Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 34 (2008) 266–279
Expert Systems with Applications
www.elsevier.com/locate/eswa

Image semantics discovery from web pages for semantic-based image retrieval using self-organizing maps

Hsin-Chang Yang [a,*], Chung-Hong Lee [b]

[a] Department of Computer Science and Information Engineering, Chang Jung University, Tainan 711, Taiwan
[b] Department of Electrical Engineering, National Kaohsiung University of Applied Science, Kaohsiung, Taiwan

Abstract

Traditional content-based image retrieval (CBIR) systems often fail to meet a user's need due to the 'semantic gap' between the extracted features of the systems and the user's query. The cause of the semantic gap is the failure of extracting real semantics from an image and the query. To extract semantics of images, however, is a difficult task. Most existing techniques apply some predefined semantic categories and assign the images to appropriate categories through some learning processes. Nevertheless, these techniques always need human intervention and rely on content-based features. In this paper we propose a novel approach to bridge the semantic gap which is the major deficiency of CBIR systems. We conquer the deficiency by extracting semantics of an image from the environmental texts around it. Since an image generally co-exists with accompanying texts in various formats, we may rely on such environmental texts to discover the semantics of the image. We apply a text mining process, which adopts the self-organizing map (SOM) learning algorithm as a kernel, on the environmental texts of an image to extract the semantic information from this image. Some implicit semantic information of the images can be discovered after the text mining process. We also define a semantic relevance measure to achieve the semantic-based image retrieval task. We performed experiments on a set of images which are collected from web pages and obtained promising results.

Keywords: Semantic-based image retrieval; Text mining; Self-organizing map

1. Introduction

Recently the task of image retrieval has received a great deal of attention from the web community since there are so many useful images on web pages. Image retrieval is a branch of information retrieval whose task is to retrieve some pieces of information (the *documents*) to fulfill a user's information needs according to certain (semantic) relevance measurements. Currently most information retrieval systems retrieve documents based on their 'contents'. That is, they measure the relevance between the query and a document according to internal representations or derived features. Such representations or features will vary for different document styles and retrieval schemes. For text retrieval systems, the contents are often represented by a set of selected keywords that are intended to capture the semantics of the documents. Many studies have successfully represented the semantics of text documents (Baeza-Yates & Ribeiro-Neto, 1999). For image retrieval systems, the representation of image content generally contains a set of visual features extracted from the image that hopefully may effectively represent the image. Many schemes have been proposed to describe the image contents (De Marsicoi, Cinque, & Levialdi, 1997; Doermann, 1998; Gupta & Jain, 1997). However, the semantics of an image are difficult to be revealed by these features, so that irrelevant images are retrieved even when they have

* Corresponding author.
E-mail addresses: hcyang@mail.cju.edu.tw (H.-C. Yang), leechung@mail.ee.kuas.edu.tw (C.-H. Lee).
URL: http://www.im.cju.edu.tw/hcyang (H.-C. Yang).

0957-4174/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2006.09.016

similar features. For example, an image with 70% green color and 30% blue color could be either a scenic view of a meadow or a book cover. Another example is a round object with a hole in it, which could either be a wheel or a compact disc. These examples show that a user may easily obtain images that are totally irrelevant to his query through CBIR approaches. This difference between the user's information need and the image representation is called the 'semantic gap' in CBIR systems. Thus CBIR systems only work well in relatively small domains of data sets, and to obtain more reasonable results, semantic-based image retrieval (SBIR) systems must be devised to bridge the semantic gap.

Unlike CBIR systems, which use 'contents' to retrieve images, SBIR systems try to discover the real semantic meaning of an image and use it to retrieve relevant images. However, understanding and discovering the semantics of a piece of information are high-level cognitive tasks, and thus hard to automate. Several attempts have been made to tackle this problem. Most of these methods use CBIR techniques such that primitive features are used to derive higher order image semantics. However, CBIR systems use no explicit knowledge about the image and limit their applications to fields such as fingerprint identification and trade mark retrieval, etc. To increase user satisfaction with query results, we must incorporate more semantics into the retrieval process. However, there are three major difficulties in doing so. The first is that we must have some kind of high-level description scheme for describing the semantics. The second is that a semantics extraction scheme is necessary to map visual features to high-level semantics. Finally, we must develop a query processing and matching method for the retrieval process. Many techniques have been devised to remedy these difficulties, as we discuss later. In this work we propose a novel approach to solve these difficulties using a simple framework. First we incorporate explicit knowledge into the framework by representing the images with their surrounding texts in the web pages. Such representation also solves the difficulty of semantic representation. The semantic extraction process is achieved in our framework by using a text mining process on these texts. We also design a semantic relevance measure for matching the user's query and images in the image collection, which solves the third difficulty. Our idea comes from the recognition that it is too difficult to directly extract semantics from images. Thus we avoid direct access of the image contents, which is generally time-consuming and imprecise. Instead, we try to obtain image semantics by their environmental texts, which are generally contextually relevant to the images.

