-ontology

In some cases, semantics can be easily derived from our daily language. For example, sky can be described as 'upper, uniform, and blue region'. In systems using such simple semantics, firstly, different intervals are defined for the low-level image features, with each interval corresponding to an intermediate-level descriptor of images, for example, 'light green, medium green, dark green'. These descriptors form a simple vocabulary, the so-called 'object-ontology' which provides a qualitative definition of high-level query concepts. Database images can be classified into different categories by mapping such descriptors to high-level semantics (keywords) based on our knowledge [31,65–67,73], for example, 'sky' can be defined as region of 'light blue' (color), 'uniform' (texture), and 'upper' (spatial location).

A typical example of such ontology-based system is presented in Ref. [31]. In this system, each region of an image is described by its average color in lab color space, its position in vertical and horizontal axis, its size and shape. The object ontology is shown in Fig. 4.

Quantization of color and texture feature is the key in such systems. To support semantic-based image retrieval, a more effective and widely used way to quantize color information is by color naming. Although millions of colors can be defined in computer system, the colors that can be named by users are limited to about 10–20 [74,75]. Color naming models intend to relate a numerical color space with semantic color names used in natural language. The well-known color naming system is 'CNS' (color naming system) proposed by Berk, Brownston and Kaufman [75]. 'CNS' quantizes HSL space into 627 distinct colors. The basic idea is to quantize the hue value into a set of basic colors. Saturation and luminance are quantized into different bins as adjectives signifying the richness and brightness of the color. The complete set of generic hue names in CNS is *red*, *orange*, *brown*, *yellow*, *green*, *blue* and *purple*, with the addition of achromatic terms *black*, *gray* and *white*, form 10 base colors.

In Ref. [21], 12 hues are defined as fundamental colors, *yellow*, *red*, *green*, *blue*, *orange*, *purple*, and six other colors obtained as the linear combination of them. Then, five levels of luminance and three levels of saturation are identified. This results in 180 reference colors. To relate colors to expression (emotionally) and impression (visually) for painting retrieval, different types of contrasts are defined, light-dark contrast, warm-cold contrast, complementary contrast, etc. For example, colors of yellow, and orange are referred as warm, green and blue are referred as cold. Example query is like this 'find paintings with the following contrasts: light-dark, cold-warm'.

In Ref. [25], the dominant color of a region (in HSV space) is converted to a set of 35 semantic color names as: red, orange, yellow, brown, etc. Semantic color names are related to objects in natural scene images like grass, sky. Example query is 'find images with a sky blue region'. In Ref. [65], based on the author's observation that a small number of colors are usually sufficient to characterize the color information in image region, eight colors are defined based on their RGB values, *red*, *green*, *blue*, *yellow*, *magenta*, *cyan*, *black*, and *white*. These color names are related to objects in natural scenes, for example, white are related to snow, cloud, etc. In this way, the system reduces the 'semantic gap' and supports query by keywords.

Similar to CNS, there is a parallel need for a texture naming system which would standardize the description and representation of textures [76]. However, texture naming is found to be more difficult and so far there is no such a texture naming system available. As a first step towards creating a texture naming system, some researchers try to identify the important features human beings use in texture perception [43,76]. Based on subjective experiment, Rao and Lohse have shown that repetitiveness, directionality and complexity are the three attributes most important to human perception of textures [76]. However, how to obtain these features, and how to map other low-level texture features to these three domains are yet to be further studied [30,76].

Compared with color, texture is not well modelled and understood, much research still needs to be done. Instead of using texture names as keyword for query which is still impossible so far, some researchers quantize perceptual texture features into different intervals and define meaningful texture descriptors. In Refs. [66,67], Tamura features are quantized to different levels as very coarse, medium coarse, fine, very fine; low contrast, high contrast, etc. Combination of such features in logical relationships with *and*, *or* form queries like 'find very fine and low contrast textures'.

For database with specifically collected images, such simple semantics derived based on object-ontology may work fine. However, with large collections of images with various contents, more powerful tools are required to learn the semantics.

3.2. Machine learning

In most cases, to derive high-level semantic features require the use of formal tools such as supervised or unsupervised machine learning techniques [2,28,68,77]. The goal of supervised learning is to predict the value of an outcome measure (for example, semantic category label) based on a set of input measure. In unsupervised learning, there is no outcome measure, and the goal is to describe how the input data are organized or clustered [78].

3.2.1. Supervised learning

Supervised learning such as support vector machine (SVM) [24,27,79], Bayesian classifier [80] are often used to learn high-level concepts from low-level image features.

With strong theoretical foundations available, SVM has been used for object recognition, text classification, etc. and is considered a good candidate for learning in image retrieval system [35,69,81]. SVM is originally designed for binary classification. Assume that we have a set of training data $\{x_1, x_2, \ldots, x_n\}$ as vectors in space $X \subseteq R^d$ belonging to two separate classes with their labels $\{y_1, y_2, \ldots, y_n\}$ and $y_i \in \{-1, 1\}$. We want to find a hyper-plane to separate the data. Among many possible hyper-planes, the *optimal separating plane* (OSP) is the one which maximizes the margin (the distance between the hyper-plane and the nearest data point of each class). As in Fig. 5, the vectors lying on one side are labelled as -1, and those lying on the other side are labelled as $+1$. '*Support vectors*' refer to the training samples that lie closest to the hyper-plane. To learn multiple concepts for image retrieval, a SVM has to be trained for each concept. For example, in Ref. [27], SVM is employed for image annotation. In the training stage, a binary SVM model is trained for each of the 23 selected concepts. In the testing stage, unlabelled regions are fed into all the models, the concept from the model giving the highest positive result is associated with the region.

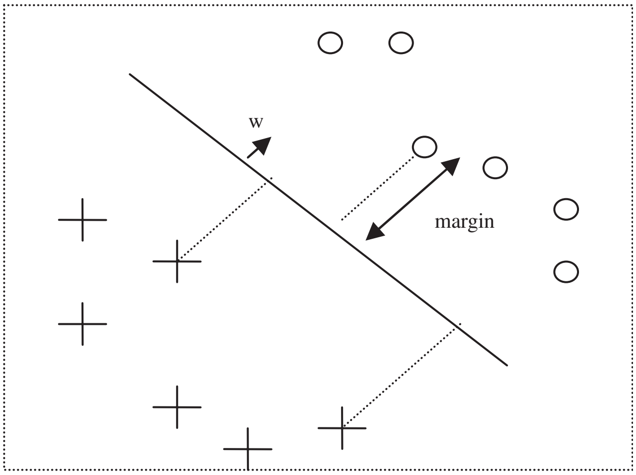


Fig. 5. A simple linear support vector machine.

Another widely used learning method is Bayesian classification [82]. In Ref. [68], using binary Bayesian classifier, high-level concepts of natural scenes are captured from low-level image features. Database images are automatically classified into general types as indoor/outdoor, and the outdoor images are further classified into city/landscape, etc. In Ref. [77], Bayesian network is used for indoor/outdoor image classification.

Other learning techniques such as neural network are also used for concept learning. In Ref. [28], firstly, the author choses 11 categories (concepts): brick, cloud, fur, grass, ice, road, rock, sand, skin, tree, and water. Then a large amount of training data (low-level features of segmented regions) are fed into the neural network classifiers to establish the link between low-level features of an image and its high-level semantics (category labels). A disadvantage of this algorithm is that it requires large amount of training data and is computationally intensive.

In Ref. [24], it is stated that conventional learning algorithms suffer from two problems: (1) a large amount of labelled training samples are needed, and it is very tedious and error-prone to provide such data; (2) the training set is fixed during the learning and application stages. Hence, if the application domain changes, new labelled samples have to be provided to ensure the effectiveness of the classifier. A bootstrapping approach is presented in Ref. [24] to tackle these problems. It starts from a small set of labelled training samples. By using a co-training approach, in which two statistically independent classifiers are used to co-train and co-annotate the unlabelled samples, the algorithm successively annotates a larger set of unlabelled samples. Their experiments show that an improvement of 10% in retrieval accuracy is obtained compared with SVM (400 labelled images for training), with much fewer labelled training samples (only 20 labelled seeds).

Besides the above mentioned algorithms, decision tree (supervised learning) techniques are also used to derive semantic features. Decision tree induction methods such as ID3, C4.5 (improved version of ID3), and CART build up a tree structure by recursively partitioning the input attribute space into a set of non-overlapping spaces [78]. A set of decision rules can be obtained by following the paths from the root of the tree to the leaves. In Ref. [2], the CART decision tree methodology is used to derive decision rules mapping global color distribution (HSV space color histogram) in a given image to textual description (four keywords: Sunset, Marine, Arid images and Nocturne). In Ref. [83], a C4.5 decision tree is built based on a set of images relevant to the query, and then used as a model to classify database images into two classes: relevant and irrelevant. This algorithm is used in the RF loop (will be discussed in Section 3.3) to provide relevant images for the user to label at next iteration. A similar methodology is employed in Ref. [84]. To enhance the performance of RF, the system uses ID3 decision tree to classify the images as relevant/irrelevant based on their color features, instead of directly ranking the images using the modified query obtained in last iteration.

Compared with other learning methods, decision tree learning is conceptually simple, robust with respect to incomplete and noisy input features. In addition, decision tree can be easily translated into a set of rules which can be integrated into an expert system for automatic decision making [78,85]. However, to be used in high-level concepts learning for image retrieval, the underlying problem is the lack of modularity [86,87]. For example, the methods mentioned above use nominal input attributes, but usually low-level image features have continuous values. Though some algorithms [88,89] have been designed to discrete continuous attributes, whether these generally designed algorithms can always provide meaningful splitting of image feature space is a question.

3.2.2. Unsupervised learning

Unlike supervised learning in which the presence of the outcome variable guides the learning process, unsupervised learning has no measurements of outcome, the task is rather to find out how the input feature are organized or clustered.

Image clustering is the typical unsupervised learning technique for retrieval purpose. It intends to group a set of image data in a way to maximize the similarity within clusters and minimize the similarity between different clusters. Each resulting cluster is associated with a class label and images in same cluster are supposed to be similar to each other.

The traditional k-means clustering and its variations are often used for image clustering. In Ref. [90], k-means clustering is applied to low-level color features of a set of training images. Then, the statistics measuring the variation with each cluster are used to derive a set of mappings between the low-level features and the optimal textual characterization (keywords) of the corresponding cluster. The mapping rules derived could be used further to index new untagged images added to the database. In Ref. [80], in order to automatically annotate database images for retrieval purpose, the system firstly cluster image regions into region clusters using a variant of k-means clustering called pair-wise constraints k-means (PCK-means) [91]. The number of clusters is empirically set to 300. Then, the posterior probability of every concept (59 concepts are defined for the image database used) given a region is derived using a semi-naı¨ve Bayesian method [80]. Thus, a new image can be annotated by choosing the concepts with highest probabilities.

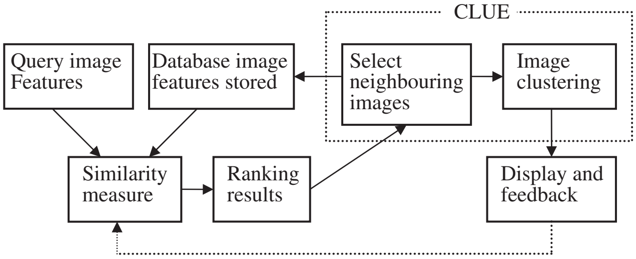


Fig. 6. Image retrieval with CLUE.

Due to the complex distribution of image data (data points are sampled from non-linear manifold), traditional methods such as k-means clustering often cannot well-separate images with different concepts [36]. To handle this problem, a spectral clustering method Normalized cut (NCut) [19] is proposed and has been successfully used in several applications such as image segmentation, image clustering. An extended version of NCut can be found in Ref. [92].

In Ref. [14], a method named ‘CLUE’ is presented to reduce the ‘semantic gap’ in CBIR. Unlike other CBIR systems which display the top matched target images to the users, this system attempts to retrieve semantically coherent image clusters. Given a query image, a collection of target images similar to the query are selected as the neighbour of the query. Based on the hypothesis that images of the same semantics tend to be clustered, NCut clustering is used to cluster these target images into different semantic classes. Then the system displays the image clusters and adjusts the model of similarity measure according to user feedbacks. Fig. 6 is the diagram of the system. Though successful in manifold data clustering, NCut cannot produce an explicit mapping function. To deal with new data points, similarities between the new points and all training data have to be measured. The computation of similarities could be very complicated due to the large size of training set [36]. To tackle these problems, in Ref. [36], a locality preserving clustering (LPC) method is proposed for image clustering. LPC can provide an explicit mapping function. Experimental results show that LPC provides retrieval accuracy comparable to that of NCut, but is more computationally efficient. In addition, retrieval result of LPC is proved to be more accurate than that of k-means clustering.

Probabilistic classification based on Bayes theory is among the most powerful clustering tools. The common maximum-a-posteriori or MAP classifier and its variation maximum-likelihood or ML classifier have shown great promise for the CBIR problem [93,94]. However, traditionally it is difficult to apply the classifiers due to the complexity of the MAP similarity function. In Ref. [94], Vasconelos has shown that the similarity function can be computed efficiently when vector quantizers and Gaussian mixtures are used as models for the probability density functions of the image features.

3.2.3. Object recognition techniques for image retrieval

Object recognition in images is an important problem in computer vision with applications in image annotation, surveillance and image retrieval. Supervised or unsupervised object recognition algorithms have been developed recently which can be used for semantic-based image retrieval. For example, in Ref. [95], an unsupervised scale-invariant learning method is presented to learn and recognize object class models from unlabelled and unsegmented cluttered scenes. In this method, objects are modelled as flexible constellations of parts and a probabilistic representation is used for all aspects of the object: shape, appearance, occlusion and relative scale. In recognition, this model is used in a Bayesian manner to classify images. The flexible nature of the model is demonstrated by excellent results over a range of datasets including geometrically constrained classes (such as faces, cars) and flexible objects (such as animals).

It is recognized that most users like to retrieve images based on objects in images. In Ref. [96], the authors developed a new semi-supervised version of the EM algorithm for learning the distributions of the object classes. Images are represented as sets of feature vector of multiple types of abstract regions. Each abstract region is modelled as a mixture of Gaussian distributions over its feature space. As regions used in recognition can come from different segmentation processes, the regions used are referred to as ‘abstract region’. A key part of this approach is that it does not need to know the location of objects in each image. The experiments on a set of 860 images demonstrate the efficiency of the approach.

In Ref. [97], a two-phrase generative/discriminative learning approach is proposed that can learn to recognize objects using multiple feature types. The goal of this work is to develop a classification methodology for the automatic classification of outdoor scene images. The generative phrase normalizes the description length of images, which can have an arbitrary number of extracted features of each type. In the discriminative phase, a classifier learns which images, as represented by this fixed-length description, contain the target object. Their experimental results, using color, texture and structure features, show promising retrieval performance on 31 elementary object categories and 20 high-level concepts.

Most current approaches to learn visual object categories require thousands of training images. In addition, most algorithms presented in the literature have been tested on only about 10–20 object categories. In Ref. [98], an incremental Bayesian algorithm was developed to learn generative models of object categories from just a few training images. This method makes use of prior information, assembled from object categories which were previously learnt. A generative probabilistic model is used to represent the shape and appearance of a constellation of features belonging to the object. The parameters of the model are learnt incrementally in a Bayesian manner. The algorithm is tested on images of 101 widely varied object categories including face, laptop, strawberry, zebra, cup, chair, etc.

3.3. Relevance feedback (RF)

Compared with the off-line processing algorithms discussed above, RF is an on-line processing which tries to learn the users’ intentions on the fly.

RF is a powerful tool traditionally used in text-based information retrieval systems [99]. It was introduced to CBIR during mid 1990s, with the intention to bring user in the retrieval loop to reduce the ‘semantic gap’ between what queries represent (low-level features) and what the user thinks. By continuous learning through interaction with end-users, RF has been shown to provide significant performance boost in CBIR systems [100,101].

A typical scenario for RF in CBIR is as below [102]:

(1) The system provides initial retrieval results through query-by-example, sketch, etc.
(2) User judges the above results as to whether and to what degree, they are relevant (positive examples)/irrelevant (negative examples) to the query.
(3) Machine learning algorithm is applied to learn the user’ feedback. Then go back to (2).

(2)–(3) are repeated till the user is satisfied with the results. Fig. 7 shows a simple diagram of a CBIR system with RF.

A typical approach in step (3) is to adjust the weights of low-level features to accommodate the users’ need (re-weighting). In this way, the burden of specifying the weight is removed from the user. Examples of such systems are in Refs. [16,100]. ‘Re-weighting’ dynamically updates the weights embedded in the query (not only the weights to different types of features such as color, texture, shape, but also the weights to different components in same feature vector) to model the high-level concepts and perception subjectivity [100].

Another method is called query-point-movement (QPM) [16,103,104]. QPM improves the estimation of the query

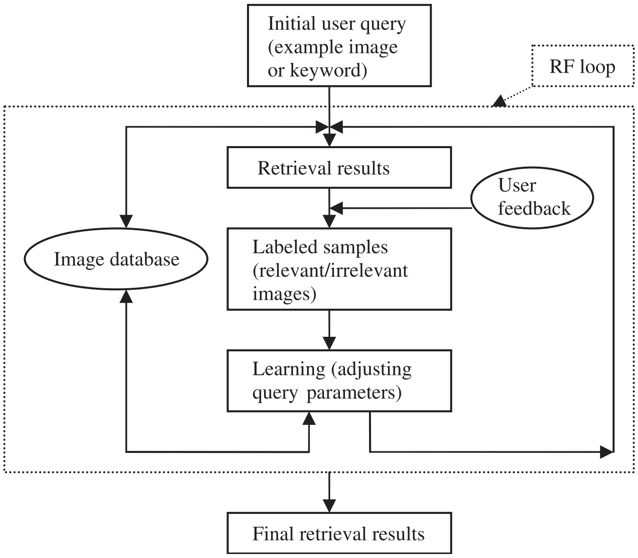

Initial user query (example image or keyword)
RF loop
Retrieval results
User feedback
Image database
Labeled samples (relevant/irrelevant images)
Learning (adjusting query parameters)
Final retrieval results

Fig. 7. CBIR with RF.

point by moving it towards the positive examples and away from the negative examples. The technique often used to iteratively improve this estimation is the Rocchio's formula [55,105,106]

$$Q' = \alpha Q + \beta \left(\frac{1}{N_{R'}} \sum_{i \in D'_R} D_i \right) - \gamma \left(\frac{1}{N_{N'}} \sum_{i \in D'_N} D_i \right), \quad (4)$$

where Q and Q' are the original query and updated query, respectively, D'_R and D'_N are the positive and negative samples returned by the user, $N_{R'}$, $N_{N'}$ are the number of samples in D'_R and D'_N, respectively, and α, β, γ are selected constants.

Both query re-weighting and QPM use nearest-neighbour sampling. That is, the system returns top ranked images for the user to examine and then the query is refined based on the user's feedback [35].

Machine learning techniques can be used in step 3 of RF loop as well. SVM is often used to capture the query concept by separating the relevant images from the irrelevant images using a hyper-plane in a projected space [16,31,69]. One advantage of SVM over other learning algorithms lies in its high generalization performance without the need to add a priori knowledge [69]. Another advantage is that it can work for small training sets [69,103]. To effectively use negative and non-labelled samples, and to learn a query concept faster and more accurately, an active learning method named SVMactive is proposed in Ref. [35].

Generally, the labelled samples provided by the user are limited, and such small training data set will result in weak classification of database images (as relevant/irrelevant). In Refs. [107,108], D-EM (Discriminant-EM) is used to boost the classifier learnt from the limited labelled training data. D-EM is an improved version of EM. EM has the disadvantage that a large number of parameters have to be estimated due to the high dimensionality of the generative model used to model data distribution. D-EM alleviates this problem by adding a D-step. The E-step estimates the membership for each unlabelled sample to augment the labelled training set. D-step identifies a mapping such that the data are clustered in the mapped feature space (a discriminating subspace). Based on the augmented data set, M-step estimates the parameters of the generative model in the lower dimensional discriminating space.

In some papers, decision-tree learning methods such as C4.5, ID3 are used in RF loop to classify the database images into two classes (relevant/irrelevant) depending on whether they are similar to the query image [83,84]. Then the relevant images are presented to the user for another round of RF.

There are different methods adopting different assumptions or problem settings, though under the same notion of 'RF'. A more detailed survey can be found in Ref. [102].