The paper is organized as follows. In Section 2 we review some related works on semantic-based image retrieval and text mining. We then describe the text mining process and its application on semantic image retrieval in Section 3. In Section 4 we show some experimental results, and in the last section we give some conclusions.

2. Related studies

We review some work in different aspects related to our work.

2.1. Text mining studies

First we discuss some related work about the text mining technique used in this work. Several research efforts at creating enhanced text mining techniques were developed to make the system and mining process more effective. Feldman uses text category labels (associated with the Reuters newswire service) to find 'unexpected patterns' among text articles (Dagan, Feldman, & Hirsh, 1996; Feldman & Dagan, 1995; Feldman, Dagan, & Hirsh, 1998; Feldman, Klosgen, & Zilberstein, 1997). In their systems, all documents are labeled by keywords, and thus the knowledge discovery task is carried out by analyzing the co-occurrence frequencies of the various keywords labeling the documents. The main approach is to compare distributions of category assignments within subsets of the document collection. Another related project can be found in so-called On-line New Event Detection (Allan, Carbonell, Doddington, Yamron, & Yang, 1998), whose input is a stream of news stories in chronological order, and whose output is a yes/no decision for each story, made at the time the story arrives, indicating whether or not the story is the first reference to a newly occurring event. In other words, the system must detect the first instance of what will become a series of reports on an important topic. Although this can be viewed as a standard classification task (where the class is a binary assignment to the new-event class) it is more in the spirit of data mining, in that the focus is on discovery of the beginning of a new theme or trend. Both the above approaches are built upon the uses of pre-defined text metadata associated with text documents, rather than the text content themselves. A project that aims at constructing methods for exploring full-text document collections, the WEBSOM (Honkela, Kaski, Lagus, & Kohonen, 1996; Kaski, Honkela, Lagus, & Kohonen, 1998; Kohonen, 1998), initiated from Honkela's suggestion to use "self-organizing semantic maps" (Ritter & Kohonen, 1989) as a preprocessing stage for encoding documents. Such maps are, in turn, used to automatically organize (i.e. cluster) documents according to the words that they contain. When the documents are organized on a map in such a way that nearby locations contain similar documents, exploration of the collection is facilitated by the intuitive neighborhood relations. Thus, users can easily navigate a word category map and zoom in on groups of documents related to a specific group of words.

2.2. SBIR schemes

To describe image semantics, Eakins (1996) classified queries into three levels, according to their abilities of abstraction. The lowest level, Level 1, comprises primitive

features such as color, texture, and shape. Essentially this level uses no semantic information in an image. Most works on CBIR relate to this level. Level 2 comprises derived attributes which involve some degree of logical inference about the identity of the objects depicted in the image. This level includes object semantics such as object classes and spatial relations among objects. Those systems which can resolve this level of queries are considered as retrieval by semantic content. The third level comprises retrieval by abstract attributes, involving a high degree of abstract – and possibly subjective – reasoning about the meaning and purpose of the objects or scenes depicted. Examples in this level include scene semantics, behavior semantics, and emotion semantics. Eakins's classification is helpful for describing the capabilities and limitations of different retrieval techniques. We often refer to the retrieval tasks that fulfill the Level 2 and 3 queries as semantic image retrieval (Gudivada & Raghavan, 1995), and the difference between Level 1 and 2 as "semantic gap".

Most of the semantic extraction methods use a multi-level abstraction scheme, as suggested by Al-Khatib, Day, Ghafoor, and Berra (1999). This extraction process is divided into three layers, namely feature extraction layer, object recognition layer, as well as semantic modeling and knowledge representation layer. The feature extraction layer deals with low-level image processing in order to find portions of the raw data which match the user's requested pattern. In the object recognition layer features are analyzed to recognize objects and faces in an image database. The major function of the last layer is to maintain the domain knowledge for representing spatial semantics associated with image databases. Since our method uses no image content in the retrieval process, we cannot classify this method as pertaining to Eakins's levels or Al-Khatib's layers.