Most of the current RF-based systems uses only the low-level image features to estimate the ideal query parameters and do not address the 'semantic' content of images. Such system works well if the feature vectors can well describe the query. However, for specific object that cannot be sufficiently represented by low-level features, these RF systems will not return many relevant results even with a large number of user feedbacks [109]. To address the limitations of such systems, Ref. [109] provides a system named 'iFind' that performs RF on both the low-level feature vectors and the semantic contents of images represented by keywords. Firstly, a semantic network is constructed on top of an image database and a simple machine learning technique is used to learn from user queries and feedbacks to further improve this semantic network. With the semantic network formed on top of the keyword association with the images, the system can accurately derive the image sematic content for retrieval purposes. In this way, semantic and low-level feature-based RF are seamlessly integrated. Experiments on real-world image collections demonstrate its accuracy and effectiveness.

In most of the RF-based systems, the similarity measurement is fixed while the importance or weight of each descriptor is estimated through the RF from users. In contrast to this conventional approach, the Doulamis' have proposed a generalized nonlinear RF algorithm for image retrieval [110]. In this approach, instead of adjusting the degree of importance of each descriptor, the similarity measure itself is estimated through an online learning mechanism. The method is based on a recursive optimal estimation of a nonlinear parametric relation of known functional components. However, due to the problem of optimization itself, the computation can be expensive and the algorithm may be trapped into local minima.

3.4. Semantic template

'ST', though not yet widely used as the above mentioned techniques, is a promising approach in semantic-based image retrieval. ST is a map between high-level concept and low-level visual features. ST is usually defined as the 'representative' feature of a concept calculated from a collection of sample images [29,70]. In some systems, icons or sample images are provided as well for the convenience of user query [111].

In Ref. [111], Chang et al. introduced the idea of semantic visual template (SVT) to link low-level image feature to high-level concepts for video retrieval. A visual template is a set of icons or example scenes/objects denoting a personalized view of concepts such as meetings, sunsets. The feature vectors of these example scenes/objects are extracted for query process. To generate SVTs, the user first defines the template for a specific concept by specifying the objects and their spatial and temporal constraints, the weights assigned to each feature of each object. This initial query scenario is provided to the system. Through the interaction with users, the system finally converges to a small set of exemplar queries that 'best' match (maximize the recall) the concept in the user' mind.

The generation of SVT in Ref. [111] depends on the interaction with the user and requires the user's in-depth understanding of image features. This impedes its application to ordinary users. Compared to this, the system in Ref. [70] generates ST automatically in the process of RF, based on the understanding that RF is a process by which the user embodies the query semantics. Firstly, the user submits a query image with a concept (keyword) representing the image. After several iterations, the system returns some relevant images to the user. The feature centroid of these images are calculated and used as the representation of the query concept. Then the ST is defined as $ST = \{C, F, W\}$ with C the query concept, F the centroid feature obtained, and W being the weight applied to feature vectors. WordNet [112] is used in this system to construct a network of ST. During the retrieval process, once the user submits a query concept (keyword), the system can find a corresponding ST, and use the corresponding F and W to find similar images. The retrieval process is shown in Fig. 8. The user is imperceptible of the template generation, and can use the system even without any knowledge of feature representation.

Another interesting work is presented by Smith and Li in Ref. [29]. They use the so-called CRTs to decode image semantics. The CRTs define the prototypal spatial arrangements of regions in the images. Given a semantic class, a set of sample images are collected. The system firstly segments each image into homogeneous color regions and extracts five region strings by scanning the image vertically. Then, the system consolidates the region strings by counting the frequencies of the CRTs in the set of region strings obtained from all the sample images. Pooling together the CRTs from each semantic class forms a CRT library. Semantic description of unknown images can be generated by matching the arrangements of image regions to the CRTs in the library. The experiments with a set of 10 semantic classes (beach, building, crab, divers, etc.) demonstrate that this method improves retrieval accuracy compared to traditional methods using color histogram and texture features.

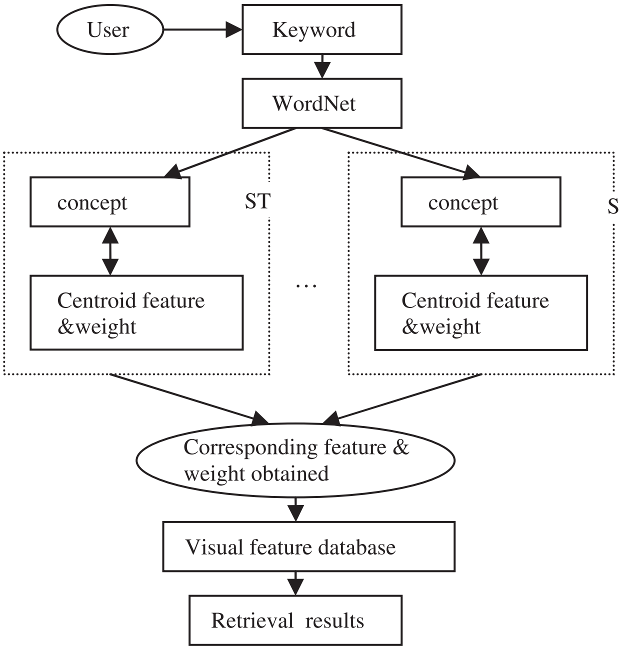


Fig. 8. Image retrieval supported by WordNet and ST.

3.5. Web image retrieval

We classify Web image retrieval as one of the state-of-the-art techniques in high-level image retrieval rather than a specific application domain, as it has some technical difference from image retrieval in other applications.

One advantage in Web image retrieval is that some additional information on the Web is available to facilitate semantic-based image retrieval. For example, the URL of image file often has a clear hierarchical structure including some information about the image such as image category. In addition, the HTML document also contains some useful information in image title, ALT-tag, the descriptive text surrounding the image, hyperlinks, etc. However, such information can only annotate images to a certain extend [63,72].

Existing Web image searching such as Google and AltaVista search images based on textual evidences only [63,64]. Though these approaches can find many relevant images, the retrieval precision is poor as they cannot confirm whether the retrieved images really contain the query concepts. The result is that users have to go through the entire list to find the desired images. This is a time-consuming process as the returned results always contain multiple topics which are mixed together. To improve Web image retrieval performance, researchers are making effort to fuse the evidences from textual information and visual image contents.

In Ref. [72], a bootstrapping co-training framework is used to automatically annotate Web images with a given set

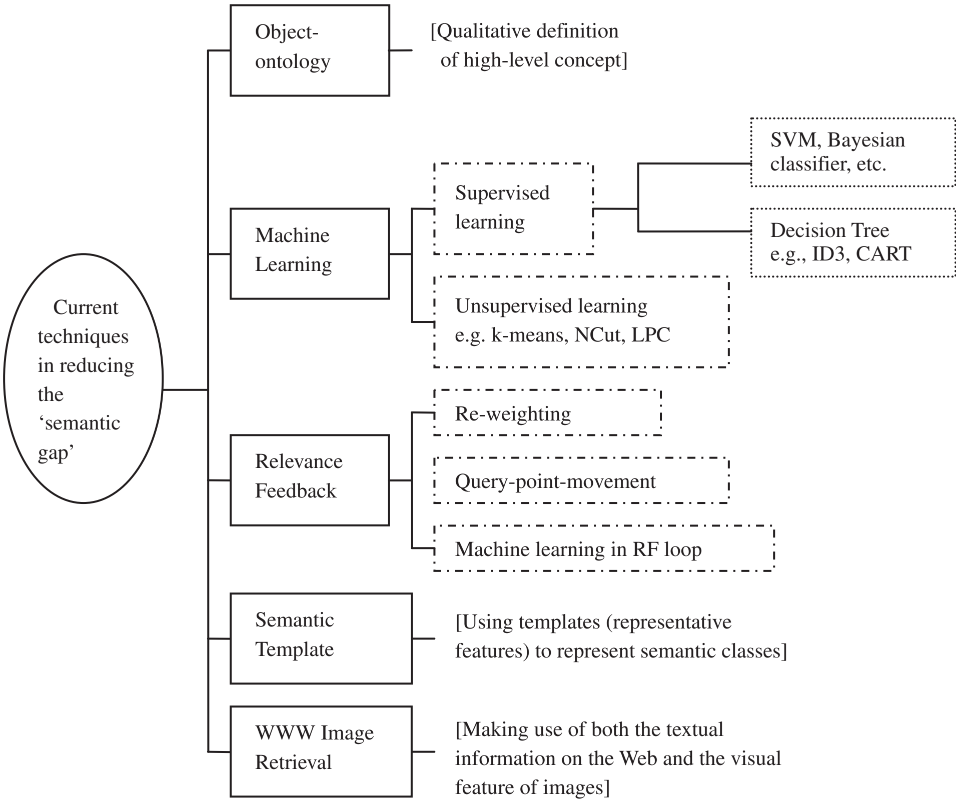

Current techniques in reducing the 'semantic gap'
Object-ontology
[Qualitative definition of high-level concept]
Machine Learning
Supervised learning
SVM, Bayesian classifier, etc.
Decision Tree e.g., ID3, CART
Unsupervised learning e.g. k-means, NCut, LPC
Relevance Feedback
Re-weighting
Query-point-movement
Machine learning in RF loop
Semantic Template
[Using templates (representative features) to represent semantic classes]
WWW Image Retrieval
[Making use of both the textual information on the Web and the visual feature of images]

Fig. 9. Summary of the current techniques in reducing the 'semantic gap'.

of concepts for retrieval purpose. The system exploits the evidences from both the HTML text and visual features of images and develops two independent classifiers based on text and visual image features, respectively. The experimental results using a pre-defined set of 15 concepts demonstrate the substantial performance of the system. However, due to the inaccuracy in textural information extraction, the performance for certain concepts is not satisfied.

MSRA (Microsoft Research Asia) is developing a promising system for Web image retrieval [63,64]. The purpose is to cluster the search results of conventional Web image search engines, so that users can find the desired images quickly. Firstly, a intelligent vision-based segmentation algorithm is designed to segment a web-page into blocks. From the block containing the image, the textual and link information of an image can be accurately extracted. Then an image graph is constructed by using block-level link analysis techniques. Hence for each image, three types of representations are obtained, visual feature-based representation, textual feature-based representation and graph-based representation. Initial experimental results show that by combining textual and graph-based representation for image clustering, the system can reveal the semantic structure of the Web images. The search results are clustered into different semantic categories. For each category, several images were selected as representative images, so that the user can quickly understand the main topics of the search results. The images in each category are then reorganized based on their visual features to make the cluster more visually desirable to users. A thorough experimental evaluation needs to be carried out to investigate the robustness of the technique.

3.6. Summary

We have identified five major categories of current techniques used in reducing the 'semantic gap' as summarized in Fig. 9. Ontology-based algorithms are easy to design and are suitable to applications with simple semantic features. However, in most cases, machine learning techniques are required to learn more complex semantics. Due to its simplicity in implementation and the intuitive mapping from low-level features to high-level concepts using decision rules, decision tree is a promising tool for image retrieval if the learning problem can be well modelled. RF has been proved to be effective in boosting image retrieval accuracy. The problem is that most current systems requires about five or even more iterations before it converges to a stable performance level, but users are usually impatient and may give up after two or three tries [16,35,69,109]. Using ST to support image retrieval seems to be a practical and promising way to reduce the 'semantic gap'. Web image retrieval is an active research area, and we look forward to a practical product to be delivered in the near future. Many systems combine one or more of these techniques to implement semantic-based image retrieval. For example, RF is often combined with object-ontology, and machine learning [27,31,70], Web image retrieval systems usually employ one or more of the other four types of techniques [63,72] to derive semantic features.

Besides the major techniques discussed above, there are some other interesting works. For example, in Ref. [34], based on statistical parameters derived from some testing data, the database images are classified into semantic categories, such as texture and non-texture, graph and photograph. In Ref. [113], an image is spectrally separated into different layers, each retaining only pixels in areas with similar 'busyness'. In this way, it associates color features with perceptual meanings. For example, a flat area is very possible to be associated with backgrounds or interior of an object and a busy area may be associated with textured surfaces or object boundaries. The algorithm in Ref. [114] attempts to relate human perception to low-level image features by recognizing the central object of an image as the region with significant color distribution. This is based on the assumption that people tend to locate the most interesting object at the center of the frame when they take a picture.

4. Image database and performance evaluation

There are so far no standard test data and PE model for CBIR systems.

4.1. Image databases

In the surveyed papers, more than half of systems use a subset of Corel image dataset [115] to test retrieval performance, others use either self-collected images or other image sets such as LA resource pictures [116], Kodak database of consumer images [77]. Brodatz textures [45] are widely used in perceptual texture feature studies [30,44,67]. Images collected from Internet serve as another data source especially for systems targeting at Web image retrieval [24,63].

Many researchers tend to use natural scenery images as test bed for semantic extraction as such images are easier to analyse than other images. The reasons are tow-fold. Firstly, the types of objects are limited. Main scenery object types include sky, tree, building, mountain, grass, water, and snow, etc. Secondly, compared with other features of image regions, shape features are less important in analysing scenery images than in other images. Thus we can avoid our weakness in extracting high-level semantics from shape features due to segmentation inaccuracy [50].

Corel image database contains a large amount of images of various contents ranging from animals and outdoor sports to natural sceneries. These images are pre-classified into different categories of size 100 by domain professionals. Some researchers think that Corel image dataset meets all the requirements to evaluate an image retrieval system, because of its large size, heterogeneous content and human annotated ground truth available [101]. But some other researchers consider Corel image database not suitable for CBIR performance evaluation because the associated ground truth (category labels) are often too high-level to be useful in performance analysis [117,118]. Although it is still controversial about Corel images dataset is suitable for CBIR performance evaluation or not, it is so far the most widely used.

In our opinion, Corel image database is good in its large size and various contents available. However, to be used for CBIR performance evaluation, some pre-processing work is necessary for the following two reasons: (1) some images with similar content are divided into different categories. For examples, the images in 'Ballon1' and 'Ballon2' are actually in the same category, same for category 'Cuisine' and 'Cuisines'; (2) some 'category labels' are very abstract and the images within the same category can be largely varied in content. For instance, the category 'Australia' includes pictures of city building, crowds in street, Australian wild animals, etc. Fig. 10 gives a few examples. It is very difficult to measure image similarities within such groups.

Hence, it is appropriate to select a subset of these images as ground truth, or to make some necessary changes in setting group truth data.

Considering the above mentioned problems in Corel image database, in Ref. [119], a new reference data set is presented for evaluating image retrieval algorithms. The authors have collected a large data set of human evaluations of retrieval results, both for query by image example and query by text. The data domain is 16,000 images from the Corel data set. Totally 20,000 query-result pairs were evaluated for query by example image, and 5000 pairs for query by text. The data is claimed to be independent of any particular image retrieval algorithm and can be used to compare many algorithms without further data collection. The data and calibration software are available online at http://kobus.ca/research/data.

For video retrieval, standard test data is available from TREC video retrieval evaluation (TRECVID). The TREC conference series is sponsored mainly by National Institute of Standards and Technology (NIST) to encourage research in information retrieval by providing large test collection, uniform scoring procedures and a forum for organizations interested in comparing their results. In 2001 and 2002, a video 'track' is sponsored for research in automatic segmentation, indexing and content-based retrieval of digital video. From 2003, this track became an independent evaluation workshop two days before TREC conference.

4.2. Vocabulary

To find an 'ideal' vocabulary representing the rich semantics of images is not an easy task. In Ref. [120], psychophysical experiments are conducted to gain insight into the semantic categories that guide the human perception of

Fig. 10. Example corel images from category ‘Australia’.

image similarity. By analysing the perceptual data, the most important 20 semantic categories (for example, portraits, crowds, cityscapes) in the perception of image similarity were established. Then 40 low-level features were discovered that best describe each category, such as number of regions, color composition, number of edges, and the presence of central object. In Ref. [121], the authors establish a so-called ‘lexical basis functions’ which contains 98 words to represent images. In Ref. [112], a ‘WordNet’ on-line lexical reference system is described. ‘WordNet’ organizes English words into synonym sets, each representing one underlying lexical concepts. It is a ‘dictionary’ based on psycholinguistic principles so that searching can be done conceptually instead of alphabetically.

The primary criterion in choosing a set of categories is to ensure that they are sufficiently well-defined in terms of the image descriptors and yet general enough to give meaningful semantic associations [28]. The vocabulary used in a system depends mainly on the image data set used. For natural scenery images, usually the images are classified into about 10–20 categories including water, sky, tree, sand, grass, mountain, snow, etc. For example, in Ref. [28], 11 categories are chosen: brick, cloud, fur, grass, ice, road, rock, sand, skin, tree, and water. In Ref. [116], 10 semantic categories are defined: beach, building, Disneyland, desert, mountain, freeway, downtown, park, people, and unknown. In Ref. [50], the authors discuss the identification of six high-level scenery features: sky, building, tree, water wave, placid water, and ground.

However, for real-world image database retrieval, such small vocabulary is far from enough. It is believed that humans can recognize about 5000–30,000 object categories. Category learning with such large vocabulary is very difficult and much work still remains to be done in this area. In Ref. [98], an incremental Bayesian algorithm is developed to recognize 101 object categories. To our knowledge, this is so far the largest vocabulary set used in object recognition.

4.3. Performance evaluation

Usually precision and recall are used in CBIR system to measure retrieval performance. Precision (Pr) is defined as the ratio of the number of relevant images retrieved (N_r) to the number of total retrieved images K. Recall (Re) is defined as the number of retrieved relevant images N_r over the total number of relevant images available in the database N_t

$$\mathrm{Re} = N_r/N_t, \quad \mathrm{Pr} = N_r/K. \tag{5}$$

It is ‘ideal’ to have both high Pr and Re. Therefore, instead of using Pr or Re individually, usually a joint Pr(Re) curve is used to characterize the performance of image retrieval system [101].

As recall is often low in color image retrieval system, Pr(Re) curve is less meaningful than it is in text-based retrieval systems. Many researchers are adopting precision-scope curve to evaluate image retrieval performance [122]. Scope(Sc) specifies the number of images returned to the user, that is K in Eq. (5). For a particular scope Sc, Pr(Sc) can be computed as

$$\mathrm{Pr}(Sc) = N_r/Sc. \tag{6}$$

Another performance measure used is the rank (Ra) measure [122–124]. The rank measure is defined as the average rank of the retrieved images. It is clear that the smaller the rank, the better the performance.

While Pr(Sc) only cares if a relevant image is retrieved or not, $Ra(Sc)$ also cares the rank of the retrieved image. Suppose there are two retrieval systems, ‘system1’ and ‘system2’. If $\mathrm{Pr}_1(Sc) > \mathrm{Pr}_2(Sc)$ and $Ra_1(Sc) < Ra_2(Sc)$, then definitely ‘system1’ is better than ‘system2’. However, if $\mathrm{Pr}_1(Sc) > \mathrm{Pr}_2(Sc)$ and $Ra_1(Sc) > Ra_2(Sc)$, we cannot tell which system is better.

5. Research issues

Most of the current image retrieval systems focus on improving the accuracy of retrieval. From system point of view, there are some other issues to be further studied.

5.1. Query language design

Query mechanisms play an important role in bridging the ‘semantic gap’. A specialized query language designed for CBIR could provide a means of addressing many of the problems associated with conventional query paradigms such as query-by-example and query-by-sketch. However, there has little recent work addressing this issue [125].

In Ref. [125], the authors argue that "query languages constitute an important avenue for further work in developing CBIR query mechanisms." They design a retrieval language—the OQUEL query language. The retrieval process takes place entirely within the ontology domain and is defined by the syntax and semantics of the query. The format of text queries is highly flexible as the system does not reply on the pre-annotation of images. The vocabulary has 400 words relating to the semantic descriptors (assigned to segmented regions on the basis of low-level features) including synonyms obtained by WordNet [112]. Query example, "some green colored vegetation in the center which is of similar size as blue sky at the top." The OQUEL language supports queries with either simple keyword phrases or complex compound.

In Ref. [51], a natural query language is designed for querying image databases. The vocabulary of the query language is based on the concept of 'semantic indicators' (elementary semantic categories, such as sky, flower), while the syntax captures the basic patterns in human perception of semantic categories (such as 'crowds', 'outdoor scenes') [51]. The language is claimed to be simple yet expressive. It is simple as the words of the language are almost limited to the names of the semantic indicators which are often described with a single word (e.g., snow, mountain). These words can be used to construct sentence expressing an assertion about the image. For instance, "the number of skin regions is greater than 5". During retrieval process, all the database images are tested against the query and only those satisfying the assertion are selected.

In Ref. [126], the authors use sub-image to represent the semantic content of the query in a Search and Retrieve Web (SRW) service for searching databases containing metadata and objects. The semantic content is captured using the multiscale color coherent vector and the texture features computed from wavelet decomposition. The user can use the sub-image query to express 'find a picture with person or object like this', 'find a painting with this class of cracks', etc.

Compared with the other methods in reducing the 'semantic gap', query language is relatively ill-understood and deserves greater attention [125].

5.2. High-dimensional indexing of image features

As the size of image database is increasing rapidly, retrieval speed will be an important factor to be concerned. Hence off-line multi-dimensional image data indexing is more and more necessary. Among the surveyed papers, only a few include multi-dimensional feature indexing as an integrated part of their CBIR systems. For example, in Ref. [127], a k-means clustering algorithm [128] is used to cluster regions according to their features. In Ref. [129], R-tree [130] is used to index MBR (maximum bounding box) of regions.