2.3. Image semantics representation

There are several types of representation schemes for image semantics, and the first is textual representation schemes. Hermes, Klauck, Kreyβ, and Zhang (1995) use a similarity technique to derive the natural language description of a outdoor scene image in the IRIS system. The textual description is generated by four sub-steps: extraction of features like colors, textures, and contours, segmentation, and interpretation of part-whole relations. Such text description is then processed by traditional text retrieval techniques. However, such textual representation is insufficient to represent the complex relations among concepts, and is only applicable to specific applications. The second type uses traditional knowledge representation schemes such as semantic networks, predicate logic, and frames. Recently several researchers have devised different models for semantics representation. Examples are the fuzzy boolean model (Zhuang, Mehrotra, & Huang, 1999), formal language theory (Colombo, DelBimbo, & Pala, 1999), fuzzy logic (Meghini, 1997), and semiotic approaches (Cavazza, Green, & Palmer, 1998). These approaches are capable of representing and matching semantics in specific domains. However, there is still no approach that can be applied in the general domain.

2.4. Web image semantics extraction

The effectiveness of the proposed method relies on the correct annotations of an image. Several works adopt similar annotation schemes for image retrieval. Chen, Liu, Zhang, Li, and Zhang (2001) adopted an unified approach which combines low-level visual features, such as color, texture, and shape, and high-level semantic text features. The high-level text features they used are similar to those in this work, including image filename and URL, ALT text, surrounding text, page title, etc. Sclaroff, Cascia, Sethi, and Taycher (1999) proposed a system that combines textual and visual statistics in a single index vector for content-based search of a Web image database. In their method, textual statistics are captured in vector form using latent semantic indexing based on text in the containing HTML document. Visual statistics are captured in vector form using color and orientation histograms. The system assigns different weights to the words appearing in the title and headers and in the ALT fields of the `img` tags along with words emphasized with different fonts like bold and italics. However, these weight values were chosen heuristically, and they did not consider the proximity weighting. There is a number of systems that try to assign texts to images to achieve Web image retrieval, as we discussed here. WebSEEk proposed by Chang, Smith, Beigi, and Benitez (1997) uses only Web URL addresses and HTML tags associated with the images to extract the keywords. Harmandas, Sanderson, and Dunlop (1997) use the text after an image URL until the end of a paragraph or until a link to another image is encountered as the caption of the image. The Marie-3 system (Rowe & Frew, 1998) uses text 'near' an image to identify a caption. 'Nearness' is defined as within a fixed number of lines in the parse of the source HTML file. WebSeer (Frankel, Swain, & Athitsos, 1996) defines the caption of an image to be the text in the same `center` tag as the image, within the same cell of a table as an image or the same paragraph. However, none of these techniques can perform reasonably well on all types of HTML pages.

2.5. Image semantics matching

To match the semantics between the query and an image in the database, Chang, Chen, and Sundaram (1998) proposed a semantic visual template model, which is used to approximate user queries. In this, the user is asked to identify a possible range of colors, textures, shapes or motion parameters to express his or her query, which is then refined using relevance feedback techniques. When the user is satisfied, the query is given a semantic label (such as "sunset") and stored in a query database for later use. Over time, this query database becomes a kind of visual thesaurus, linking

each semantic concept to the range of primitive image features most likely to retrieve relevant items. However, this method requires an initial sketch, which is hard for naive users to generate. In addition, generating a semantic visual template is time-consuming and needs human intervention, making it unsuitable for large datasets. Dori and Hel-Or (1998) proposed the visual-object process diagram (VOPD) which incorporates both low-level image features and texture key sentences as descriptors of the image. Querying is performed by representing the sought image with a VOPD and finding the images in the database whose VOPDs best match the query. However, generation of VOPDs for both query and images in the database is a manual task which is time-consuming and requires knowledge about the CASE tool. Ojala, Rautiainen, Matinmikko, and Aittola (2001) used HSV autocorrelograms for image retrieval. However, the semantics in their work was designated manually but not automatically. Zhao and Grosky (2002) adopted the latent semantic indexing (LSI) (Deerwester, Dumais, Furnas, & Landauer, 1990) technique to find the latent correlation between low-level visual features and high-level semantics. Srihari and Burhans (1994) proposed a visual semantics model which is closely related to our work. However, although they discover image semantics from the text accompanying the images as we do, only picture captions are used, which makes their system, PICTION, apply only to limited domains. Wu, Sitharama Iyengar, and Zhu (2001) adopt the SOM learning for web image retrieval. They use various kinds of features such as textual feature, color histogram, image entropy, shape, and texture feature. However, they did not provide evaluations on the results. Laaksonen, Koskela, Laakso, and Oja (2000) proposed PicSOM system which uses Tree structured self-organized maps for CBIR, and uses no semantic features. The major difference between the proposed method and the above two methods is the text mining process, which is the key to bridge the semantic gap. Chen et al. (2001) proposed a web mining approach for image retrieval. They analyzed the user feedback log to improve image representations and discover the relationship between low-level and high-level features. Their method belongs to the log mining category in web mining techniques, and is different to our method which belongs to the content mining category.

3. Text mining for semantic image retrieval

In this section we give a detailed description of the proposed method, starting from the preprocessing and encoding of documents in Section 3.2. The encoded documents are then trained by the self-organizing map algorithm in Section 3.3. In this section a text mining process is also applied to the training results to discover the relationships among images as well as the subjects of the images, which are considered as a kind of semantics of the images. Finally, we use the text mining results to retrieve images in Section 3.4.

3.1. System overview

We first give a brief sketch of the proposed method here. The functional components and the processing flow are depicted in Fig. 1. In the preprocessing and feature generation stages, we collect suitable web pages and transform them into feature vectors, which are indexed and stored in a database. Next we apply the proposed text mining process on this set of feature vectors, and obtain the image cluster map (ICM) and keyword cluster map (KCM). In the final stage we match the user query with each image and obtain the ranking of the images. The dotted lines enclose those functional components of each stage.

3.2. Document preprocessing

A document in our corpus is a typical web page, which contains both texts and images. A web page generally exists in HTML format which uses a set of predefined tags to perform actions such as typesetting the page, inserting hyperlinks and multimedia objects, etc. In this article we are interested in image retrieval, so we first extract images from the documents, and then extract environmental texts from the web pages corresponding to each extracted image. Note that a web page is discarded if it contains no qualified image or environmental texts.

3.2.1. Extracting images from web pages

In HTML we use the <img> tag to add images to the web pages. Thus we can extract images from the web pages by examining the <img> tags. We discard extremely small images because they often contain little information. For example, many web pages contain small icons such as buttons and rulers that are used for alignment, segmentation, and marking, etc. The threshold of image size is determined experimentally by collecting all images and then determining a threshold that can discriminate such small images. A typical threshold value is a few kilobytes. Other than image size, there is almost no limit on the image properties in our method since we do not directly extract features from the images. Notice that we treat duplicated images as separate ones to allow different interpretations of the same images.

3.2.2. Extracting environmental texts

The remaining part of the web page is further processed to extract necessary texts. These texts are called the environmental texts (ETs) of these images. There are several types of environmental texts related to an image. For example, we can identify nearby texts, captions, alternate texts, and filenames from an HTML-format web page. In this article, two types of ETs are extracted, namely $ET_{Caption}$ and ET_{Normal}. $ET_{Caption}$ includes texts that are regulated by HTML tags, such as image URL and ALT text. On the other hand, ET_{Normal} consists of ordinary texts which are located outside any HTML tags. $ET_{Caption}$ is easy to be extracted using the <img> tags. However, the

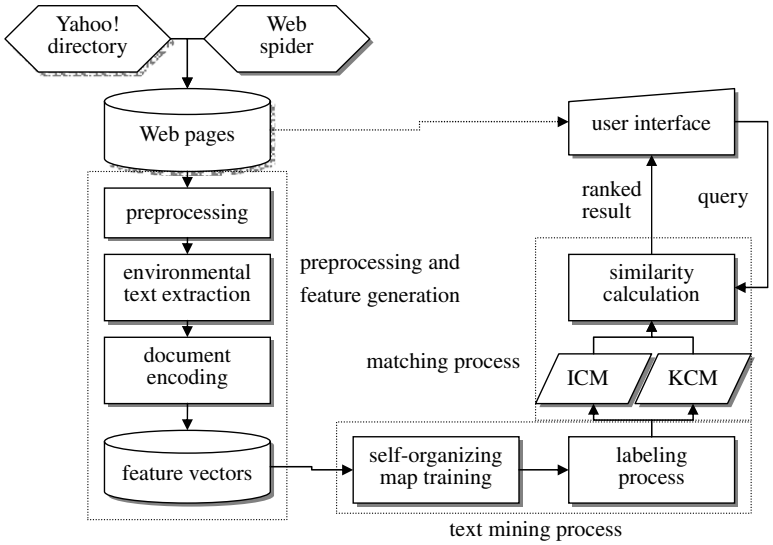


Fig. 1. The processing stages and functional components of the proposes method.