As the dimensionality of image features are usually high (up to tens or hundreds), traditional indexing algorithms such as k-d-b tree [131], quad-tree [132], and R-tree [130] are not suitable for image feature space indexing, due to the well-known 'curse of dimensionality' problem [133]. That is, the performance of these indexing algorithms degrades as the dimensionality of feature space increases. It is reported that when the dimensionality is above 10, the performance is no better than a simple sequential scan [134]. To relieve this problem, high-dimensional indexing algorithms such as X-tree [135], VA-file [134], and i-Distance [136] have been introduced. However, such algorithms focus only on how to index but not what to index. That is, they are designed without considering the specific properties of image features.

Some effort has been made in designing indexing algorithms specifically for image database. For example, in Ref. [137], a prototype image database system is implemented—the FIDS (Flexible Image Database System) system. In this system, the bare-bones triangle inequality algorithm is used to index image data and to sharply reduce the number of images needed to be directly compared to a query image for a given distance measure. FIDS system allows user great flexibility in run-time to find similar images using complex combinations of many pre-defined distance measures. In Ref. [138], a RBIR system using index is designed. In this system, the regions in the database images are indexed using an algorithm named A_0^{WS} to speed up the evaluation of k-nearest neighbor queries. This algorithm computes the optimal matching between regions in the query image and regions in a database image, so as to maximize the overall similarity score between images.

Further work is still to be done in efficient high-dimensional image feature indexing for real-world image database retrieval.

5.3. Standard DBMS extended for image retrieval

In many image retrieval systems such as Photobook [5], the data and features are typically stored in files addressed by names. When trying to scale up to a large database and a large number of users, this approach is likely to run into data integrity and performance problems. It is clear that when large image database come into view, the connection between CBIR and database management system (DBMS) is inevitable.

QBIC [4] and Virage [6] systems have taken one step beyond the read-only database and extended standard DBMS for image retrieval. In Ref. [139], a relational database system POSTGRES is used for storing and managing digital images and their associated textual data.

Making image retrieval as a plug-in module in an existing DBMS not only solves the image data integrity problem and allows dynamic updates, but also provides natural integration with features derived from other sources [15]. A truly integrated CBIR system would require the integration of content-based similarity, interaction with users, visualization of image database, database management for retrieval relevant images, etc. [15].

5.4. Standard image testbed and performance evaluation model

Though many researchers choose to use Corel images as test data to evaluate their CBIR systems, there is so far no standard test bed and different subsets of Corel images are used in different systems for performance evaluation. In addition, though precision and recall are often used to measure retrieval performance, the queries performed by different researchers are usually different. Hence, it is hard to compare the performance of different CBIR systems.

In Ref. [118], using same subset of Corel images and the same set of performance measures, the authors evaluate the retrieval performance of same CBIR system in different ways, by submitting different query images and by setting different ground truth data. The results show that it is very easy to get different retrieval performance, even with the same image collection, the same CBIR system and the same performance measures. It demonstrated that it is impossible to objectively compare the performances of different CBIR systems unless it is clearly stated which images were used as test data, which were used as queries, and which parameters have been used to measure performance.

Hence, a standard image database with a query set and corresponding performance measure model is highly in need for objective performance evaluation of CBIR systems.

6. Conclusions

Research in content-based image retrieval (CBIR) in the past has been focused on image processing, low-level feature extraction, etc. Extensive experiments on CBIR systems demonstrate that low-level image features cannot always describe high-level semantic concepts in the users' mind. It is believed that CBIR systems should provide maximum support in bridging the 'semantic gap' between low-level visual features and the richness of human semantics.

This paper provides a comprehensive survey of recent work towards narrowing down the 'semantic gap'. We have identified five major categories of state-of-the-art techniques: (1) using object ontology to define high-level concepts; (2) using supervised or unsupervised machine learning methods to associate low-level features with query concepts; (3) introducing relevance feedback into retrieval loop for continuous learning of users' intention; (4) generating semantic template to support high-level image retrieval; (5) making use of the textual information on the Web and the visual content of images for WWW image retrieval. We observe that though significant amount of work has been done in this area, there is so far no generic approach for high-level semantic-based image retrieval. In addition, current systems focus on retrieval at Level 2, and there is yet no good solution for Level 3 retrieval.

Focusing on the differences between CBIR with high-level semantics and traditional systems with low-level features, this paper also provides useful insights into how to obtain salient low-level features to facilitate 'semantic gap' reduction. In addition, current techniques in image similarity measure are described. As conventional Minkowski metric-based similarity measure cannot effectively model human perception, perceptual image similarity measure is to be further studied. Test dataset and performance evaluation of CBIR systems are also discussed. We believe that establishing a standard test set and evaluation model is necessary for objective performance comparison.

Based on the current technologies available and the demand from practical applications, a few open issues are identified from system point of view, including query-language design, integration of image retrieval with database management system, high-dimensional image feature indexing, etc.

To implement a full-fledged image retrieval system with high-level semantics requires the integration of salient low-level feature extraction, effective learning of high-level sematics, friendly user inferface, and efficeint indexing tool. Most systems understandably limit their contributions to one or two of these components. A CBIR framework providing a more balanced view of all the constituent components is in need.

Acknowledgement

The authors are grateful for the many specific and valuable comments made by the reviewer.

References

[1] J. Eakins, M. Graham, Content-based image retrieval, Technical Report, University of Northumbria at Newcastle, 1999.

[2] I.K. Sethi, I.L. Coman, Mining association rules between low-level image features and high-level concepts, Proceedings of the SPIE Data Mining and Knowledge Discovery, vol. III, 2001, pp. 279–290.

[3] S.K. Chang, S.H. Liu, Picture indexing and abstraction techniques for pictorial databases, IEEE Trans. Pattern Anal. Mach. Intell. 6 (4) (1984) 475–483.

[4] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, W. Equitz, Efficient and effective querying by image content, J. Intell. Inf. Syst. 3 (3–4) (1994) 231–262.

[5] A. Pentland, R.W. Picard, S. Scaroff, Photobook: content-based manipulation for image databases, Int. J. Comput. Vision 18 (3) (1996) 233–254.

[6] A. Gupta, R. Jain, Visual information retrieval, Commun. ACM 40 (5) (1997) 70–79.

[7] J.R. Smith, S.F. Chang, VisualSeek: a fully automatic content-based query system, Proceedings of the Fourth ACM International Conference on Multimedia, 1996, pp. 87–98.

[8] W.Y. Ma, B. Manjunath, Netra: a toolbox for navigating large image databases, Proceedings of the IEEE International Conference on Image Processing, 1997, pp. 568–571.

[9] J.Z. Wang, J. Li, G. Wiederhold, SIMPLIcity: semantics-sensitive integrated matching for picture libraries, IEEE Trans. Pattern Anal. Mach. Intell. 23 (9) (2001) 947–963.

[10] F. Long, H.J. Zhang, D.D. Feng, Fundamentals of content-based image retrieval, in: D. Feng (Ed.), Multimedia Information Retrieval and Management, Springer, Berlin, 2003.

[11] Y. Rui, T.S. Huang, S.-F. Chang, Image retrieval: current techniques, promising directions, and open issues, J. Visual Commun. Image Representation 10 (4) (1999) 39–62.

[12] A. Mojsilovic, B. Rogowitz, Capturing image semantics with low-level descriptors, Proceedings of the ICIP, September 2001, pp. 18–21.

[13] X.S. Zhou, T.S. Huang, CBIR: from low-level features to high-level semantics, Proceedings of the SPIE, Image and Video Communication and Processing, San Jose, CA, vol. 3974, January 2000, pp. 426–431.

[14] Y. Chen, J.Z. Wang, R.Krovetz, An unsupervised learning approach to content-based image retrieval, IEEE Proceedings of the International Symposium on Signal Processing and its Applications, July 2003, pp. 197–200.

[15] A.W.M. Smeulders, M. Worring, A. Gupta, R. Jain, Content-based image retrieval at the end of the early years, IEEE Trans. Pattern Anal. Mach. Intell. 22 (12) (2000) 1349–1380.

[16] F. Jing, M. Li, L. Zhang, H.-J. Zhang, B. Zhang, Learning in region-based image retrieval, Proceedings of the International Conference on Image and Video Retrieval (CIVR2003), 2003, pp. 206–215.

[17] H. Feng, D.A. Castanon, W.C. Karl, A curve evolution approach for image segmentation using adaptive flows, Proceedings of the International Conference on Computer Vision (ICCV'01), 2001, pp. 494–499.

[18] W.Y. Ma, B.S. Majunath, Edge flow: a framework of boundary detection and image segmentation, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1997, pp. 744–749.

[19] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. (PAMI) 22 (8) (2000) 888–905.

[20] D. Comaniciu, P. Meer, Robust analysis of feature spaces: color image segmentation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1997, pp. 750–755.

[21] P.L. Stanchev, D. Green Jr., B. Dimitrov, High level color similarity retrieval, Int. J. Inf. Theories Appl. 10 (3) (2003) 363–369.

[22] K.A. Hua, K. Vu, J.-H. Oh, SamMatch: a flexible and efficient sampling-based image retrieval technique for large image databases, Proceedings of the Seventh ACM International Multimedia Conference (ACM Multimedia'99), November 1999, pp. 225–234.

[23] Y. Deng, B.S. Manjunath, Unsupervised segmentation of color-texture regions in images and video, IEEE Trans. Pattern Anal. Mach. Learn. (PAMI) 23 (8) (2001) 800–810.

[24] H. Feng, T.-S. Chua, A boostrapping approach to annotating large image collection, Workshop on Multimedia Information Retrieval in ACM Multimedia, November 2003, pp. 55–62.

[25] Y. Liu, D.S. Zhang, G. Lu, W.-Y. Ma, Region-based image retrieval with perceptual colors, Proceedings of the Pacific-Rim Multimedia Conference (PCM), December 2004, pp. 931–938.

[26] C. Carson, S. Belongie, H. Greenspan, J. Malik, Blobworld: image segmentation using expectation-maximization and its application to image querying, IEEE Trans. Pattern Anal. Mach. Intell. 8 (8) (2002) 1026–1038.

[27] R. Shi, H. Feng, T.-S. Chua, C.-H. Lee, An adaptive image content representation and segmentation approach to automatic image annotation, International Conference on Image and Video Retrieval (CIVR), 2004, pp. 545–554.

[28] C.P. Town, D. Sinclair, Content-based image retrieval using semantic visual categories, Society for Manufacturing Engineers, Technical Report MV01-211, 2001.

[29] J.R. Smith, C.-S. Li, Decoding image semantics using composite region templates, IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL-98), June 1998, pp. 9–13.

[30] W.K. Leow, S.Y. Lai, Scale and orientation-invariant texture matching for image retrieval, in: M.K. Pietikainen (Ed.), Texture Analysis in Machine Vision, World Scientific, Singapore, 2000.

[31] V. Mezaris, I. Kompatsiaris, M.G. Strintzis, An ontology approach to object-based image retrieval, Proceedings of the ICIP, vol. II, 2003, pp. 511–514.

[32] K.N. Plataniotis, A.N. Venetsanopoulos, Color Image Processing and Applications, Springer, Berlin, 2000.

[33] B.S. Manjunath, et al., Color and texture descriptors, IEEE Trans. CSVT 11 (6) (2001) 703–715.

[34] J.Z. Wang, J. Li, D. Chan, G. Wiederhold, Semantics-sensitive retrieval for digital picture libraries, Digital Library Magazine, vol. 5(11), 1999.

[35] E. Chang, S. Tong, SVM$_{\text{active}}$-support vector machine active learning for image retrieval, Proceedings of the ACM International Multimedia Conference, October 2001, pp. 107–118.

[36] X. Zheng, D. Cai, X. He, W.-Y. Ma, X. Lin, Locality preserving clustering for image database, Proceedings of the 12th ACM Multimedia, October 2004.

[37] B.S. Manjunath, et al., Introduction to MPEG-7, Wiley, New York, 2002.

[38] T. Gevers, A. Smeulders, Content-based image retrieval by viewpoint-invariant color indexing, Image Vision Comput. 17 (1999) 475–488.

[39] W. Wang, Y. Song, A. Zhang, Semantics retrieval by content and context of image regions, Proceedings of the 15th International Conference on Vision Interface (VI'2002), May 2002, pp. 17–24.

[40] K.N. Plataniotis, et al., Adaptive fuzzy systems for multichannel signal processing, Proc. IEEE 87 (9) (1999) 1601–1622.

[41] R. Lukac, et al., Vector filtering for color imaging, IEEE Signal Process. Mag. (2005) 74–86.

[42] P. Stanchev, Using image mining for image retrieval, IASTED Conference "Computer Science and Technology," Cancun, Mexico, May 2003, pp. 214–218.

[43] H. Tamura, S. Mori, T. Yamawaki, Texture features corresponding to visual perception, IEEE Trans. Syst. Man Cybern. 8 (6) (1978) 460–473.

[44] F. Liu, R.W. Picard, Periodicity, directionality, and randomness: wold features for image modeling and retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 18 (7) (1996) 722–733.

[45] P. Brodatz, Textures, A Photographic Album for Artists & Designers, Dover, New York, NY, 1966.

[46] Y. Liu, X. Zhou, W.Y. Ma, Extraction of texture features from arbitrary-shaped regions for image retrieval, International Conference on Multimedia and Expo (ICME04), Taipei, June 2004, pp. 1891–1894.

[47] P.W. Huang, S.K. Dai, Image retrieval by texture similarity, Pattern Recognition 36 (2003) 665–679.

[48] R. Mehrotra, J.E. Gary, Similar-shape retrieval in shape data management, IEEE Comput. 28 (9) (1995) 57–62.

[49] F. Mokhtarian, S. Abbasi, Shape similarity retrieval under affine transforms, Pattern Recognition 35 (2002) 31–41.

[50] Y. Song, W. Wang, A. Zhang, Automatic annotation and retrieval of images, J. World Wide Web 6 (2) (2003) 209–231.

[51] A. Mojsilovic, B. Rogowitz, ISee: perceptual features for image library navigation, Proceedings of the SPIE, Human Vision and Electronic Imaging, vol. 4662, 2002, pp. 266–277.

[52] S.K. Chang, Q.Y. Shi, C.W. Yan, Iconic indexing by 2D string, IEEE Trans. Pattern Anal. Mach. Intell. 9 (3) (1987) 413–428.

[53] W. Ren, M. Singh, C. Singh, Image retrieval using spatial context, Ninth International Workshop on Systems, Signals and Image Processing (IWSSIP'02), Manchester, November, 2002.

[54] D. Androutsos, K.N. Plataniotis, A.N. Venetsanopoulos, Distance measures for color image retrieval, Proceedings of the International Conference on Image Processing, vol. 2, 1998, pp. 770–774.

[55] Z. Chen, B. Zhu, Some formal analysis of Rocchio's similarity-based relevance feedback algorithm, Inf. Retr. 5 (2002) 61–86.

[56] S. Ardizzoni, I. Bartolini, M. Patella, Windsurf: region-based image retrieval using wavelets, 10th International Workshop on Database and Expert Systems Applications, Florence, Italy, 1999, pp. 167–173.

[57] Y. Rubner, C. Tomasi, L. Guibas, A metric for distributions with applications to image databases, Proceedings of the IEEE International Conference on Computer Vision (ICCV'98), January 1998, pp. 59–67.

[58] B. Li, E. Chang, C.-T. Wu, DPF-a perceptual function for image retrieval, Proceedings of the International Conference on Image Processing (ICIP), vol. II, September 2002, pp. 597–600.

[59] S. Berretti, A.D. Bimbo, P. Pala, Retrieval by shape similarity with perceptual distance and effective indexing, IEEE Trans. Multimedia 2 (4) (2000) 225–239.

[60] N. Vasconcelos, A. Lippman, A multiresolution manifold distance for invariant image similarity, IEEE Trans. Multimedia 7 (1) (2005) 127–142.

[61] A. Kushki, et al., Retrieval of image from artistic repositories using a decision fusion framework, IEEE Trans. Image Process. 13 (3) (2004) 277–289.

[62] A. Kushki, et al., Query feedback for interactive image retrieval, IEEE Trans. CSVT 14 (5) (2004) 644–655.

[63] D. Cai, X. He, Z. Li, W.-Y. Ma, J.-R. Wen, Hierachical clustering of WWW image search results using visual, textual and link information, Proceedings of the ACM International Conference on Multimedia, 2004.

[64] D. Cai, X. He, W.-Y. Ma, J.-R. Wen, H. Zhang, Organizing WWW images based on the analysis of page layout and web link structure, Proceedings of the International Conference on Multimedia and Expo (ICME), Taipei, 2004.

[65] J. Ren, Y. Shen, L. Guo, A novel image retrieval based on representative colors, Proceedings of the Image and Vision Computing, N.Z., November 2003, pp. 102–107.

[66] S. Kulkarni, B. Verma, Fuzzy logic for texture queries in CBIR, Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA), Xi'an, China, 2003, pp. 223–226.

[67] C.-Y. Chiu, H.-C. Lin, S.-N. Yang, Texture retrieval with linguistic descriptors, IEEE Pacific Rim Conference on Multimedia, 2001, pp. 308–315.

[68] A. Vailaya, M.A.T. Figueiredo, A.K. Jain, H.J. Zhang, Image classification for content-based indexing, IEEE Trans. Image Process. 10 (1) (2001) 117–130.

[69] L. Zhang, F. Liu, B. Zhang, Support vector machine learning for image retrieval, International Conference on Image Processing, October 2001, pp. 7–10.

[70] Y. Zhuang, X. Liu, Y. Pan, Apply semantic template to support content-based image retrieval, Proceedings of the SPIE, Storage and Retrieval for Media Databases, vol. 3972, December 1999, pp. 442–449.

[71] W. Chang, J. Wang, Metadata for multi-level content-based retrieval, Third IEEE Meta-Data Conference, April 1999.

[72] H. Feng, R. Shi, T.-S. Chua, A bootstrapping framework for annotating and retrieving WWW images, Proceedings of the ACM International Conference on Multimedia, 2004.

[73] M. Obeid, B. Jedynak, M. Daoudi, Image indexing and retrieval using intermediate features, Proceedings of the Ninth ACM International Conference on Multimedia, Ottawa, Canada, 2001, pp. 531–533.

[74] D.M. Conway, An experimental comparison of three natural language color naming models, Proceedings of the East–West International Conference on Human-Computer Interactions, St. Petersburg, Russia, 1992, pp. 328–339.

[75] T. Berk, L. Brownston, A. Kaufman, A new color-naming system for graphics language, IEEE Comput. Graphics Appl. 2 (3) (1982) 37–44.

[76] A.R. Rao, G.L. Lohse, Towards a texture naming system: identifying relevant dimensions of texture, IEEE Proceedings of the Fourth Conference on Visualization, 1993, pp. 220–227.

[77] J. Luo, A. Savakis, Indoor vs outdoor classification of consumer photographs using low-level and semantic features, International Conference on Image Processing (ICIP), vol II, October 2001, pp. 745–748.

[78] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, New York, 2001.

[79] V.N. Vapnik, Statistical Learning Theory, Wiley, New York, 1998.

[80] W. Jin, R. Shi, T.-S. Chua, A semi-naı¨ve bayesian method incorporating clustering with pair-wise constraints for auto image annotation, Proceedings of the ACM Multimedia, 2004.

[81] S. Tong, E. Chang, Support vector machine active learning for image retrieval, Proceedings of the ACM International Conference on Multimedia, Ottawa, Canada, 2001, pp. 107–118.

[82] N. Vasconcelos, A. Lippman, Library-based coding: a representation for efficient video compression and retrieval, Proceedings of the Data Compression Conference (DCC97), March 1997, pp. 121–130.

[83] S.D. MacArthur, C.E. Brodley, C.-R. Shyu, Relevance feedback decision trees in content-based image retrieval, Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL'00), June 2000, pp. 68–72.

[84] W.-C. Low, T.-S. Chua, Color-based relevance feedback for image retrieval, Proceedings of the International Workshop on Multimedia DBMS (IM-MMDBMS'98), August 1998, pp. 116–123.

[85] M. Pal, P.M. Mather, Decision tree based classification of remotely sensed data, Proceedings of the 22nd Asian Conference on Remote Sensing (ACRS), Singapore, vol. 1, November 2001, pp. 245–248.

[86] L.O. Hall, N. Chawla, K.W. Bowyer, Decision tree learning on very large data sets, IEEE International Conference on System, Man and Cybernetics (SMC) 1998, pp. 187–222.

[87] J.R. Quanlan, Induction of decision tree, Machine Learning, vol. 1, Kluwer Acedemic Publisher, Boston, 1986, pp. 81–106.