extraction of ET_{Normal} is considerably difficult since there exists much ambiguity in texts that around an image. That is, it is rather difficult to determine which part of texts is relevant to this image. To resolve such ambiguity, we adopt the scheme that was used by Mukherjea and Cho (1999). In their work, they used criteria such sa visual distance, syntactic distance, regular patterns in a table, and groups of images to assign text to images. According to their report, visual distance, regular pattern and image group criteria have high accuracy in assigning text to images. Therefore, we extract ET_{Normal} according to these criteria. When both the ET_{Normal} and $ET_{Caption}$ associated with each image have been extracted, a word extractor is used to segment these ETs into a list of terms. We discard those terms in a standard stoplist to reduce the vocabulary size. We also discard terms that occur only once in the ETs. We use the resulting list of terms to represent the image. We call these terms the environmental keywords (EKs) of their associated image. Fig. 2 shows the ETs and EKs of an image extracted from an example web page.

3.2.3. Encoding images

We adopt a binary vector representation similar to the vector space model to encode the images. All the ETs associated with an image are collectively transformed to a binary vector such that each component corresponds to an EK associated with this image. A component of the vector with value 1 or 0 indicates the presence or absence of an EK in its associated document, respectively. We do not use any term weighting method such as the *tf·idf* scheme (Baeza-Yates & Ribeiro-Neto, 1999).

One problem with this encoding method is that if the vocabulary is very large, then the dimensionality of the vector is also high. In practice, the resulting dimensionality of the space is often huge, since the number of dimensions is determined by the number of distinct index terms in the corpus. In general, feature spaces on the order of 1000 to 100000 are very common for even reasonably small collections of documents. As a result, techniques for controlling the dimensionality of the vector space are required. Such a problem could be solved, for example, by eliminating some of the most common and some of the rarest indexed terms

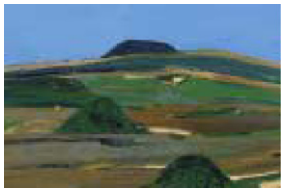
image	environmental texts	environmental keywords
	*ET-Normal:* "The Fleurieu Prize for Australian Landscape 2002." "by Joe Furlonger" *ET-Caption:* "../images/ 2002winner_furlonger.jpg"	Fleurieu Prize Australian Landscape Joe Furlonger images winner jpg

Fig. 2. The ETs and EKs of the indicated image in the web page with URL:http://www.artprize.com.au/fppages/lnew_winners.htm. In ET_{Normal} we only include the paragraph where the image occurs.

in the preprocessing stage. Several other techniques may also be used to tackle the problem. Examples are multidimensional scaling (Cox & Cox, 1994), principal component analysis (Jolliffe, 1986), and latent semantic indexing (Deerwester et al., 1990).

3.3. Discovering image semantics by text mining

3.3.1. Document clustering using self-organizing maps

In this subsection we will describe how to organize images and EKs into clusters by their co-occurrence similarities. The images in the corpus are first encoded into a set of vectors as described in Section 3.2. We intend to organize these images into a set of clusters such that similar images will fall into the same cluster. Moreover, similar clusters should be 'close' in some manner. The unsupervised learning algorithm of SOM networks (Kohonen, 1997) matches our needs. The SOM algorithm organizes a set of high-dimensional vectors into a two-dimensional map of neurons according to the similarities among the vectors. Similar vectors, i.e. vectors within small distances, will map to the same or nearby neurons after the training (or learning) process. Furthermore, the similarity between vectors in the original space is preserved in the mapped space. Applying the SOM algorithm to the image vectors, we actually perform a clustering process about the images in the corpus in a sense that a neuron in the map can be considered as a cluster. Similar images will fall into the same or neighboring neurons (clusters). Furthermore, the similarity of two clusters can be measured by the geometrical distance between their corresponding neurons. To determine the cluster to which an image or an EK belongs, we apply a labeling process to the images and the EKs, respectively. After the labeling process, each image is associated with a neuron in the map. We record such associations to form the image cluster map. In the same manner, we also label each EK to the map and form the keyword cluster map. We then use these two maps for image semantics discovery.

We next define some denotations and describe the training process. Let $\mathbf{x}_i = \{x_{i_n} \in \{0,1\} | 1 \leqslant n \leqslant N\}, 1 \leqslant i \leqslant M$, be the encoded vector of the ith image in the corpus, where N and M are the total number of EKs and images in the corpus, respectively. We use these vectors as the training inputs to the SOM network. The network consists of a regular grid of neurons which has N synapses each. Let $\mathbf{w}_j = \{w_{j_n} | 1 \leqslant n \leqslant N\}, 1 \leqslant j \leqslant J$, be the synaptic weight vector of the jth neuron in the network, while J is the number of neurons in the network. We train the network using the SOM algorithm:

Step 1: Randomly select a training vector $\mathbf{x}_i$.

Step 2: Find the neuron j with synaptic weights $\mathbf{w}_j$ which is closest to $\mathbf{x}_i$, i.e.

$$\|\mathbf{x}_i - \mathbf{w}_j\| = \min_{1 \leqslant k \leqslant J} \|\mathbf{x}_i - \mathbf{w}_k\|. \quad (1)$$

Step 3: For every neuron l in the neighborhood of neuron j, update its synaptic weights by

$$\mathbf{w}_l^{\text{new}} = \mathbf{w}_l^{\text{old}} + \alpha(t)(\mathbf{x}_i - \mathbf{w}_l^{\text{old}}), \quad (2)$$

where $\alpha(t)$ is the training gain at time stamp t.

Step 4: Increase the time stamp t. If t reaches the preset maximum training time T, halt the training process; otherwise decrease $\alpha(t)$ and the neighborhood size, goto Step 1.

The neighborhood of neuron j in Step 3 is a set of neurons that is geometrically close to this neuron. The closeness is often defined according to the geometrical distance between each pair of neurons. Generally the initial distance for two neurons being neighbors is about half of the dimension of the map. The training process stops after time T, which is sufficiently large so that every vector may be selected as training input for certain times. The training gain and neighborhood size both decrease as t increases.

3.3.2. Mining image and keyword associations

When the image clustering process is accomplished, we will perform a labeling process on the trained network to establish the association between each image and one of the neurons. The labeling process is described as follows. Each image's feature vector $\mathbf{x}_i$, $1 \leqslant i \leqslant M$ is compared to every neuron's weight vector in the map. We will *label* the ith image to the jth neuron if they satisfy Eq. (1). After the labeling process, each image is labeled to some neuron or, from a different point of view, each neuron is labeled by a set of images. We record the labeling results and obtain the ICM. In the ICM, each neuron is labeled by a list of images which are considered similar and are in the same cluster. That is, the SOM algorithm should cluster images.

Next we explain why the SOM algorithm performs a clustering process. In the labeling process those images which contain similar EKs will map to the same or neighboring neurons because they have relatively 'similar' feature vectors. In the mean time, since the number of neurons is usually much smaller than the number of the images in the corpus, multiple images may be labeled to the same neuron. Thus a neuron forms an image cluster. Besides, neighboring neurons should represent image clusters of similar context, i.e. high EK co-occurrence frequency. On the other hand, it is possible that some neurons may not be labeled by any image. We call these neurons *unlabeled neurons*, and they occur in two situations. One is when the number of images is relatively small compared to the number of neurons. Another situation is when the corpus contains too many semantically similar images so that a great part of images will fall into a small set of neurons. However, unlabeled neurons will not diminish the result of the clustering since they do not affect the similarity measurement between any pair of clusters.