[88] U.M. Fayyad, K.B. Irani, Multi-interval discretization of continuous-valued attributes for classification learning, Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI), France, 1993, pp. 1022–1027.

[89] U.M. Fayyad, K.B. Irani, On the handling of continuous-valued attributes in decision tree generation, Mach. Learn. 8 (1992) 87–102.

[90] D. Stan, I.K. Sethi, Mapping low-level image features to semantic concepts, Proceedings of the SPIE: Storage and Retrieval for Media Databases, 2001, pp. 172–179.

[91] M. Bilenko, S. Basu, R.J. Mooney, Integrating constraints and metric learning in semi-supervised clustering, Proceedings of the 21st International Conference on Machine Learning (ICML), July 2004, pp. 81–88.

[92] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, Advances in Neural Information Processing Systems, vol. 14, MIT Press, Cambridge, MA, 2002.

[93] N. Vasconcelos, The design of end-to-end optimal image retrieval systems, in: Proceedings of the International Conference on ANN, Istanbul, Turkey, 2003.

[94] N. Vasconcelos, On the efficient evaluation of probabilistic similarity functions for image retrieval, IEEE Trans. Inf. Theory 50 (7) (2004) 1482–1496.

[95] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, Proceedings of the Computer Vision and Pattern Recognition, 2003.

[96] Y. Li, J. Bilmes, L.G. Shapiro, Object class recognition using images of abstract regions, International Conference on Pattern Recognition, August 2004.

[97] Y. Li, L.G. Shapiro, J. Bilmes, A generative/discriminative learning algorithm for image classification, International Conference on Computer Vision, October 2005.

[98] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories, Computer Vision and Pattern Recognition, Workshop on Generative-Model Based Vision, 2004.

[99] G. Salton, Automatic Text Processing, Addison-Wesley, Reading, MA, 1989.

[100] Y. Rui, T.S. Huang, M. Ortega, S. Mehrotra, Relevance feedback: a power tool for interactive content-based image retrieval, IEEE Trans. Circuits Video Technol. 8 (5) (1998) 644–655.

[101] Y. Rui, T.S. Huang, Optimizing learning in image retrieval, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, June 2000, pp. 1236–1243.

[102] X.S. Zhu, T.S. Huang, Relevance feedback in image retrieval: a comprehensive review, Multimedia System 8 (6) (2003) 536–544.

[103] G.-D. Guo, A.K. Jain, W.-Y. Ma, H.-J. Zhang, Learning similarity measure for natural image retrieval with relevance feedback, IEEE Trans. Neural Networks 13 (4) (2002) 811–820.

[104] Y. Rui, T.S. Huang, S. Mehrotra, Content-based image retrieval with relevance feedback in Mars, Proceedings of the IEEE International Conference on Image Processing, 1997, pp. 815–818.

[105] Z. Chen, B. Zhu, On the complexity of Rocchio's similarity-based relevance feedback algorithm, ISAAC, 2005.

[106] F. Jing, et al., Relevance feedback in region-based image retrieval, IEEE Trans. CSVT 14 (5) (2004) 672–681.

[107] T.S. Huang, X.S. Zhou, M. Nakazato, Y. Wu, I. Cohen, Learning in content-based image retrieval, International Conference on Development and Learning (ICDL'02), 2002, pp. 155–162.

[108] Q. Tian, Y. Yu, T.S. Huang, Incorporate discriminant analysis with EM algorithm in image retrieval, Proceedings of the International Conference on Multimedia and Expo (ICME), 2000, pp. 299–302.

[109] Y. Lu, C. Hu, X. Zhu, H. Zhang, Q. Yang, A unified framework for semantics and feature based relevance feedback in image retrieval systems, ACM International Conference on Multimedia, 2000, pp. 31–37.

[110] A.D. Doulamis, N.D. Doulamis, Generalized nonlinear relevance feedback for iterative content-based retrieval and organization, IEEE Trans. CSVT 14 (5) (2004) 656–671.

[111] S.-F. Chang, W. Chen, H. Sundaram, Semantic visual templates: linking visual features to semantics, International Conference on Image Processing (ICIP), Workshop on Content Based Video Search and Retrieval, vol. 3, October 1998, pp. 531–534.

[112] G. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. Miller, Introduction to Wordnet: an on-line lexical database, Int. J. Lexicography 3 (1990) 235–244.

[113] G. Qiu, K.-M. Lam, Spectrally layered color indexing, Proceedings of the International Conference on Image and Video Retrieval (CIVR), 2002, pp. 100–107.

[114] S. Kim, S. Park, M. Kim, Central object extraction for object-based image retrieval, International Conference on Image and Video Retrieval (CIVR), 2003, pp. 39–49.

[115] ⟨http://www.corel.com⟩.

[116] Z. Yang, C.-C. Jay Kuo, Learning image similarities and categories from content analysis and relevance feedback, Proceedings of the ACM Multimedia Workshops, 2000, pp. 175–178.

[117] X.Y. Jin, CBIR: difficulty, challenge, and opportunity, Microsoft PPT, October 2002.

[118] H. Mueller, S. Marchand-Maillet, T. Pun, The truth about Corel-evaluation in image retrieval, Proceedings of the International Conference on Image and Video Retrieval (ICIVR), 2002, pp. 38–49.

[119] N.V. Shirahatti, K. Barnard, Evaluating image retrieval, Proceedings of the Computer Vision and Pattern Recognition (CVPR), San Diego, CA, vol. 1, June 2005, pp. 955–961.

[120] A. Mojsilovic, B. Rogowitz, Capturing image semantics with low-level descriptors, International Conference on Image Processing (ICIP), Greece, 2001, pp. 18–21.

[121] J.A. Black, K. Kahol, P. Kuchi, G. Fahmy, S. Panchanathan, Characterizing the high-level content of natural images using lexical basis functions, Proceedings of the SPIE, vol. 5007, Human Vision and Electronic Imaging, Santa Clara, 2003, pp. 378–391.

[122] J. Huang, S. Kuamr, M. Mitra, W.-J. Zhu, R. Zabih, Image indexing using color correlogram, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'97), 1997, pp. 762–765.

[123] R. Zhao, W.I. Grosky, From feature to semantics: some preliminary results, Proceedings of the International Conference on Multimedia and Expo (ICME), 2000, pp. 679–682.

[124] R. Zhao, W.I. Grosky, Negotiating the semantic gap: from feature maps to semantic landscapes, J. Pattern Recognition 35 (2002) 593–600.

[125] C.P. Town, D. Sinclair, Language-based querying of image collections on the basis of an extensible ontology, Int. J. Image Vision Comput. 22 (3) (2004) 251–267.

[126] P.H. Lewis, et al., An integrated content and metadata based retrieval system for art, IEEE Trans. Image Process. 13 (3) (2004) 302–313.

[127] J.Z. Wang, Y. Du, Scalable integrated region-based image retrieval using IRM and statistical clustering, IEEE Proceedings of the ACM and IEEE Joint Conference on Digital Libraries, Roanoke, VA, ACM, June 2001, pp. 268–277.

[128] J.A. Hartigan, M.A. Wong, Algorithm AS136: a k-means clustering algorithm, Appl. Stat. 28 (1) (1979) 100–108.

[129] Y. Chahir, L. Chen, Spatialized multi-visual features-based image retrieval, Int. J. Comput. Appl. 6 (4) (1999) 190–199.

[130] A. Guttman, R-tree: a dynamic index structure for spatial searching, Proceedings of the ACM SIGMOD International Conference on Management of Data, Boston, MA, 1984, pp. 47–57.

[131] J. Robinson, The k-d-b-tree: a search structure for large multidimensional dynamic indexes, Proceedings of the ACM SIGMOD International Conference on Management of Data, 1981, pp. 10–18.

[132] R. Finkel, J. Bentley, Quad-tree: a data structure for retrieval on composite keys, Acta Inf. 4 (1) (1974) 1–9.

[133] H. Yamamoto, H. Iwasa, N. Yokaya, H. Takemura, Content-based similarity retrieval of images based on spatial color distribution, 10th International Conference on Image Analysis and Processing, Venice, Italy, September 1999, pp. 951–956.

[134] R. Weber, H.-J. Schek, S. Blott, A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces, Proceedings of the 24th VLDB Conference, New York, USA, 1998, pp. 194–205.

[135] S. Berchtold, D.A. Keim, H.-P. Kriegel, The X-tree: an index structure for high-dimensional data, Proceedings of the 22nd VLDB Conference Mumbai (Bombay), India, 1996, pp. 28–39.

[136] C. Yu, B.C. Ooi, K.-L. Tan, H.V. Jagadish, Indexing the distance: an efficient method to KNN processing, Proceedings of the 27th VLDB Conference, Roma, Italy, 2001, pp. 421–430.

[137] A.P. Berman, L.G. Shapiro, Triangle-inequality-based pruning algorithms with triangle tries. Proceedings of the SPIE Conference on Storage and Retrieval for Image and Video Databases, January 1999.

[138] I. Bartolini, P. Ciaccia, M. Patella, A sound algorithm for region-based image retrieval using an index, International Workshop on Database and Expert Systems Applications (DEXA), 2000, pp. 930–934.

[139] V.E. Ogle, CHABOT-retrieval from a relational database of images, Computer 28 (9) (1995) 40–48.

About the Author—Ms. YING LIU received her B.Sc. and M.Sc. degree from Dept. of Infor. Eng. from Xidian University, China, in 1993 and 1996, respectively. Then she served as an associate lecturer in the same Dept. for 2 years. She received her M.Eng. degree in Dept. of E.E. from National University of Singapore in 2000. After this, she worked as a research Engineer in Center for Signal Processing, Nanyang Technological University in Singapore. Ms. Liu is now a Ph.D. candidate in Gippsland School of Computing and Information Technology, Monash University, Australia.

About the Author—Dr. DENGSHENG ZHANG received B.Sc. in Mathematics and B.A. in English in 1985 and 1987, respectively, both from China. He spent 12 years on teaching Mathematics and Computing before he was involved in his Ph.D. program in 1999. He received Ph.D. in Computer Technology from Monash University, Australia, in 2002. He is now a lecturer in Gippsland School of Computing and Information Technology of Monash University. Dr. Zhang has over 10 years research experience in the area of multimedia and has published over 20 referred international journal and conference papers.

About the Author—Dr. GUOJUN LU obtained his Ph.D. in 1990 from Loughborough University of Technology, and B.Eng. in 1984 from Nanjing Institute of Technology (now South East University). He is currently an associate professor at Gippsland School of Computing and Information Technology, Monash University, Australia. He has held positions in Loughborough University of Technology, National University of Singapore, and Deakin University. Dr. Lu's main research interests are in multimedia information indexing and retrieval, multimedia data compression, quality of service management, and multimedia compression. He has published over 50 technical papers in these areas and authored the books Communication and Computing for Distributed Multimedia Systems (Artech House, 1996), and Multimedia Database Management Systems (Artech House, to appear in 1999). He has over 10 years research experience in multimedia computing and communications.

About the Author—Dr. WEI-YING MA received his B.S. degree in E.E. from National Tsing Hua University in Taiwan in 1990, and his M.S. and Ph.D. degrees in E.C.E. from the University of California at Santa Barbara (UCSB) in 1994 and 1997, respectively. From 1994 to 1997 he was engaged in the Alexandria Digital Library (ADL) project in UCSB while completing his Ph.D. He developed the Netra system which is regarded as one of the most representative image retrieval systems. From 1997 to 2001, he was with HP Labs working in the field of multimedia adaptation and distributed media services infrastructure for mobile Internet. He joined Microsoft Research Asia in April 2001 as the Research Manager of the Web Search and Mining Group, leading the research in the areas of information retrieval, text mining, search, multimedia management, and mobile browsing. He currently serves as an Editor for the ACM/Springer Multimedia Systems Journal and Associate Editor for the Journal of Multimedia Tools and Applications published by Kluwer Academic Publishers. He has served on the organizing and program committees of many international conferences including ACM Multimedia, ACM SIGIR, CVPR, etc.

Inf Retrieval (2008) 11:77–107
DOI 10.1007/s10791-007-9039-3

Features for image retrieval: an experimental comparison

Thomas Deselaers · Daniel Keysers · Hermann Ney

Received: 19 May 2007 / Accepted: 28 November 2007 / Published online: 11 December 2007

Abstract An experimental comparison of a large number of different image descriptors for content-based image retrieval is presented. Many of the papers describing new techniques and descriptors for content-based image retrieval describe their newly proposed methods as most appropriate without giving an in-depth comparison with all methods that were proposed earlier. In this paper, we first give an overview of a large variety of features for content-based image retrieval and compare them quantitatively on four different tasks: stock photo retrieval, personal photo collection retrieval, building retrieval, and medical image retrieval. For the experiments, five different, publicly available image databases are used and the retrieval performance of the features is analyzed in detail. This allows for a direct comparison of all features considered in this work and furthermore will allow a comparison of newly proposed features to these in the future. Additionally, the correlation of the features is analyzed, which opens the way for a simple and intuitive method to find an initial set of suitable features for a new task. The article concludes with recommendations which features perform well for what type of data. Interestingly, the often used, but very simple, color histogram performs well in the comparison and thus can be recommended as a simple baseline for many applications.

Keywords Image retrieval · Features · Image classification · Quantitative comparison

T. Deselaers (✉) · H. Ney
Human Language Technology and Pattern Recognition, Computer Science Department,
RWTH Aachen University, Aachen, Germany
e-mail: deselaers@cs.rwth-aachen.de

H. Ney
e-mail: ney@cs.rwth-aachen.de

D. Keysers
Image Understanding and Pattern Recognition, German Research Center for Artificial Intelligence
(DFKI), Kaiserslautern, Germany
e-mail: daniel.keysers@dfki.de

1 Introduction

Image retrieval in general and content-based image retrieval (CBIR) in particular are well-known fields of research in information management in which a large number of methods have been proposed and investigated but in which still no satisfying general solutions exist. The need for adequate solutions is growing due to the increasing amount of digitally produced images in areas like journalism, medicine, and private life, requiring new ways of accessing images. For example, medical doctors have to access large amounts of images daily (Müller et al. 2004), home-users often have image databases of thousands of images (Sun et al. 2002), and journalists also need to search for images by various criteria (Markkula and Sormunen 1998; Armitage and Enser 1997). In the past, several CBIR systems have been proposed and all these systems have one thing in common: images are represented by numeric values, called features or descriptors, that are meant to represent the properties of the images to allow meaningful retrieval for the user.

Only recently have some standard benchmark databases and evaluation campaigns been created which allow for a quantitative comparison of CBIR systems. These benchmarks allow for the comparison of image retrieval systems under different aspects: usability and user interfaces, combination with text retrieval, or overall performance of a system. However, to our knowledge, *no quantitative comparison of the building blocks of the systems, the features that are used to compare images, has been presented so far*. In (Shirahatti and Barnard 2005) a method for comparing image retrieval systems was proposed relying on the Corel database, which has restricted copyrights, is no longer commercially available today, and can therefore not be used for experiments that are meant to be a basis for other comparisons.

Another aspect of evaluating CBIR systems are the requirements of the users. In (Markkula and Sormunen 1998) and (Armitage and Enser 1997) studies of user needs in searching image archives are presented and the outcome in both studies is that CBIR alone is very unlikely to fulfill the needs but that semantic information obtained from meta data and textual information is an important additional knowledge source. Although today the semantic analysis and understanding of images is much further developed due to the recent achievements in object detection and recognition, still most of the requirements specified are not satisfiable fully automatically. Therefore, in this paper *we compare the performance of a large variety of visual descriptors*. These can then later be combined with the outcome of textual information retrieval as described e.g., in (Deselaers et al. 2006).

The main question we address in this paper is: Which features are suitable for which task in image retrieval? This question is thoroughly investigated by examining the performance of a wide variety of different visual descriptors for four different types of CBIR tasks.

The question of which features perform how well is closely related to the question which features can be combined to obtain good results in a particular task. Although we do not directly address this question here, the results from this paper lead to a new and intuitive method to choose an appropriate combination of features based on the correlation of the individual features.

For the evaluation of the features we use five different publicly available databases which are a good starting point to evaluate the performance of new image descriptors.

Although today various initiatives for evaluation of CBIR systems have evolved, only few of them resulted in evaluation campaigns with participants and results: *Benchathlon*[1]

[1] http://www.benchathlon.net/

was started in 2001 and located at the SPIE Electronic Imaging conference but has become smaller over time. *TRECVID*[2] is an initiative by the TREC (Text Retrieval Conference) on video retrieval in which video retrieval systems are compared. *ImageCLEF*[3] is part of the Cross-Language Evaluation Framework (CLEF) and started in 2003 with only one task aiming at a combination of multi-lingual information retrieval with CBIR. In 2004, it comprised three tasks, one of them focused on visual queries and in 2005 and 2006 there were four tasks, one and two of them purely visual, respectively. We can observe that evaluation in the field of CBIR is at a far earlier stage than it is in textual information retrieval (e.g., Text REtrieval Conference, TREC) or in speech recognition (e.g., Hub4-DARPA evaluation). One reason for this is likely to be the smaller commercial impact that (content-based) image retrieval has had in the past. However, with the increasing amount of visual data available in various form, this is likely to change in the future.

The main contributions of this paper are answers to the questions above, namely

- an extensive overview of features proposed for CBIR, including features that were proposed in the early days of CBIR and techniques that were proposed only recently in the object recognition and image understanding literature as well as a subset of features from the MPEG7 standard.
- a quantitative analysis of the performance of these features for various CBIR tasks (in particular: stock photo retrieval, personal photo retrieval, building/touristic image retrieval, and medical image retrieval).
- pointing out a set of five databases from four different domains that can be used for benchmarking CBIR systems.

Note that we do not focus on the combination of features nor on the use of user feedback for content-based image retrieval in this paper; several other authors propose and evaluate approaches to these important issues (Yavlinski et al. 2004; Heesch and Rüger 2003; Müller et al. 2000; Müller et al. 2000; MacArthur et al. 2000). Instead, we mainly investigate the performance of single features for different tasks.

1.1 State of the art in content-based image retrieval

This section gives an overview on literature on CBIR. We mainly focus on different descriptors and image representations. More general overviews on CBIR are given in (Smeulders et al. 2000; Forsyth and Ponce 2002; Rui et al. 1999). Two recent reviews of CBIR techniques are given in (Datta et al. 2005; Lew et al. 2006).

In CBIR, there are, roughly speaking, two different main approaches: a *discrete approach* and a *continuous approach* (de Vries and Westerveld 2004). (1) The discrete approach is inspired by textual information retrieval and uses techniques like inverted files and text retrieval metrics. This approach requires all features to be mapped to binary features; the presence of a certain image feature is treated like the presence of a word in a text document. (2) The continuous approach is similar to nearest neighbor classification. Each image is represented by a feature vector and these features are compared using various distance measures. The images with lowest distances are ranked highest in the

[2] http://www.nlpir.nist.gov/projects/trecvid/

[3] http://www.imageclef.org

retrieval process. A first, though not exhaustive, comparison of these two models is presented in (de Vries and Westerveld 2004).

Among the first systems that were available were the QBIC system from IBM (Faloutsos et al. 1994) and the Photobook system from MIT (Pentland et al. 1996). QBIC uses color histograms, a moment based shape feature, and a texture descriptor. Photobook uses appearance features, texture features, and 2D shape features. Another well known system is Blobworld (Carson et al. 2002), developed at UC Berkeley. In Blobworld, images are represented by regions that are found in an Expectation-Maximization-like (EM) segmentation process. In these systems, images are retrieved in a nearest-neighbor-like manner, following the continuous approach to CBIR. Other systems following this approach include SIMBA (Siggelkow et al. 2001), CIRES (Iqbal and Aggarwal 2002), SIMPLIcity (Wang et al. 2001), IRMA (Lehmann et al. 2005), and our own system FIRE (Deselaers et al. 2005; Deselaers et al. 2004). The Moving Picture Experts Group (MPEG) defines a standard for content-based access to multimedia data in their MPEG-7 standard. In this standard, a set of descriptors for images is defined. A reference implementation for these descriptors is given in the XM Software.[4] A system that uses MPEG-7 features in combination with semantic web ontologies is presented in Bloehdorn et al. (2005). In Di et al. (2002) a method starting from low-level features and creating a semantic representation of the images is presented and in Meghini et al. (2001) an approach to consistently fuse the efforts in various fields of multimedia information retrieval is presented.

In (Squire et al. 1999), the VIPER system is presented which follows the discrete approach. VIPER is now publicly available as the GNU Image Finding Tool (GIFT) and several enhancements have been implemented during the last years. An advantage of the discrete approach is that methods from textual information retrieval can easily be transferred as e.g., user interaction and storage handling. Nonetheless, most image retrieval systems follow the continuous approach often using some optimization, for example pre-filtering and pre-classification (Smeulders et al. 2000; Wang et al. 2001; Park et al. 2002), to achieve better runtime performance, e.g., (Faloutsos et al. 1994; Pentland et al. 1996; Carson et al. 2002; Siggelkow et al. 2001).