We construct the KCM by labeling each neuron in the trained network with certain EKs. Such labeling is achieved by examining the neurons' synaptic weight

vectors and is based on the following observation. Since we use a binary representation for the image feature vectors, ideally the trained map should consist of synaptic weight vectors with many component values near either 0 or 1. Since a value of 1 in an image vector indicates the presence of corresponding EK of that image, a component with value near 1 in a synaptic weight vector also shows that such neuron has recognized the importance of the EK and tries to 'learn' the EK. According to such interpretation, we design the following keyword labeling process. For the weight vector of the jth neuron $\mathbf{w}_j$, if its nth component exceeds a predetermined threshold, the corresponding EK of that component is labeled to this neuron. To achieve better results, the threshold is a real value near 1. By virtue of the SOM algorithm, a neuron may be labeled by several EKs which often co-occur in a set of images. Thus a neuron forms a keyword cluster. A schematic drawing of the keyword labeling process is depicted in Fig. 3. The labeling method may not completely label every EK in the corpus. We call these EKs the *unlabeled keywords*. Unlabeled keywords occur in two circumstances. One is when several neurons compete for an EK during the training process. The competition often results in imperfect convergence of weights, so some EKs may not be learned well, i.e. their corresponding components may not have values near 1 in any neuron's weight vector. The other circumstance is when an EK is not universal in that neuron's labeled images. Since the EK is not present in all input vectors that associated with this neuron, the corresponding component for that EK should not always be 1 during training. Thus the component value will not be high and the EK will not be labeled to this neuron. We solve the unlabeled keywords problem by examining all the neurons in the map and labeling each unlabeled keyword to the neuron with the largest value of the corresponding component for that EK. That is, the nth EK is labeled to the jth neuron if

$$w_{j_n} = \max_{1 \leqslant k \leqslant J} w_{k_n}, label_j \neq \emptyset, \tag{3}$$

where $label_j$ is the set of images that labeled to neuron j.

The above keyword labeling process may be further extended to include more neurons in deciding the importance of an EK for a neuron. For example, we may label an EK to a neuron j when its overall corresponding component value over a set of neighboring neurons exceeds the threshold. These neighboring neurons are selected according to their distances to the neuron j. We may denote the previous scheme as an 0-order keyword labeling process, whereas the extended version is a k-order labeling process where k is the maximum distance allowed for those neurons which are considered neighbors. Fig. 4 depicts the neighboring neurons for different orders. The overall corresponding component value is the average of the component values of all neighboring neurons. We can also apply such an extension to the unlabeled keywords. In this work we only apply the 0-order process.

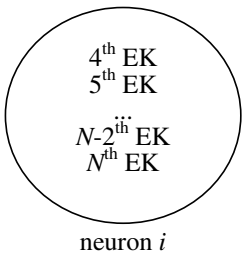


Fig. 3. A keyword is labeled to a neuron when its corresponding component in the neuron's weight vector has a value above a threshold which is near 1. In this figure we set the threshold to 0.9.

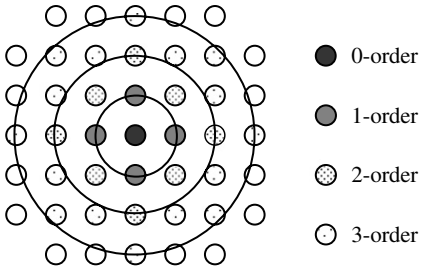


Fig. 4. The neighborhood selection for different orders.

The KCM autonomously clusters EKs according to their similarity of co-occurrence. Words that tend to occur simultaneously in the ETs of the same image will be mapped to neighboring neurons in the map. For example, the words "LCD" and "monitor" often occur simultaneously in the ETs of an image. They will map to the same neuron, or neighboring neurons, in the map because their corresponding components in the encoded image vector are both set to 1. Thus a neuron will try to learn these two EKs simultaneously. On the contrary, EKs that do not co-occur in the ETs of the same document will map to distant neurons in the map. Thus we can reveal the relationship between two EKs according to their corresponding neurons in the KCM.

3.4. Image retrieval by semantics

After obtaining the image clusters and keyword clusters, we may use them for semantic image retrieval. Two types of query methods are allowed in our scheme, the keyword-based queries and query by example, as discussed below.