We can clearly observe that many different image description features have been developed. However, only few works have quantitatively compared different features. Interesting insights can also be gained from the outcomes of the ImageCLEF image retrieval evaluations (Clough et al. 2004; Clough et al. 2006) in which different systems are compared on the same task. The comparison is not easy because all groups use different retrieval systems and text-based information retrieval is an important part of these evaluations. Due to the lack of standard tasks, in many papers on image retrieval, new benchmark sets are defined to allow for quantitative comparison of the proposed methods to a baseline system. A problem with this approach is that it is simple to create a benchmark for which you can show improved results (Müller et al. 2002).

Recently, local image descriptors are getting more attention within the computer vision community. The underlying idea is that objects in images consist of parts that can be modelled with varying degrees of independence. These approaches are successfully used for object recognition and detection (Dorkó 2006; Fei-Fei and Perona 2005; Fergus et al. 2003; Opelt et al. 2006; Marée et al. 2005; Deselaers et al. 2005) and CBIR (Deselaers et al. 2004; Jain 2004; Schmid and Mohr 1997; van Gool et al. 2001). For the representation of local image parts, SIFT features (Lowe 2004) and raw image patches are commonly used and a bag-of-features approach, similar to the bag-of-words approach in natural language

[4] http://www.lis.ei.tum.de/research/bv/topics/mmdb/e_mpeg7.html

processing, is commonly taken. The features described in Section 3.7 also follow this approach and are strongly related to the modern approaches in object recognition. In contrast to the methods described above, the image is not modelled as a whole but rather image parts are modelled individually. Most approaches found in the literature on part-based object recognition learn (often complicated) models from a large set of training data. This approach is impractical for CBIR applications since it would require an enormous amount of training data on the one hand and would lead to tremendous computing times to create these models on the other hand. However, some of these approaches are applicable for limited domain retrieval, e.g., on the IRMA database (cf. Section 5.3) (Deselaers et al. 2006).

Overview. The remainder of this paper is structured as follows. The next section describes the retrieval metric used to rank images given a feature and a distance measure and the performance measures used to compare different settings. Section 3 gives an overview of 19 different image descriptors and distance measures which are used for the experiments. Section 4 presents a method to analyze the correlation of different image descriptor/distance combinations. In Section 5, five different benchmark databases are described that are used for the experiments presented in Section 6. The experimental section is subdivided into three parts: Section 6.1 directly compares the performance of the different methods for the different tasks, Section 6.2 describes the results of the correlation analysis, and Section 6.3 analyzes the connection between the error rate and the mean average precision. The paper concludes with answers to the questions posed above.

2 Retrieval metric

The CBIR framework used to conduct the experiments described here follows the continuous approach: images are represented by vectors that are compared using distance measures. For the experiments we use our CBIR system FIRE.[5] FIRE was designed as a research system with extensibility and flexibility in mind. For the evaluation of features, only one feature and one query image is used at a time, as described in the following.

Retrieval Metric. Let the database $\{x_1, \ldots x_n, \ldots, x_N\}$ be a set of images represented by features. To retrieve images similar to a query image q, each database image x_n is compared with the query image using an appropriate distance function $d(q, x_n)$. Then, the database images are sorted according to the distances such that $d(q, x_{n_i}) \leq d(q, x_{n_{i+1}})$ holds for each pair of images x_{n_i} and $x_{n_{i+1}}$ in the sequence $(x_{n_1} \ldots, x_{n_i}, \ldots x_{n_N})$. If a combination of different features is used, the distances are normalized to be in the same value range and then a linear combination of the distances is used to create the ranking.

To evaluate CBIR, several performance evaluation measures have been proposed (Müller et al. 2001) based on the precision P and the recall R:

$$P = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved}},$$

$$R = \frac{\text{Number of relevant images retrieved}}{\text{Total number of relevant images}},$$

Precision and recall values are usually represented in a precision-recall-graph $R \rightarrow P(R)$ summarizing $(R, P(R))$ pairs for varying numbers of retrieved images. The most common

[5] freely available under the terms of the GNU General Public Licencse at http://www-i6.informatik.rwth-aachen.de/~deselaers/fire.htm

way to summarize this graph into one value is the mean average precision that is also used e.g., in the TREC and CLEF evaluations. The average precision AP for a single query q is the mean over the precision scores after each retrieved relevant item:

$$AP(q) = \frac{1}{N_R}\sum_{n=1}^{N_R} P_q(R_n),$$

where R_n is the recall after the nth relevant image was retrieved. N_R is the total number of relevant documents for the query. The mean average precision MAP is the mean of the average precision scores over all queries:

$$MAP = \frac{1}{|Q|}\sum_{q\in Q} AP(q),$$

where Q is the set of queries q.

An advantage of the mean average precision is that it contains both precision and recall oriented aspects and is sensitive to the entire ranking.

We also indicate the classification error rate ER for all experiments. To do so we consider only the most similar image according to the applied distance function. We consider a query image to be classified correctly, if the first retrieved image is relevant. Otherwise the query is misclassified:

$$ER = \frac{1}{|Q|}\sum_{q\in Q}\begin{cases}0 & \text{if the most similar image is relevant/from the correct class}\\ 1 & \text{otherwise.}\end{cases}$$

This is in particular interesting if the database for retrieval consists of images labelled with classes, which is the case for some of the databases considered in this paper. For databases without defined classes but with selected query images and corresponding relevant images, the classes to be distinguished are "*relevant*" and "*irrelevant*" only.

This is in accordance with precision at document X being used as an additional performance measure in many information retrieval evaluations. The ER used here is equal to $1 - P(1)$, where $P(1)$ is the precision after one document retrieved. In (Deselaers et al. 2004) it was experimentally shown that the error rate and $P(50)$, the precision after 50 documents, are correlated with a coefficient of 0.96 and thus they essentially describe the same property. The precision oriented evaluation is interesting, because most search engines, both for images and text, return between 10 and 50 results, given a query.

Using the ER, the image retrieval system can be viewed as a nearest neighbor classifier using the same features and the same distance function as the image retrieval system. The decision rule of this classifier can be written in the form

$$q \rightarrow r(q) = \arg \min_{k=1,\ldots,K} \{ \min_{n=1,\ldots,N_k} d(q, x_{nk})\}.$$

The query image q is predicted to be from the same class as the database image that has the smallest distance to it. Here, x_{nk} denotes the n-th image of class k.

3 Features for CBIR

In this section we give an overview of the features tested, with the intention to include as many features as possible. Obviously we cannot cover all features that have been proposed in the literature. For example, we have left out the Blobworld features (Carson et al. 2002) because for comparing images based on these features, user interaction to select the relevant regions in the query image is required. Furthermore, a variety of texture representations have not been included and we have not investigated different color spaces.

However, we have tried to make the selection of features as representative and at the state-of-the-art as possible. Roughly speaking, the features can be grouped into the following types: (a) color representation, (b) texture representation, (c) local features, and (d) shape representation.[6] The features that are presented in the following are grouped according to these four categories in Table 1. Table 1 also gives the timing information on feature extraction and retrieval time for a database consisting of 10 images.[7]

The distance function used to compare the features representing an image obviously also has a big influence on the performance of the system. Therefore, we refer to the used distance functions for each feature in the particular sections. We have chosen distance functions that are known to work well for the features used as the discussion of their influence is beyond the scope of this paper. Different comparison measures for histograms are presented e.g., in (Puzicha et al. 1999; Nölle 2003) and dissimilarity metrics for direct image comparison are presented in Keysers et al. (2007).

3.1 Appearance-based image features

The most straight-forward approach is to directly use the pixel values of the images as features: the images are scaled to a common size and compared using the Euclidean distance. In this work, we have used a 32×32 down-sampled representation of the images and these have been compared using the Euclidean distance. It has been observed that for classification and retrieval of medical radiographs, this method serves as a reasonable baseline (Keysers et al. 2007).

In Keysers et al. (2007) different methods were proposed to directly compare images accounting for local deformations. The proposed *image distortion model* (IDM) is shown to be a very effective means of comparing images with reasonable computing time. IDM clearly outperforms the Euclidean distance for optical character recognition and medical radiographs. The IDM is a non-linear deformation model, it was also successfully used to compare general photographs (Deselaers 2003) and for sign language and gesture recognition (Zahedi et al. 2005). In this work it is used as a second comparison measure to compare images directly. Therefore the images are scaled to have a common width of 32 pixels while keeping the aspect ratio constant, i.e., the images may be of different heights.

[6] Note that no features that fully cover the shapes in the images are included since therefore an algorithm segmenting the images into meaningful regions is required, but since fully-automatic segmentation for general images is an unsolved problem, it is not covered here. The features that we mark to represent shape only represent shape in a local (for the SIFT features) and very rough global context (for appearance-based image features). There are however, overview papers on the shape features defined in MPEG7 which use databases consisting of segmented images for benchmarks (Bober 2001).

[7] These experiments have been carried out on a 1.8 GHz machine with our standard C++ implementation of the software. The SIFT feature extraction was done with the software from Gyuri Dorko (http://www.lear.inrialpes.fr/people/dorko/downloads.html), the MPEG7 experiments were performed with the MPEG7 XM reference implementation (http://www.lis.ei.tum.de/research/bv/topics/mmdb/mpeg7.html), and the downscaling of images was performed using the ImageMagick library (http://www.imagemagick.org/). The timings include the time to load all data and initialize the system.

Table 1 Grouping of the features into different types

Feature name	Section	Comp. measure	Type	Extr.[s]	Retr.[s]
Appearance-based image features					
32 × 32 image	3.1	Euclidean	abcd	0.25	0.19
X × 32 image	3.1	IDM	abcd	0.25	9.72
Color histograms	3.2	JSD	a	0.77	0.16
Tamura features	3.3	JSD	b	14.24	0.13
Global texture descriptor	3.4	Euclidean	b	3.51	0.16
Gabor histogram	3.5	JSD	b	8.01	0.12
Gabor vector	3.5	Euclidean	b	8.68	0.17
Invariant feature histograms					
w. monomial kernel	3.6	JSD	ab	28.93	0.16
w. relational kernel	3.6	JSD	ab	18.23	0.14
LF patches					
Global search	3.7	–	ac	4.69	7.13
Histograms	3.7	JSD	ac	4.69 + 5.17	0.27
Signatures	3.7	EMD	ac	4.69 + 3.37	0.55
LF SIFT					
Global search	3.7	–	cd	11.91	9.23
Histograms	3.7	JSD	cd	11.91 + 6.23	0.27
Signatures	3.7	EMD	cd	11.91 + 4.50	1.03
MPEG 7: scalable color	3.8.1	MPEG7-internal	a	0.48	0.42
MPEG 7: color layout	3.8.2	MPEG7-internal	ad	0.20	0.33
MPEG 7: edge histogram	3.8.3	MPEG7-internal	b	0.16	0.43

(a) color representation, (b) texture representation, (c) local features, (d) shape representation. The table also gives the time to extract the features from 10 images and to query 10 images in a 10 image database to give an impression of the computational costs of the different features (experiments were performed on a 1.8 GHz machine)

3.2 Color histograms

Color histograms are among the most basic approaches and widely used in image retrieval (Smeulders et al. 2000; Faloutsos et al. 1994; Deselaers 2003; Puzicha et al. 1999; Swain and Ballard 1991). To show performance improvements in image retrieval systems, systems using only color histograms are often used as a baseline. The color space is partitioned and for each partition the pixels with a color within its range are counted, resulting in a representation of the relative frequencies of the occurring colors. We use the RGB color space for the histograms. We observed only minor differences with other color spaces which was also observed in (Smith and Chang 1996). In accordance with (Puzicha et al. 1999), we use the Jeffrey divergence or Jensen-Shannon divergence (JSD) to compare histograms:

$$d_{JSD}(H,H') = \sum_{m=1}^{M} H_m \log \frac{2H_m}{H_m + H'_m} + H'_m \log \frac{2H'_m}{H'_m + H_m},$$

where H and H' are the histograms to be compared and H_m is the mth bin of H.

3.3 Tamura features

In Tamura et al. (1978) the authors propose six texture features corresponding to human visual perception: *coarseness*, *contrast*, *directionality*, *line-likeness*, *regularity*, and *roughness*. From experiments testing the significance of these features with respect to human perception, it was concluded that the first three features are very important. Thus, in our experiments we use coarseness, contrast, and directionality to create a histogram describing the texture (Deselaers 2003) and compare these histograms using the Jeffrey divergence (Puzicha et al. 1999). In the QBIC system (Faloutsos et al. 1994) histograms of these features are used as well.

3.4 Global texture descriptor

In Deselaers (2003) a texture feature consisting of several parts is described: *Fractal dimension* measures the roughness of a surface. The fractal dimension is calculated using the reticular cell counting method (Haberäcker 1995). *Coarseness* characterizes the grain size of an image. It is calculated depending on the variance of the image. *Entropy* of pixel values is used as a measure of disorderedness in an image. The *spatial gray-level difference statistics* describe the brightness relationship of pixels within neighborhoods. It is also known as co-occurrence matrix analysis (Haralick et al. 1973). The *circular Moran autocorrelation function* measures the roughness of the texture. For the calculation a set of autocorrelation functions is used (Gu et al. 1989). From these, we obtain a 43 dimensional vector consisting of one value for the fractal dimension, one value for the coarseness, one value for the entropy and 32 values for the difference statistics, and 8 values for the circular Moran autocorrelation function. This descriptor has been successfully used for medical images in Lehmann et al. (2005).

3.5 Gabor features

Gabor features have been widely used for Texture analysis (Park et al. 2002; Squire et al. 1999). Here we use two different descriptors derived from Gabor features:

- Mean and standard deviation: Gabor features are extracted at different scales and directions from the images and the mean and standard deviation of the filter responses is calculated. We extract Gabor features in five different orientations and five different scales leading to a 50 dimensional vector.
- A bank of 12 different circularly symmetric Gabor filters is applied to the image, the energy for each filter on the bank is quantized into 10 bands and a histogram of the mean filter outputs over image regions is computed to give a global measure of the texture characteristics of the image (Squire et al. 1999). These histograms are compared using the JSD.

3.6 Invariant feature histograms

A feature is called invariant with respect to certain transformations if it does not change when these transformations are applied to the image. The transformations considered here are translation, rotation, and scaling. In this work, invariant feature histograms as presented in (Siggelkow 2002) are used. These features are based on the idea of constructing invariant features by integration, i.e., a certain feature function is integrated over the set of all considered transformations. The feature functions we have considered are monomial and relational functions (Siggelkow et al. 2001) over the pixel intensities. Instead of summing over translation and rotation, we only sum over rotation and create a histogram over translation. This histogram is still invariant with respect to rotation and translation. The resulting histograms are compared using the JSD. Previous experiments have shown that the characteristics of invariant feature histograms and color histograms are very similar and that invariant feature histograms can sometimes outperform color histograms (Deselaers et al. 2004).

3.7 Local image descriptors

Image patches, i.e., small subimages of images, or features derived thereof currently are a very promising approach for object recognition, e.g., (Deselaers et al. 2005; Fergus et al. 2005; Paredes et al. 2001). Obviously, object recognition and CBIR are closely related fields (Vailaya et al. 2001; Antani et al. 2002) and for some clearly defined retrieval tasks, object recognition methods might actually be the only possible solution: e.g., looking for all images showing a certain person, clearly a face detection and recognition system would deliver the best results (Pentland et al. 1996; Deselaers et al. 2005).

We consider two different types of local image descriptors or local features (LF): (a) patches that are extracted from the images at salient points and dimensionality reduced using PCA transformation (Deselaers et al. 2005) and (b) SIFT descriptors (Lowe 2004) extracted at Harris interest points (Dorkó 2006, chapters 3, 4).

We employ three methods to incorporate local features into our image retrieval system. The methods are evaluated for both types of local features described above:

LF histograms. The first method follows (Deselaers et al. 2005): local features are extracted from all database images and jointly clustered to form 2,048 clusters. Then for each of the local features all information except the identifier of the most similar cluster center is discarded and for each image a histogram of the occurring patch-cluster identifiers is created, resulting in a 2,048 dimensional histogram per image. These histograms are then used as features in the retrieval process and are compared using the Jeffrey divergence. This method was shown to produce good performance in object recognition and detection tasks (Deselaers et al. 2005). Note that the timing information in Table 1 does not give the time to create the cluster model, since this is only done once for a database and can be computed offline.

LF signatures. The second method is derived from the method proposed in (Mikolajczyk et al. 2005). Local features are extracted from each database image and clustered for each image separately to form 32 clusters per image. Then for each image, the parameters of the clusters, i.e., the mean and the variance, are saved and the according cluster-identifier histogram of the extracted features is created. These "local feature signatures" are then used as features in the retrieval process and are compared using the Earth Mover's Distance (EMD) (Rubner et al. 1998). This method was shown to produce good performance in object recognition and detection tasks (Mikolajczyk et al. 2005).

LF global search. The third method is based on global patch search and derived from the method presented in (Paredes et al. 2001). Here, local features are extracted from all database images and stored in a KD tree to allow for efficient nearest neighbor searching.

Given a query image, we extract local features from the image in the same way as we did for the database images and search for the k nearest neighbors for each of the query-patches in the set of database-patches. Then, we count how many patches from each of the database image were found for the query patches and the database images with the highest number of patch-hits are returned. Note that the timing information in Table 1 does not include the time to create the KD tree, since this is only done once for a database and can be computed offline.

3.8 MPEG-7 features

The Moving Picture Experts Group (MPEG) has defined several visual descriptors in their standard referred to as *MPEG-7 standard*.[8] An overview of these features can be found in (Eidenberger 2003; Manjunath et al. 2001; Ohm 2001; Yang and Kuo 1999). The MPEG initiative focuses strongly on features that are computationally inexpensive to obtain and to compare and also strongly optimizes the features with respect to the required memory for storage.

Coordinated by the MPEG, a reference implementation of this standard has been developed.[9] This reference implementation was used in our framework for experiments with these features. Unfortunately, the software is not yet in a fully functional state and thus only three MPEG7 features could be used in the experiments. For each of these features, we use the comparison measures proposed by the MPEG standard and implemented in the reference implementation. The feature types are briefly described in the following:

3.8.1 MPEG 7: scalable color descriptor

The *scalable color descriptor* is a color histogram in the HSV color space that is encoded by a Haar transform. Its binary representation is scalable in terms of bin numbers and bit representation accuracy over a broad range of data rates. Retrieval accuracy increases with the number of bits used in the representation. We use the default setting of 64 coefficients.

3.8.2 MPEG 7: color layout descriptor

This descriptor effectively represents the spatial distribution of the color of visual signals in a very compact form. This compactness allows visual signal matching functionality with high retrieval efficiency at very small computational costs. It allows for query-by-sketch queries because the descriptor captures the layout information of color features. This is a clear advantage over other color descriptors. This approach closely resembles the use of very small thumbnails of the images with a quantization of the colors used.

3.8.3 MPEG 7: edge histogram

The *edge histogram descriptor* represents the spatial distribution of five types of edges, namely four directional edges and one non-directional edge. According to the MPEG-7

[8] http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.html

[9] http://www.lis.e-technik.tu-muenchen.de/research/bv/topics/mmdb/e_mpeg7.html

standard, the image retrieval performance can be significantly improved if the edge histogram descriptor is combined with other descriptors such as the color histogram descriptor. The descriptor is scale invariant and supports rotation invariant and rotation sensitive matching operations.

4 Correlation analysis of features for CBIR

After discussing various features, now let us assume that a set of features is given, some of which account for color, others accounting for texture, and maybe others accounting for shape. A very interesting question then is, how features that can be used in combination can be chosen. Automatic methods for feature selection have e.g., been proposed in (Vasconcelos and Vasconcelos 2004; Najjar et al. 2003). These automatic methods, however do not directly explain why features are chosen, are difficult to manipulate from a user's perspective, and normally require labelled training data.

The method proposed here does not require training data but only analyzes the correlations between the features themselves, and instead of automatically selecting a set of features it provides the user with information helping to select an appropriate set of features.

To analyze the correlation between different features, we analyze the correlation between the distances $d(q, X)$ obtained for each feature of each of the images X from the database given a query q. For each pair of query image q and database image X we create a vector $(d_1(q, X), d_2(q, X),\ldots d_m(q, X),\ldots,d_M(q, X))$ where $d_m(q, X)$ is the distance of the query image q to the database image X for the mth feature. Then we calculate the correlation between the d_m over all $q \in \{q_1,\ldots,q_l,\ldots q_L\}$ and all $X \in \{X_1,\ldots,X_n,\ldots,X_N\}$.

The M × M covariance matrix Σ of the d_m is calculated over all N database images and all L query images as:

$$\Sigma_{ij} = \frac{1}{NL}\sum_{n=1}^{N}\sum_{l=1}^{L}(d_i(q_l, X_n) - \mu_i)\cdot\left(d_j(q_l, X_n) - \mu_j\right) \tag{1}$$

with $\mu_i = \frac{1}{NL}\sum_{n=1}^{N}\sum_{l=1}^{L} d_i(q_l, X_n)$.

Given the covariance matrix Σ, we calculate the correlation matrix $\mathcal{R}$ as $\mathcal{R}_{ij} = \Sigma_{ij}/\sqrt{\Sigma_{ii}\Sigma_{jj}}$. The entries of this correlation matrix can be interpreted as similarities of different features. A high value $\mathcal{R}_{ij}$ means a high similarity between features i and j. This similarity matrix can then be analyzed to find out which features have similar properties and which do not. One way to do this is to visualize it using multi-dimensional scaling (Hand et al. 2001, p. 84ff). *Multi-dimensional scaling (MDS)* seeks a representation of data points in a lower dimensional space while preserving the distances between data points as well as possible. To visualize this data by multi-dimensional scaling, we convert the similarity matrix $\mathcal{R}$ into a dissimilarity matrix $\mathcal{D}$ by setting $\mathcal{D}_{ij} = 1 - |\mathcal{R}_{ij}|$. For visualization purposes, we choose a two-dimensional space for MDS.

5 Benchmark databases for CBIR

To cover a wide range of different applications in which CBIR is used, we propose benchmark databases from different domains. In the ImageCLEF evaluations large image retrieval benchmark databases have been collected. However, these are not suitable for the comparison of image features as for most of the tasks textual information is supplied and necessary for an appropriate solution of the task. Table 2 gives an overview of the

Table 2 Summary of the databases used for the evaluation with database name, number of images in the database, number of query images, average number of relevant images per query, and a description how the queries are evaluated

Database	Images	Queries	Avg. rel	Query mode
WANG	1,000	1,000	99.0	Leaving-one-out
UW	1,109	1,109	59.3	Leaving-one-out
IRMA 10000	10,000	1,000	520.2	Test & database images are disjoint
ZuBuD	1,005	105	5.0	Test & database images are disjoint
UCID	1,338	262	3.5	Leaving-one-out

databases used in the evaluations. Although the databases presented here are small in comparison to other CBIR tasks, they represent a wide variety of tasks and allow for a meaningful comparison of feature performances.

The WANG database (Section 5.1), as a subset from the Corel stock photo collection, can be considered similar to stock photo searches. The UW database (Section 5.2) and the UCID database (Section 5.5) mainly consist of personal images and represent the home user domain. The ZuBuD database (Section 5.4) and the IRMA database (Section 5.3) are limited domain tasks for touristic/building retrieval and medical applications, respectively.

5.1 WANG database

The WANG database is a subset of 1,000 images of the Corel stock photo database which have been manually selected and which form 10 classes of 100 images each. One example of each class is shown in Fig. 1. The WANG database can be considered similar to common stock photo retrieval tasks with several images from each category and a potential user having an image from a particular category and looking for similar images which have e.g. cheaper royalties or which have not been used by other media. The 10 classes are used for relevance estimation: given a query image, it is assumed that the user is searching for images from the same class, and therefore the remaining 99 images from the same class are considered relevant and the images from all other classes are considered irrelevant.

5.2 UW database

The database created at the University of Washington consists of a roughly categorized collection of 1,109 images. These images are partly annotated using keywords. The remaining images were annotated by our group to allow the annotation to be used for relevance estimation; our annotations are publicly available.[10]

The images are of various sizes and mainly include vacation pictures from various locations. There are 18 categories, for example “spring flowers”, “Barcelona”, and “Iran”. Some example images with annotations are shown in Fig. 2. The complete annotation consists of 6,383 words with a vocabulary of 352 unique words. On the average, each image has about 6 words of annotation. The maximum number of keywords per image is 22 and the minimum is 1. The database is freely available.[11] The relevance assessment for

[10] http://www-i6.informatik.rwth-aachen.de/~deselaers/uwdb/index.html

[11] http://www.cs.washington.edu/research/imagedatabase/groundtruth/

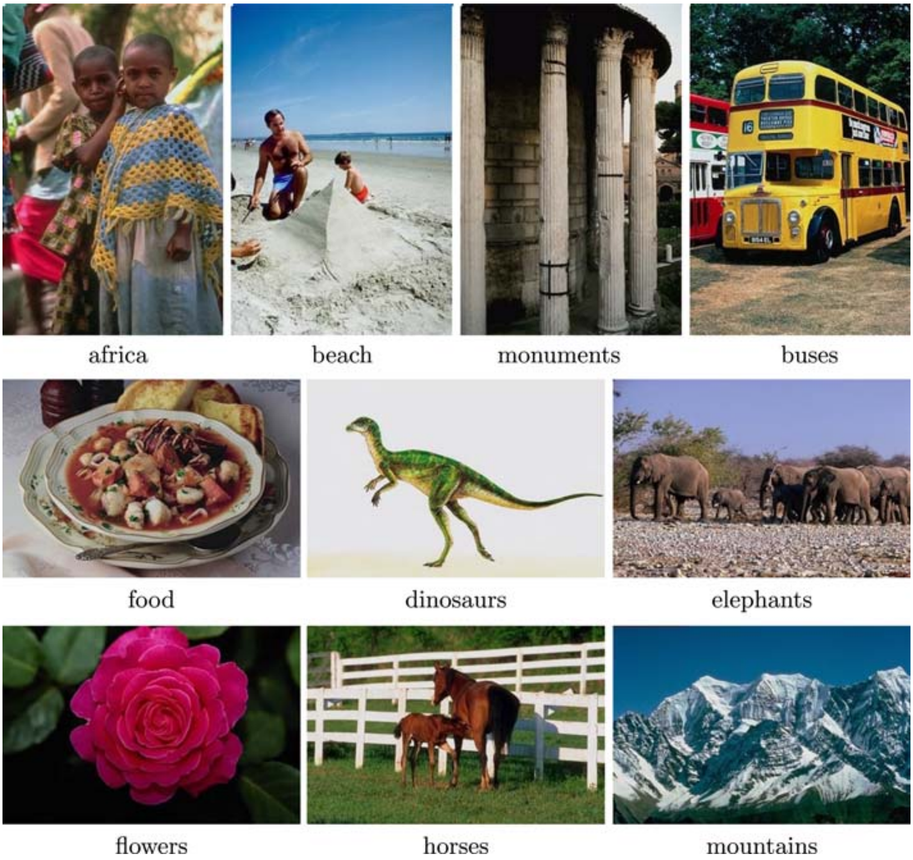


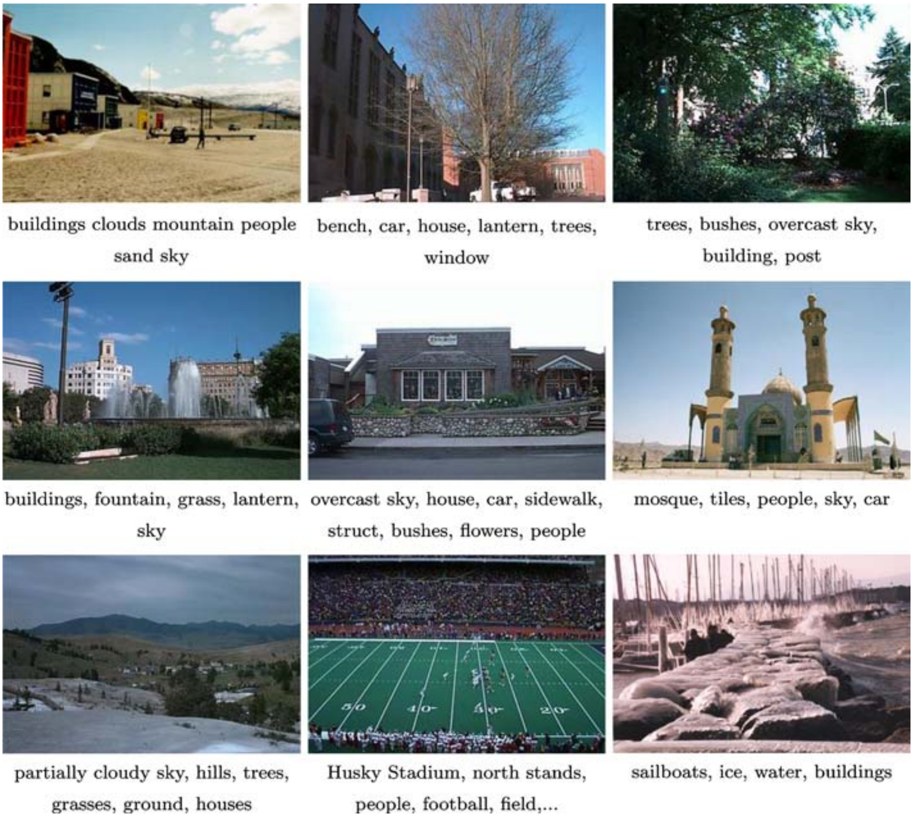
Fig. 1 One example image from each of the 10 classes of the WANG database together with their class labels

the experiments with this database were performed using the annotation: an image is considered to be relevant w.r.t. a given query image if the two images have a common keyword in the annotation. On the average, 59.3 relevant images correspond to each image. The keywords are rather general; thus for example images showing sky are relevant w.r.t. each other, which makes it quite easy to find relevant images (high precision is likely easy) but it can be extremely difficult to obtain a high recall since some images showing sky might have hardly any visual similarity with a given query.

This task can be considered a personal photo retrieval task, e.g., a user with a collection of personal vacation pictures is looking for images from the same vacation, or showing the same type of building.

5.3 IRMA-10000 database

The IRMA database consists of 10,000 fully annotated radiographs taken randomly from medical routine at the RWTH Aachen University Hospital. The images are split into 9,000 training and 1,000 test images. The images are subdivided into 57 classes. The IRMA



buildings clouds mountain people sand sky

bench, car, house, lantern, trees, window

trees, bushes, overcast sky, building, post

buildings, fountain, grass, lantern, sky

overcast sky, house, car, sidewalk, struct, bushes, flowers, people

mosque, tiles, people, sky, car

partially cloudy sky, hills, trees, grasses, ground, houses

Husky Stadium, north stands, people, football, field,...

sailboats, ice, water, buildings

Fig. 2 Examples from the UW database with annotation

database was used in the ImageCLEF 2005 image retrieval evaluation for the automatic annotation task. For CBIR, the relevances are defined by the classes, given a query image from a certain class, all database images from the same class are considered relevant. Example images along with their class numbers and textual descriptions of the classes are given in Fig. 3. This task is a medical image retrieval task and is in practical use at the Department for Diagnostic Radiology of the RWTH Aachen University Hospital.

As all images from this database are gray value images, we evaluate neither the color histograms nor the MPEG7 scalable color descriptor since they only account for color information.

5.4 ZuBuD database

The “Zurich Buildings Database for Image Based Recognition” (ZuBuD) is a database which has been created by the Swiss Federal Institute of Technology in Zurich and is described in more detail in (Shao et al. 2003a, 2003b).

The database consists of two parts, a training part of 1,005 images of 201 buildings, 5 of each building and a query part of 115 images. Each of the query images contains one of the buildings from the main part of the database. The pictures of each building are taken from

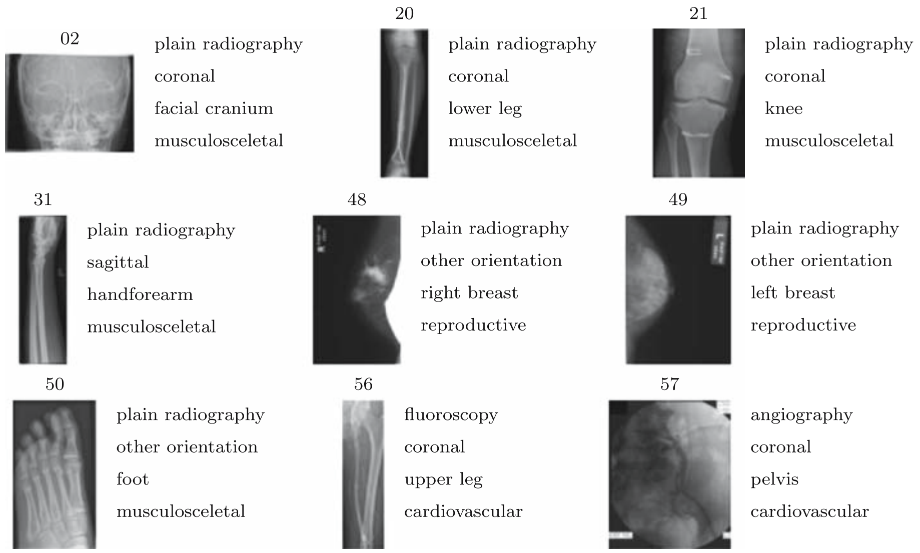


Fig. 3 Example images of the IRMA 10000 database along with their class and annotation

different viewpoints and some of them are also taken under different weather conditions and with two different cameras. Given a query image, only images showing exactly the same building are considered relevant. To give a more precise idea of this database, some example images are shown in Fig. 4.

This database can be considered as an example for a mobile travel guide task, which attempts to identify buildings in pictures taken with a mobile phone camera and then

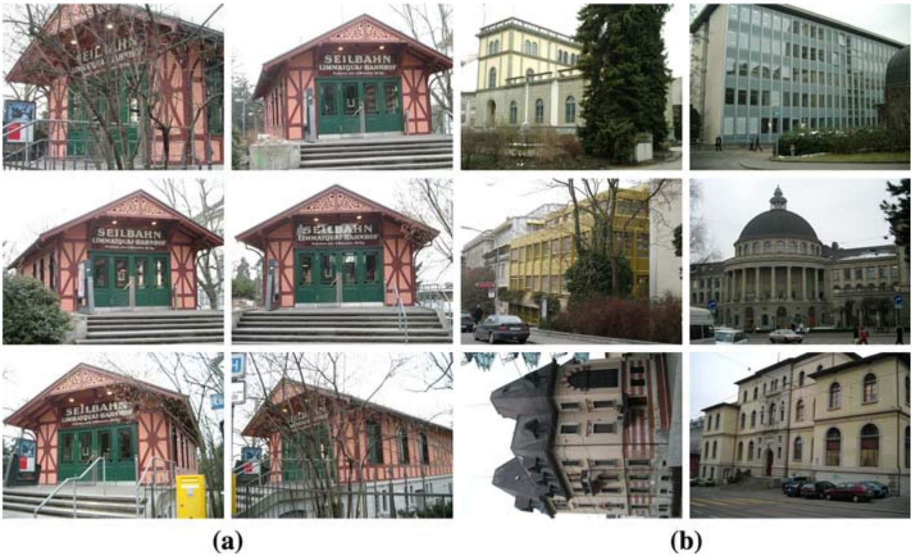


Fig. 4 (**a**) A query image and the 5 images from the same building in the ZuBuD-database (**b**) 6 images of different buildings in the ZuBuD-database

obtains certain information about the building (Shao et al. 2003). The ZuBuD database is freely available.[12]

5.5 UCID database

The UCID database[13] was created as a benchmark database for CBIR and image compression applications (Schaefer and Stich 2004). In Schaefer (2004) this database was used to measure the performance of a CBIR system using compressed domain features. This database is similar to the UW database as it consists of vacation images and thus poses a similar task.

For 264 images, manual relevance assessments among all database images were created, allowing for performance evaluation. The images that are judged to be relevant are images which are very clearly relevant, e.g., for an image showing a particular person, images showing the same person are searched and for an image showing a football game, images showing football games are considered to be relevant. The used relevance assumption makes the task easy on one hand, because relevant images are very likely quite similar, but on the other hand, it makes the task difficult, because there are likely images in the database which have a high visual similarity but which are not considered relevant. Thus, it can be difficult to have high precision results using the given relevance assessment, but since only few images are considered relevant, high recall values might be rather easy to obtain. Example images are given in Fig. 5.

6 Evaluation of the features considered

In this section we report the results of the experimental evaluation of the features. To evaluate all features on the given databases, we extracted the features from the images and executed experiments to test the particular features. For all experiments, we report the mean average precision and the classification error rate. The connection between the classification error rate and mean average precision shows the strong relation between CBIR and classification. Both performance measures have advantages. The error rate is very precision oriented and thus it is best if relevant images are retrieved early. On the contrary, the mean average precision accounts for the average performance over the complete PR graph. Furthermore, we calculated the distance vectors mentioned in Section 4 for each of the queries performed to obtain a global correlation analysis of all features.

6.1 Performance evaluation of features

The results from the single feature experiments are given in Figs. 6 and 7 and in Tables 3 and 4. The results are sorted by the average of the classification error rates. The results from the correlation analysis are given in Fig. 9. Note that the features ‘color histogram’ and ‘MPEG7 scalable color’ were not evaluated for the IRMA database because pure color descriptors are not suitable for this gray-scale database.

[12] http://www.vision.ee.ethz.ch/ZuBuD

[13] http://www.vision.doc.ntu.ac.uk/datasets/UCID/ucid.html

Fig. 5 Example images from the UCID database

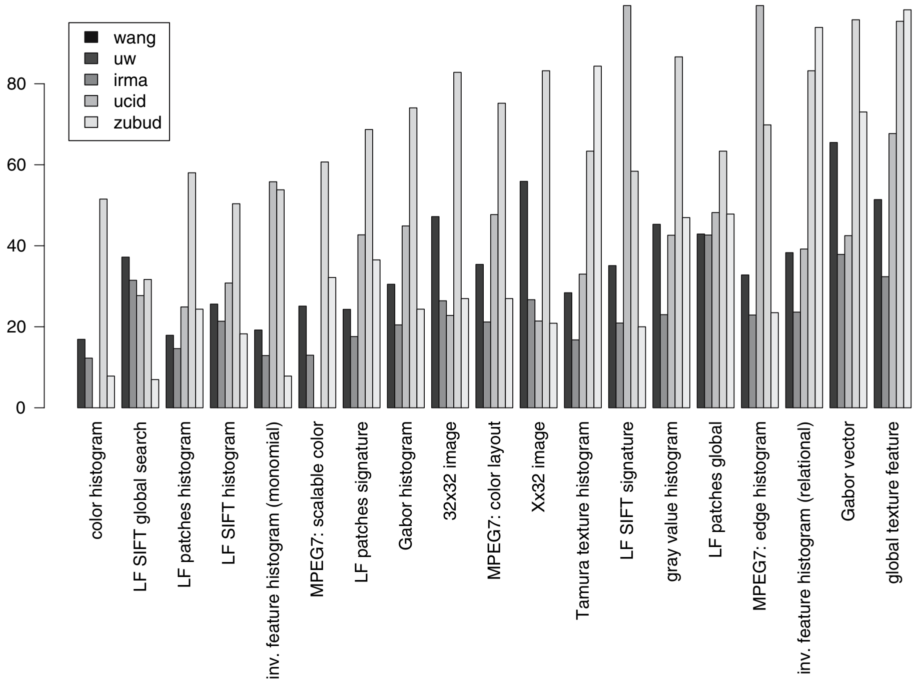


Fig. 6 Classification error rate [%] for each of the features for each of the databases (sorted by average error rate over the databases). The different shades of gray denote different databases and the blocks of bars denote different features

It can clearly be seen that different features perform differently on the databases. Grouping the features by performance results in three groups, one group of five features clearly outperforms the other features (average error rate $< 30\%$, average mean average precision $\approx 50\%$). A second group has average error rates of approximately 40%

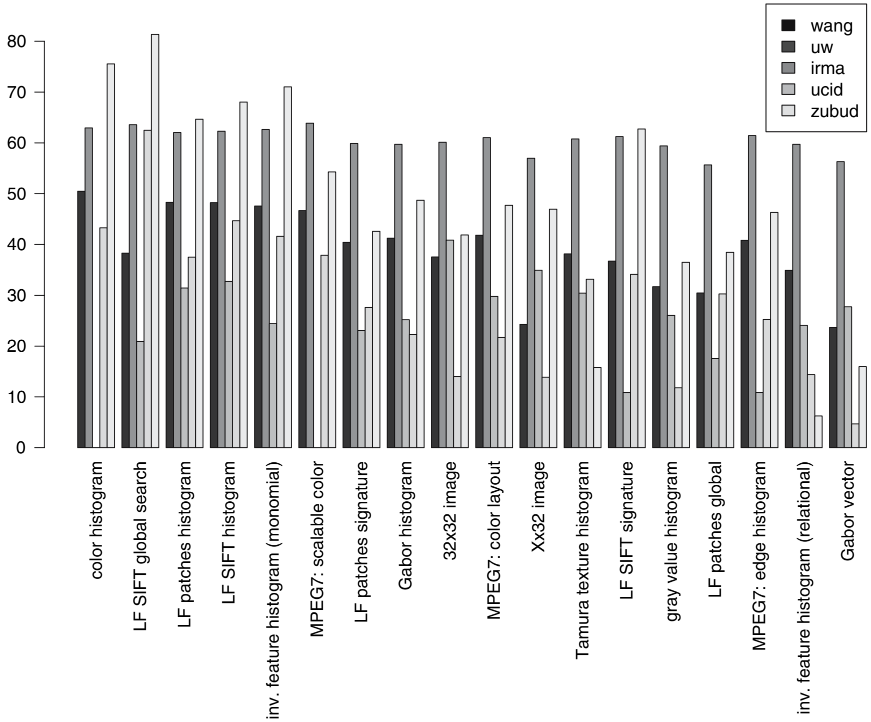


Fig. 7 Mean average precision for each of the features for each of the databases (sorted in the same order as Fig. 6 to allow for easy comparison)

(respectively average mean average precision 40%) and a last group performs clearly worse.

The top group is led by the color histogram which performs very well for *all* color tasks and has not been evaluated on the IRMA data. When all databases are considered, the global feature search (cf. Section 3.7) of SIFT features extracted at Harris points (Dorkó 2006, chapters 3, 4) performs best on the average. This good performance is probably partly due to the big success on the ZuBuD database, where features of similar type were observed to perform exceedingly well (Obdrzalek and Matas 2003). They also perform well on the UCID database, where relevant images, in contrast to the UW task, are very close neighbors. The possible high dissimilarity between relevant images in the UW database, thus explains the bad performance there. However, the patch histograms outperform the SIFT features on all other tasks as they include color information which obviously is very important for most of the tasks. They also obtain a good performance for the IRMA data. It can be observed that the error rates for the UCID database are very high in comparison to the other databases, so the UCID task can be considered to be harder than e.g., the UW task.

A similar result to the one obtained using color histogram is obtained by the invariant feature histogram with monomial kernel. This is not surprising, as it is very similar to a color histogram, except that it also partly accounts for local texture. It can be observed that the performance for the color databases is nearly identical to the color histogram. The relatively bad ranking of these features in the tables is due to the bad performance on the IRMA task.

Table 3 Error rate [%] for each of the features for each of the databases (sorted by average error rate over the databases)

Feature	WANG	UW	IRMA	UCID	ZuBuD	Average
Color histogram	16.9	12.3	–	51.5	7.8	22.1
LF SIFT global search	37.2	31.4	27.7	31.7	7.0	27.0
LF patches histogram	17.9	14.6	24.9	58.0	24.4	28.0
LF SIFT histogram	25.6	21.4	30.8	50.4	18.3	29.3
Inv. feature histogram (monomial)	19.2	12.9	55.80	53.8	7.8	29.9
MPEG7: scalable color	25.1	13.0	–	60.7	32.2	32.7
LF patches signature	24.3	17.6	42.7	68.7	36.5	38.0
Gabor histogram	30.5	20.5	44.9	74.1	24.4	38.9
32 × 32 Image	47.2	26.4	22.8	82.8	27.0	41.2
MPEG7: color layout	35.4	21.2	47.7	75.2	27.0	41.2
X × 32 image	55.9	26.7	21.4	83.2	20.9	41.6
Tamura texture histogram	28.4	16.8	33.0	63.7	84.4	45.2
LF SIFT signature	35.1	20.9	99.3	58.4	20.0	46.7
Gray value histogram	45.3	23.0	42.6	86.64	47.0	48.9
LF patches global	42.9	42.7	48.2	63.4	47.8	49.0
MPEG7: edge histogram	32.8	22.9	99.3	69.9	23.5	49.7
Inv. feature histogram (relational)	38.3	23.6	39.2	83.2	93.9	55.6
Gabor vector	65.5	37.9	42.5	95.8	73.0	62.9
Global texture feature	51.4	32.4	67.7	95.4	98.3	69.0

Leaving out the IRMA task for this feature, it would be ranked second in the entire ranking. The high similarity of color histograms and invariant feature histograms with monomial kernel can also directly be observed in Fig. 9 where it can be seen that color histograms (point 1) and invariant feature histograms with monomial kernel (point 11) have very similar properties.

The second group of features consists of four features: signatures of SIFT features, appearance-based image features, and the MPEG 7 color layout descriptor.

Although the image thumbnails compared with the image distortion model perform quite poorly for the WANG, the UW, and the UCID tasks, they perform extremely well for the IRMA task and reasonably well for the ZuBuD task. A major difference between these tasks is that the first three databases contain general color photographs of completely unconstrained scenes, whereas the latter ones contain images from limited domains only.

The simpler appearance-based feature of 32 × 32 thumbnails of the images, compared using Euclidean distance, is the next best feature, and again it can be observed that it performs well for the ZuBuD and IRMA tasks only.

As expected, the MPEG7 color layout descriptor and 32 × 32 image thumbnails obtain similar results because they both encode the spatial distribution of colors or gray values in the images.

Among the texture features (Tamura texture histogram, Gabor features, global texture descriptor, relational invariant feature histogram, and MPEG-7 edge histogram), the Tamura texture histogram and the Gabor histogram outperform the others.

Table 4 Mean average precision [%] for each of the features for each of the databases (sorted in the same order as Table 3 to allow for easy comparison)

Feature	WANG	UW	IRMA	UCID	ZuBuD	Average
color histogram	16.9	12.3	–	51.5	7.8	22.1
LF SIFT global search	37.2	31.4	27.7	31.7	7.0	27.0
LF patches histogram	17.9	14.6	24.9	58.0	24.4	28.0
LF SIFT histogram	25.6	21.4	30.8	50.4	18.3	29.3
Inv. feature histogram (monomial)	19.2	12.9	55.80	53.8	7.8	29.9
MPEG7: scalable color	25.1	13.0	–	60.7	32.2	32.7
LF patches signature	24.3	17.6	42.7	68.7	36.5	38.0
Gabor histogram	30.5	20.5	44.9	74.1	24.4	38.9
32 × 32 image	47.2	26.4	22.8	82.8	27.0	41.2
MPEG7: color layout	35.4	21.2	47.7	75.2	27.0	41.2
X × 32 image	55.9	26.7	21.4	83.2	20.9	41.6
Tamura texture histogram	28.4	16.8	33.0	63.7	84.4	45.2
LF SIFT signature	35.1	20.9	99.3	58.4	20.0	46.7
Gray value histogram	45.3	23.0	42.6	86.64	47.0	48.9
LF patches global	42.9	42.7	48.2	63.4	47.8	49.0
MPEG7: edge histogram	32.8	22.9	99.3	69.9	23.5	49.7
Inv. feature histogram (relational)	38.3	23.6	39.2	83.2	93.9	55.6
Gabor vector	65.5	37.9	42.5	95.8	73.0	62.9
Global texture feature	51.4	32.4	67.7	95.4	98.3	69.0

6.2 Correlation analysis of features

Figure 8 shows the average correlation of different features over all databases. The darker a field in this image is, the lower the correlation between the corresponding features, bright fields denote high correlations. Figure 9 shows the visualizations of the outcomes of multidimensional scaling of the correlation analysis. We applied the correlation analysis for the different tasks individually (4 top plots) and for all tasks jointly (bottom plot). Multidimensional scaling was used to translate the similarities of the different features into distances in a two-dimensional space. The further away two points are in the graph, the less similar the corresponding features are for CBIR, and conversely the closer together they appear, the higher the similarity between these features.

For each of these plots the according distance vectors obtained from all queries with all database images have been used (WANG database: 1,000,000 distance vectors, UW&UCID database: 194,482+350,557 distance vectors, IRMA database: 9,000,000 distance vectors, ZuBuD database: 115,575 distance vectors, all databases: 10,660,614 distance vectors).

The figures show a very strong correlation between color histograms (point 1) and invariant feature histograms with monomial kernel (point 11). In fact, they lead to hardly any differences in the experiments. For the databases consisting of color photographs they outperform most other features. A high similarity is also observed between the patch signatures (point 14) and the MPEG7 color layout (point 2) for all tasks.

Two other features that are highly correlated are the two methods that use local feature search for the two different types of local features (points 5 and 12). The different

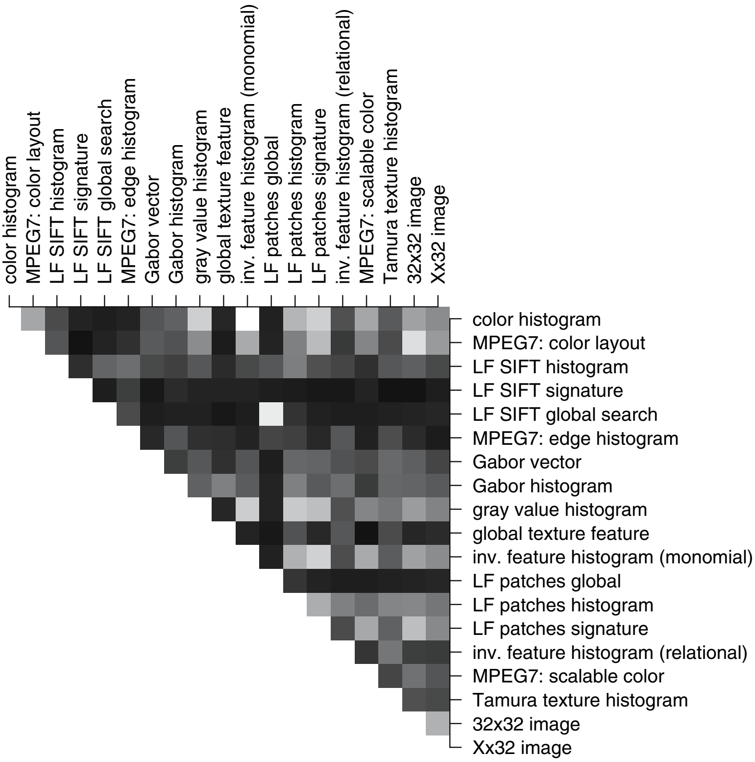


Fig. 8 Correlation of the different features. Bright fields denote high and dark fields denote low correlation. Another representation of this information is given in Fig. 9

comparison methods for local feature histograms/signature have similar performances (3, 4 and 13, 14, respectively).

Another strong correlation can be observed between 32 × 32 image thumbnails (point 18) and the MPEG7 color layout representation (point 2), which was to be expected as both of these have a rough representation of the spatial distribution of colors (resp. gray values) of the images.

Interestingly, the correlation between 32 × 32 images compared using Euclidean distance (point 18) and the X × 32 images compared using the image distortion model (point 18) is low, with only some similarity for the IRMA and the ZuBuD task. This is partly due to the exceedingly good performance of the image distortion model for the IRMA task and partly due to the missing invariance with respect to slight deformations in the images for the Euclidean distance. For example in the ZuBuD task, the image distortion model can partly compensate for the changes in the viewpoints which leads to a much better performance.

Another interesting aspect is that the various texture features (MPEG7 edge histogram (6), global texture feature (10), Gabor features (8, 7), relational invariant feature histogram (15), and Tamura texture histogram (17)) are not correlated strongly. We conclude that none of the texture features is sufficient to completely describe the textural properties of an

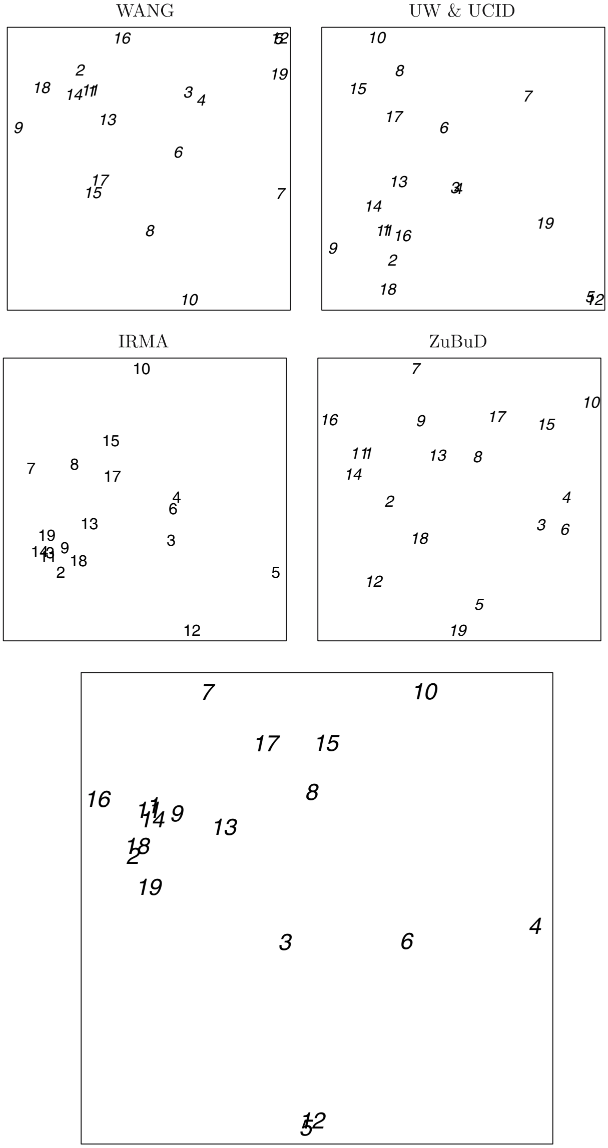


Fig. 9 Correlation of the different features visualized using multi-dimensional scaling. Features that lie close together have similar properties. Top 4 plots: database-wise visualization, bottom plot: all databases jointly. The numbers in the plots denote the individual features: 1: color histogram, 2: MPEG7: color layout, 3: LF SIFT histogram, 4: LF SIFT signature, 5: LF SIFT global search, 6: MPEG7: edge histogram, 7: Gabor vector, 8: Gabor histograms, 9: gray value histogram, 10: global texture feature, 11: inv. feature histogram (monomial), 12: LF patches global, 13: LF patches histogram, 14: LF patches signature, 15: inv. feature histogram (relational), 16: MPEG7: scalable color, 17: Tamura texture histogram, 18: 32 × 32 image, 19: X × 32 image

image. The Tamura texture histogram and the Gabor histogram outperform the other texture features, Tamura features being better in three and Gabor histograms being clearly better in two of the five tasks, both of them are a good choice for texture representation.

To give a little insight into how these plots can be used to select sets of features for a given task, we discuss how features for the WANG database could be chosen in the following paragraph. Combined features are linearly combined as described in Section 2. Here, all features are weighted equally, but some improvement of the retrieval results can be achieved by choosing different weights for the individual features. In Deselaers et al. (2007) we present an approach to automatically learning a feature combination from a set of queries with known relevant images using a discriminative maximum entropy model.

Table 5 Combining features using the results from the correlation analysis described for the WANG database

Features	ER [%]	MAP [%]
Color histograms	16.9	50.5
+ Global texture	15.7	49.5
+ Tamura histograms	13.7	51.2
+ Thumbnails	13.7	53.9
+ Patch histograms	11.6	55.7

Finding a suitable set of features. Assume we are about to create a CBIR system for a new database consisting of general photographs. We extract features from the data and create the according MDS plot (Fig. 9, top left). Since we know that we are dealing with general photographs, we start with a simple color histogram (point 1). The plot now tells us that invariant feature histograms with monomial kernel (11) would not give us much additional information. Next, we consider the various texture descriptors (points 6, 10, 15, 17, 7, 8) and choose one of these, say global texture features (10) and maybe another: Tamura texture histograms (17). Now we have covered color and texture and can consider a global descriptor such as the image thumbnails (18) or a local descriptor such as one of (12, 13, or 14) or (3, 4, or 5). After adding a feature, the performance of the CBIR system can be evaluated by the user. In Table 5 we quantitatively show the influence of adding these features for the WANG database. It can be seen that the performance is incrementally improved by adding more and more features.

6.3 Connection between mean average precision and error rate

In Figs. 10 and 11 the correlation between mean average precision and error rate is visualized database-wise and feature-wise, respectively. The correlation of error rate and mean average precision over all experiments presented in this paper is 0.87. In the keys of the figures, the correlation per database and per feature are given, respectively.

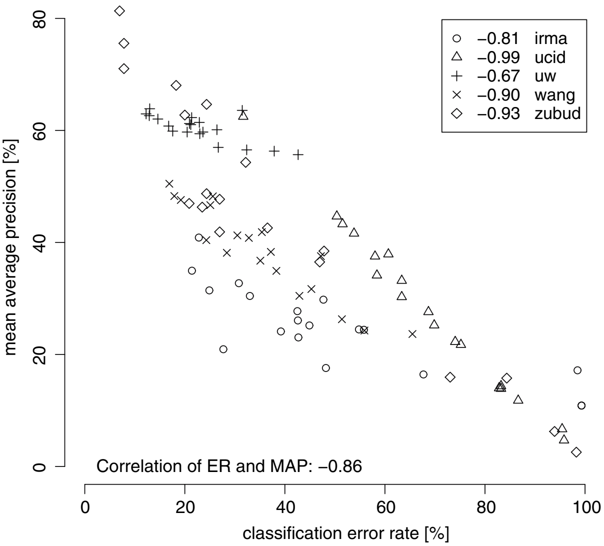


Fig. 10 Analysis of the correlation between classification error rate and mean average precision for the databases. The numbers in the legend give the correlation for the experiments performed on the individual databases

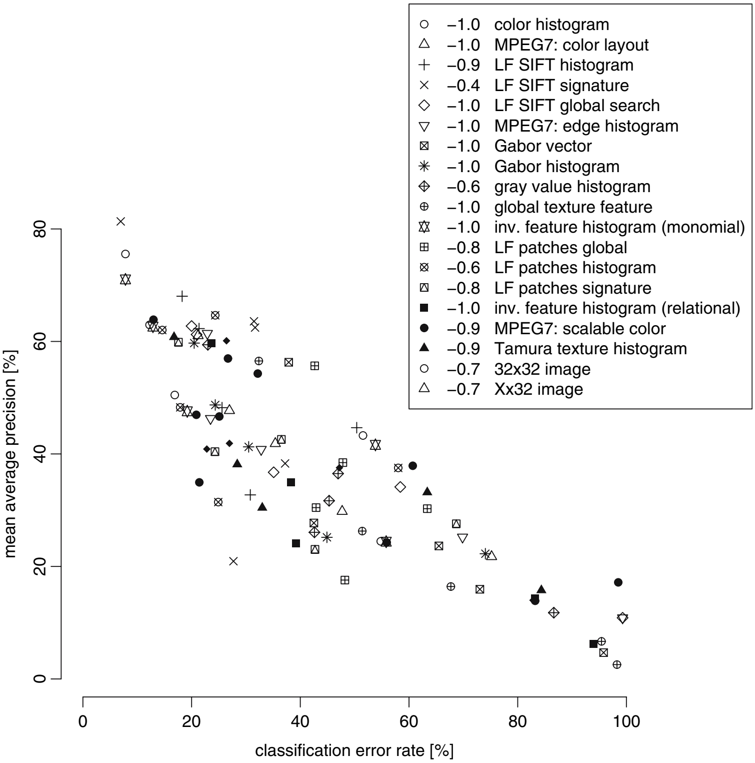


Fig. 11 Analysis of the correlation between classification error rate and mean average precision for the features. The numbers in the legend give the correlation for the experiments performed using the individual features

From Fig. 10 it can be seen that this correlation varies between the tasks between 0.99 and 0.67. For the UCID task, this correlation is markedly strong with 0.99. The correlation is lowest for the UW task which has a correlation of 0.67 and which is the only task with a correlation below 0.8.

In Fig. 11, the same correlation is analyzed feature-wise. Here, the correlation values vary strongly between 0.4 and 1.0. The LF SIFT signature descriptor has the lowest correlation and the LF patches histogram descriptor also has a low correlation of only 0.6. The two different image thumbnail descriptors have a correlation of 0.7. All other features have correlation values greater than 0.8, thus it can be said that an image representation that works well for classification will generally work well for CBIR as well and vice versa. Exemplarily, this effect can be observed when looking at the results for the WANG and IRMA database for the color histograms and the X × 32 thumbnails. On the one hand, for the WANG database, the color histograms perform very well for error rate and mean average precision; in contrast, the image thumbnails perform poorly. On the other hand, the effect is reversed for the IRMA database: here, the color histograms perform poorly and the image thumbnails outstandingly well. It can be observed that the performance increase (resp. decrease) is in the same magnitude for mean average precision and error rate. Thus, it can be seen that a feature that performs well for the task of classification on a certain dataset, it will most probably be a good choice for retrieval of images from that dataset, too.

7 Conclusion

We have discussed a large variety of features for image retrieval and a setup of five freely available databases that can be used to quantitatively compare these features. From the experiments conducted it can be deduced, which features perform well on which kind of task and which do not. In contrast to other papers, we consider tasks from different domains jointly and directly compare and analyze which features are suitable for which task.

Which features are suitable for which task in CBIR? The main question addressed in this paper, which features are suitable for which task in image retrieval, has been thoroughly investigated:

One clear finding is that color histograms, often cited as a baseline in CBIR, clearly are a reasonably good baseline for general color photographs. However, approaches using local image descriptors outperform color histograms in various tasks but usually at the cost of much higher computational costs. If the images are from a restricted domain, as they are in the IRMA and in the ZuBuD task, other methods should be considered as a baseline, e.g., a simple nearest neighbor classifier using thumbnails of the images.

Furthermore, it has been shown that, despite more than 30 years in research on texture descriptors, still none of the texture features presented can convey a complete description of the texture properties of an image. Therefore a combination of different texture features will usually lead to best results.

It should be noted that for specialized tasks, such as finding images that show certain objects, better methods exist today that can learn models of particular objects from a set of training data. However, these approaches are computationally far more expensive and always require relatively large amounts of training data.

Although the selection of features tested was not completely exhaustive, the selection was wide and the methods presented can easily be applied to other features to compare them to the features presented here. On one hand, the presented descriptors were selected such that features presented many years ago, such as color histograms (Swain and Ballard 1991), Tamura texture features (Tamura et al. 1978), Gabor features, and spatial autocorrelation features (Haralick et al. 1973), as well as very recent features such as SIFT descriptors (Lowe 2004) and patches (Deselaers et al. 2005) were compared. On the other hand, the features were selected such that descriptors accounting for color, texture, and (partly) shape, as well as local and global descriptors were covered. We also included a subset of the standardized MPEG7 features.

All features have been thoroughly examined experimentally on a set of five databases. All of these databases are freely available and pointers to their location are given in this paper. This allows researchers to compare the findings from this work with other features that were not covered here or which will be presented in future. The databases chosen are representative for four different tasks in which CBIR plays an important role.

Which features are correlated and how can features be combined? We conducted a correlation analysis of the features considered showing which features have similar properties and which do not. The outcomes of this method can be used as an intuitive help to finding suitable combinations of features for certain tasks. In contrast to other methods for feature combination, the method presented here does not rely on training data/relevance judgements to find a suitable set of features. In particular, it will tell you which features are not worth combining because they produce correlated distance results. The method is not a fully automatic feature selection method but the process of selecting features is demonstrated for one of the tasks with promising results. However, the focus of this paper is not to combine several features as this would exceed the scope and a variety of known methods have covered this aspect, e.g., (Yavlinski et al. 2004; Kittler 1998; Heesch and Rüger 2002).

Another conclusion we have drawn from this work is that the intuitive assumption that classification of images and CBIR are strongly connected is justified. Both tasks are strongly related to the concept of similarity which can be measured best if suitable features are available. In this paper, we have evaluated this assumption quantitatively by considering four different domains and analyzing the classification error rate for classification and the mean average precision for CBIR. It was clearly shown empirically that features that perform well for classification also perform well for CBIR and vice versa. This strong connection allows us to take advantage of knowledge obtained in either classification or CBIR for the other respective task. For example, in the medical domain much research has been done to classify whether an image shows a pathological case or not, likely some of the knowledge obtained in these studies can be transferred to the CBIR domain to help retrieving images from a picture archiving system.

Future Work. Future work in CBIR certainly includes finding new and better image descriptors and methods to combine these appropriately. Furthermore, the achievements in object detection and recognition will certainly find their way into the CBIR domain and a shift towards methods that automatically learn about the semantics of images is imaginable. First steps into this direction can be seen in (Nowak et al. 2007) where a method is presented that learns how to compare never seen objects and presents an image similarity measurement which works on the object level. Methods for automatic image annotation are also related to CBIR and the automatic generation of textual labels for images allows to use textual information retrieval techniques to retrieve images.

Acknowledgements This work was partially funded by the DFG (Deutsche Forschungsgemeinschaft) under contract NE-572/6. The authors would like to thank Gyuri Dorkó (formerly with INRIA Rhône-Alpes) for providing his SIFT feature extraction software and the authors of the MPEG7 XM reference implementation.

References

Antani, S., Kasturi, R., & Jain, R. (2002). A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video. *Pattern Recognition, 35*, 945–965.

Armitage, L. H., & Enser, P. G. (1997). Analysis of user need in image archives. *Journal of Information Science, 23*(4), 287–299.

Bloehdorn, S., Petridis, K., Saathoff, C., Simou, N., Tzouvaras, V., Avrithis, Y., et al. (2005). Semantic annotation of images and videos for multimedia analysis. In *European Semantic Web Conference (ESWC 05)*. Heraklian, Greece.

Bober, M. (2001). MPEG-7 Visual Shape Descriptors. *IEEE Trans on Circuits and Systems for Video Technology, 11*(6), 716–719.

Carson, C., Belongie, S., Greenspan, H., & Malik, J. (2002). Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(8), 1026–1038.

Clough, P., Mueller, H., Deselaers, T., Grubinger, M., Lehmann, T., Jensen, J., et al. (2006). The CLEF 2005 cross-language image retrieval track. In *Accessing Multilingual Information Repositories, 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005, vol. 4022 of Lecture Notes in Computer Science* (pp. 535–557). Vienna, Austria.

Clough, P., Müller, H., & Sanderson, M. (2004). The CLEF Cross Language Image Retrieval Track (ImageCLEF) 2004. In *Fifth Workshop of the Cross-Language Evaluation Forum (CLEF 2004), vol. 3491 of LNCS* (pp. 597–613).

Datta, R., Li, J., & Wang, J. Z. (2005). Content-based image retrieval—approaches and trends of the new age. In *ACM Intl. Workshop on Multimedia Information Retrieval, ACM Multimedia*. Singapore.

de Vries, A. P., & Westerveld, T. A. (2004). comparison of continuous vs. discrete image models for probabilistic image and video retrieval. In *Proc. International Conference on Image Processing* (pp. 2387–2390). Singapore.

Deselaers, T. (2003). Features for image retrieval. Master's thesis. Aachen: Human Language Technology and Pattern Recognition Group, RWTH Aachen University.

Deselaers, T., Hegerath, A., Keysers, D., & Ney, H. (2006). Sparse patch-histograms for object classification in cluttered images. In *DAGM 2006, Pattern Recognition, 27th DAGM Symposium, vol. 4174 of Lecture Notes in Computer Science* (pp. 202–211). Berlin.

Deselaers, T., Weyand, T., Keysers, D., Macherey, W., & Ney, H. (2006). FIRE in ImageCLEF 2005: Combining content-based image retrieval with textual information retrieval. In *Accessing Multilingual Information Repositories, 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005, vol. 4022 of Lecture Notes in Computer Science* (pp. 652–661). Vienna, Austria.

Deselaers, T., Weyand, T., & Ney, H. (2007). Image retrieval and annotation using maximum entropy. In C. Peters, P. Clough, F. Gey, J. Karlgren, B. Magnini, D. Oard, et al. (Eds.), *Evaluation of Multilingual and Multi-modal Information Retrieval—Seventh Workshop of the Cross-Language Evaluation Forum, CLEF 2000a, vol. 4730 of Lecture Notes in Computer Series* (pp. 725–734). Alicante.

Deselaers, T., Keysers, D., & Ney, H. (2004) Features for image retrieval—a quantitative comparison. In *DAGM 2004, Pattern Recognition, 26th DAGM Symposium, vol. 3175 of Lecture Notes in Computer Science* (pp. 228–236). Tübingen, Germany.

Deselaers, T., Keysers, D., & Ney, H. (2004). Classification error rate for quantitative evaluation of content-based image retrieval systems. In *International Conference on Pattern Recognition 2004 (ICPR 2004)* (Vol. 2, pp. 505–508). Cambridge.

Deselaers, T., Keysers, D., & Ney, H. (2005). FIRE— Flexible Image Retrieval Engine: ImageCLEF 2004 evaluation. In *Multilingual Information Access for Text, Speech and Images – Fifth Workshop of the Cross-Language Evaluation Forum, CLEF 2004, vol. 3491 of Lecture Notes in Computer Science* (pp. 688–698). Bath: Springer.

Deselaers, T., Keysers, D., & Ney, H. (2005). Discriminative training for object recognition using image patches. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 05)* (Vol. 2, pp. 157–162). San Diego.

Deselaers, T., Rybach, D., Dreuw, P., Keysers, D., & Ney, H. (2005). Face-based image retrieval-one step toward object-based image retrieval. In H. Müller & A. Hanbury (Eds.), *MUSCLE/ImageCLEF Workshop on Image and Video Retrieval Evaluation* (pp. 25–32). Vienna.

Di Sciascio, E., Donini, F. M., & Mongiello, M. (2002). Structured knowledge representation for image retrieval. *Journal of Artificial Intelligence Research, 16*, 209–257.

Dorkó, G. (2006) Selection of discriminative regions and local descriptors for generic. Object Class Recognition. Ph.D. thesis. Institut National Polytechnique de Grenoble.

Eidenberger, H. (2003). How good are the visual MPEG-7 features? In *Proceedings SPIE Visual Communications and Image Processing Conference* (Vol. 5150, pp. 476–488). Lugano.

Faloutsos, C., Barber, R., Flickner, M., Hafner, J., Niblack, W., & Petkovic, D., et al. (1994). Efficient and effective querying by image content. *Journal of Intelligent Information Systems, 3*(3/4), 231–262.

Fei-Fei, L., & Perona, P. (2005). A bayesian hierarchical model for learning natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition* (Vol. 2, pp. 524–531). IEEE: San Diego

Fergus, R., Perona, P., & Zissermann, A. (2003). Object class recognition by unsupervised scale-invariant learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 03)* (pp. 264–271). Blacksburg.

Fergus, R., Perona, P., & Zisserman, A. (2005). A sparse object category model for efficient learning and exhaustive recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 05)* (Vol. 2, pp. 380–389). IEEE: San Diego.

Forsyth, D. A., & Ponce, J. (2002) *Computer Vision: A Modern Approach* (pp. 599–619). Prentice Hall.

Gu, Z. Q., Duncan, C. N., Renshaw, E., Mugglestone, M. A., Cowan, C. F. N., & Grant, P. M. (1989). Comparison of techniques for measuring cloud texture in remotely sensed satellite meteorological image data. *Radar and Signal Processing, 136*(5), 236–248.

Haberäcker, P. (1995). *Praxis der Digitalen Bildverarbeitung und Mustererkennung*. München, Wien: Carl Hanser Verlag.

Hand, D., Manila, H., & Smyth, P. (2001). *Principles of data mining*. Cambridge: MIT Press.

Haralick, R. M., Shanmugam, B., & Dinstein, I. (1973). Texture Features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics, 3*(6), 610–621.

Heesch, D., & Rüger, S. (2002). Combining features for content-based sketch retrieval—a comparative evaluation of retrieval performance. In *European Colloquium on Information Retrieval Research, vol. 2291 of LNCS* (pp. 41–52). Glasgow, Scotland.

Heesch, D., & Rüger, S. (2003). Performance boosting with three mouse clicks—relevance feedback for CBIR. In *European Conference on Information Retrieval Research. No. 2633 in LNCS* (pp. 363–376). Pisa: Springer Verlag.

Iqbal, Q., & Aggarwal. J. (2002). CIRES: A system for content-based retrieval in digital image libraries. In *International Conference on Control, Automation, Robotics and Vision* (pp. 205–210). Singapore.

Jain, S. (2004). Fast image retrieval using local features: Improving approximate search employing seed-grow approach. Master's thesis. INPG, Grenoble.

Keysers, D., Deselaers, T., Gollan, C., & Ney, H. (2007). Deformation models for image recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(8), 1422–1435.

Kittler, J. (1998). On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*(3), 226–239.

Lehmann, T. M., Güld, M. -O., Deselaers, T., Keysers, D., Schubert, H., Spitzer, K., et al. (2005). Automatic categorization of medical images for content-based retrieval and data mining. *Computerized Medical Imaging and Graphics, 29*(2), 143–155.

Lew, M. S., Sebe, N., Djeraba, C., & Jain, R. (2006). Content-based multimedia information retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications and Applications, 2*(1), 1–19.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision, 60*(2), 91–110.

MacArthur, S. D., Brodley, C. E., & Shyu, C. -R. (2000). Relevance feedback decision trees in content-based image retrieval. In *Content-based access of image and video libraries* (pp. 68–72). IEEE: Hilton Head Island, SC.

Manjunath, B., Ohm, J. -R., Vasudevan, V. V., & Yamada, A. (2001). Color and texture descriptors. *IEEE Trans Circuits and Systems for Video Technology, 11*(6), 703–715.

Marée, R., Geurts, P., Piater, J., & Wehenkel, L. (2005) Random subwindows for robust image classification. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 34–40).

Markkula, M., & Sormunen, E. (1998). Searching for photos—journalists' practices in pictorial IR. In *Electronic Workshops in Computing—Hallenge of Image Retrieval* (pp. 1–13). Newcastle.

Meghini, C., Sebastiani, F., & Straccia, U. (2001). A model of multimedia information retrieval. *Journal of the ACM, 48*(5), 909–970.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., et al. (2005). A comparison of affine region detectors. *International Journal of Computer Vision 65*(1/2).

Müller, H., Müller, W., Squire, D. M., Marchand-Maillet, S., & Pun, T. (2000). Learning features weights from user behavior in Content-Based Image Retrieval. In S. Simoff & O. Zaiane (Eds.), *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Workshop on Multimedia Data Mining MDM/KDD2000).* Boston.

Müller, H., Müller, W., Marchand-Maillet, S., & Squire, D. M. (2000). Strategies for positive and negative relevance feedback in image retrieval. In *International Conference on Pattern Recognition, vol. 1 of Computer Vision and Image Analysis* (pp. 1043–1046). Barcelona.

Müller, H., Müller, W., Squire, D. M., Marchand-Maillet, S., & Pun, T. (2001). Performance evaluation in content-based image retrieval: overview and proposals. In H. Bunke & X. Jiang (Eds.), *Pattern Recognition Letters (Special Issue on Image and Video Indexing) 22*(5), 593–601.

Müller, H., Marchand-Maillet, S., & Pun, T. (2002) The truth about corel—evaluation in image retrieval. In *Proceedings of The Challenge of Image and Video Retrieval (CIVR2002), vol. 2383 of LNCS* (pp. 38–49). London.

Müller, H., Michoux, N., Bandon, D., & Geissbuhler A. (2004). A review of content-based image retrieval systems in medical applications–clinical benefits and future directions. *International Journal of Medical Informatics 73*(1), 1–23.

Najjar, M., Ambroise, C., & Cocquerez, J. -P. (2003). Feature selection for semi supervised learning applied to image retrieval. In *ICIP 2003* (Vol. 3, pp. 559–562). Barcelona.

Nölle, M. (2003). Distribution distance measures applied to 3-D object recognition—a case study. In *DAGM 2003, Pattern Recognition, 25th DAGM Symposium, vol. 2781 of Lecture Notes in Computer Science* (pp. 84–91). Magdeburg: Springer Verlag.

Nowak, E., & Jurie, F. (2007). Learning visual similarity measures for comparing never seen objects. In *CVPR 2007.* Minneapolis.

Obdrzalek, S., & Matas, J. (2003). Image retrieval using local compact DCT-Based representation. In *DAGM 2003, Pattern Recognition, 25th DAGM Symposium, vol. 2781 of Lecture Notes in Computer Science* (pp. 490–497). Magdeburg, Germany: Springer Verlag.

Ohm, J.-R. (2001). The MPEG-7 visual description framework—concepts, accuracy and applications. In *CAIP 2001*. No. 2124 in LNCS (pp. 2–10).

Opelt, A., Pinz, A., Fussenegger, M., & Auer, P. (2006). Generic object recognition with boosting. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 28*(3), 416–431.

Paredes, R., Perez-Cortes, J., Juan, A., & Vidal, E. (2001). Local representations and a direct voting scheme for face recognition. In *Workshop on Pattern Recognition in Information Systems* (pp. 71–79). Setúbal, Portugal.

Park, M., Jin, J. S., & Wilson, L. S. (2002). Fast content-based image retrieval using quasi-gabor filter and reduction of image feature. In *Southwest Symposium on Image Analysis and Interpretation* (pp. 178–182). Santa Fe.

Pentland, A., Picard, R., & Sclaroff, S. (1996). Photobook: Content-based manipulation of image databases. *International Journal of Computer Vision, 18*(3), 233–254.

Puzicha, J., Rubner, Y., Tomasi, C., & Buhmann, J. (1999). Empirical evaluation of dissimilarity measures for color and texture. In *International Conference on Computer Vision* (Vol. 2, pp. 1165–1173). Corfu, Greece.

Rubner, Y., Tomasi, C., & Guibas, L. J. (1998). A metric for distributions with applications to image databases. In *International Conference on Computer Vision* (pp. 59–66). Bombay.

Rui, Y., Huang, T., & Chang, S. (1999). Image retrieval: Current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation, 10*(4), 39–62.

Schaefer, G. (2004). CVPIC colour/shape histograms for compressed domain image retrieval. In *DAGM 2004. vol. 3175 of LNCS* (pp. 424–431). Tübingen, Germany.

Schaefer, G., & Stich, M. (2004) UCID-An uncompressed colour image database. In *Proc. SPIE Storage and Retrieval Methods and Applications for Multimedia* (pp. 472–480). San Jose.

Schmid, C., & Mohr, R. (1997). Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis & Machine Intelligence, 19*(5), 530–534.

Shao, H., Svoboda, T., & van Gool, L.(2003a). *ZuBuD—Zurich Buildings Database for image based recognition*. Computer Vision Lab, Swiss Federal Institute of Technology, Switzerland. Zurich, Switzerland.

Shao, H., Svoboda, T., Tuytelaars, T., & Gool, L. V. (2003b). HPAT indexing for fast object/scene recognition based on local appearance. In *Conference on Image and Video Retrieval. vol. 2728 of LNCS* (pp. 71–80). Urbana-Champaign: Springer Verlag.

Shirahatti, N. V., & Barnard, K. (2005). Evaluating image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 05)* (Vol. 1, pp. 955–961). IEEE: San Diego.

Siggelkow, S. (2002). Feature histograms for content-based image retrieval. Ph.D. thesis. University of Freiburg, Institute for Computer Science. Freiburg, Germany.

Siggelkow, S., Schael, M., & Burkhardt, H. (2001). SIMBA—Search IMages By Appearance. In *DAGM 2001, Pattern Recognition, 23rd DAGM Symposium, vol. 2191 of Lecture Notes in Computer Science* (pp. 9–17). Munich: Springer Verlag.

Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*(12), 1349–1380.

Smith, J. R., & Chang, S. -F. (1996). Tools and techniques for color image retrieval. In *SPIE Storage and Retrieval for Image and Video Databases* (Vol. 2670, pp. 426–437).

Squire, D. M., Müller, W., Müller, H., & Raki, J. (1999) Content-Based query of image databases, inspirations from text retrieval: Inverted files, frequency-based weights and relevance feedback. In *Scandinavian Conference on Image Analysis* (pp. 143–149). Kangerlussuaq.

Sun, Y., Zhang, H., Zhang, L., & Li, M. (2002). MyPhotos-A system for home photo management and processing. In *ACM Multimedia Conference* (pp. 81–82). Juan-les-Pins.

Swain, M. J., & Ballard, D. H. (1991). Color indexing. *International Journal of Computer Vision, 7*(1), 11–32.

Tamura, H., Mori, S., & Yamawaki, T. (1978). Textural features corresponding to visual perception. *IEEE Transaction on Systems, Man, and Cybernetics, 8*(6), 460–472.

Vailaya, A., Figueiredo, M. A. T., Jain, A. K., & Zhang, H. -J. (2001). Image classification for content-based indexing. *IEEE Transactions on Image Processing, 10*(1), 117–130.

van Gool, L., Tuytelaars, T., & Turina, A. (2001). Local features for image retrieval. In R. C. Veltkamp, H. Burkhardt, H.-P. Kriegel (Eds), *State-of-the-art in content-based image and video retrieval* (pp. 21–41). Kluwer Academic Publishers.

Vasconcelos, N., & Vasconcelos, M. (2004). Scalable discriminant feature selection for image retrieval and recognition. In *CVPR 2004. 2* (pp. 770–775). Washington.

Wang, J. Z., Li, J., & Wiederhold, G. (2001). SIMPLIcity: Semantics-sensitive integrated matching for picture LIbraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 23*(9), 947–963.

Yang, Z., & Kuo, C. (1999). Survey on image content analysis, indexing, and retrieval techniques and status report of MPEG-7. *Tamkang Journal of Science and Engineering, 3*(2), 101–118.

Yavlinski, A., Pickering, M. J., Heesch, D., & Rüger, S. (2004). A comparative study of evidence combination strategies. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004)* (Vol. 3, pp. 1040–1043). Montreal, Canada.

Zahedi, M., Keysers, D., Deselaers, T., & Ney, H. (2005). Combination of tangent distance and an image distortion model for appearance-based sign language recognition. In *DAGM 2005, Pattern Recognition, 26th DAGM Symposium, vol. 3663 of Lecture Notes in Computer Science* (pp. 401–408). Vienna, Austria.