3.4.1. Semantic image retrieval by keywords

When a user specifies keywords as a query, the images that are semantically relevant to this query can be retrieved by using the KCM. For this, the query is transformed to a query vector in the same way as the image vector. Here let $\mathbf{q} = \{q_i \in \{0,1\} | 1 \leqslant i \leqslant N\}$ denote the query vector. We also transform each keyword cluster in KCM to a vector, as follows: let $\mathbf{k}_j = \{k_{j_i} \in \{0,1\} | 1 \leqslant i \leqslant N\}$ be the encoded

vector for the keyword cluster associated with neuron j, where $k_{j_i} = 1$ when the ith EK is labeled to neuron j. The similarity between the query vector $\mathbf{q}$ and an image vector $\mathbf{x}$ is calculated with an extension of the cosine measurement in the vector space model:

$$S_{\mathbf{q},\mathbf{x}} = A\frac{|\mathbf{q}\cdot\mathbf{k}_j|}{\|\mathbf{q}\|\|\mathbf{k}_j\|} + \frac{|\mathbf{q}\cdot\mathbf{x}|}{\|\mathbf{q}\|\|\mathbf{x}\|}. \quad (4)$$

We let $\mathbf{x}$ be the encoded vector of some image that is associated with neuron j. The first term of the right hand side (RHS) of Eq. (4) measures the similarity between the query vector and the cluster vector. The second term measures the similarity between the query vector and the given image vector associated with neuron j. A is a scaling parameter that is big enough to differentiate the contributions from cluster and individual image. That is, A must be big enough so that the size of the second term will not greatly exceed the size of the first term and thus change the rank of a cluster. The similarity between the query and each individual image (i.e. the second term) is used only to differentiate the images in the same cluster. For example, let the query be "semantic retrieval" and let neuron 20 be labeled by the keywords "semantic", "image", and "retrieval". We also let this neuron be labeled by images 1, 7, 20, and 34. The first term of the RHS of Eq. (4) may find that neuron 20 is the closest to the query. However, the rank of the four images labeled to this neuron is determined by the second term in Eq. (4). A typical value of A should be the maximum number of images associated with a cluster on the map. Since a well-trained map should have an almost even distribution of images over the neurons, a reasonable value of A is $\frac{B\cdot M}{J}$ where B is a positive number that is greater than 1, M is the number of images in the corpus, and J is the number of neurons in the SOM. The parameter B should not be too small to allow situations of uneven distributions of images.

With parameter $A = 0$ in Eq. (4), we in effect compute the similarity between the query vector and each image vector. This situation requires no knowledge from the text mining process. Setting $A = 0$ reduces our method to the classic vector space model, so we will use this setting only for benchmark purpose.

We can also design a different kind of measurement which uses the geometrical distance between the neurons to measure the similarity. The self-organizing map algorithm clusters images according to the degree of similarity between their encoded vectors. By virtue of the SOM algorithm, similar images will be labeled to the same neuron or nearby neurons. Thus we may measure the geometrical distance between the associated neurons of two images and use it to define the similarity between them. The similarity is measured as follows:

$$S'_{\mathbf{q},\mathbf{x}} = \mathscr{B}(\|G(\mathbf{q}) - G(\mathbf{x})\|) + \frac{|\mathbf{q}\cdot\mathbf{x}|}{\|\mathbf{q}\|\|\mathbf{x}\|}, \quad (5)$$

where function $G: \boldsymbol{R}^N \to \boldsymbol{R}^2$ returns the two-dimensional grid coordinates of its argument in the ICM. The argument of G is an encoded image vector or the query vector. When a vector is passed to G as an argument, we first find its associated neuron, as in the labeling process. We then compute the two-dimensional coordinates of the neuron. As an example, let an image vector $\mathbf{x}$ be labeled to neuron 10, and let the map contain 8×8 neurons. Thus G will return (10 div 8, 10 mod 8) = (1, 2) as result. Furthermore, $\|\cdot\|$ denotes the Euclidean distance function, and $\mathscr{B}$ is a Gaussian-like function which has maximum output when its argument is equal to zero. In this case, the first term in the RHS of Eq. (5) computes a kind of 'semantic distance' which is discovered by the text mining process. The second term is used to differentiate the images that are labeled to the same neuron. The scale of $\mathscr{B}$ should be carefully adjusted to balance the contributions from the two terms, and a typical setting for $\mathscr{B}$ is similar to that for A discussed above.

3.4.2. Semantic image retrieval by example web page images

When an image is used as a query, we should first find its ETs from its associated web page, as described in Section 3.2.2. To eliminate the necessity of presenting a whole page as query, which is different from the ordinary usage, we should first associate each image with a representative web page or a set of web pages and extract ETs from this set of web pages. With this kind of transformation, the retrieval process is basically the same as described in Section 3.4.1. That is, the ETs are transformed to a query vector as in Section 3.2.3 and the similarity between the query image and an image in the corpus is measured by Eq. (4) or Eq. (5).

We now discuss some alternate approaches. To speed up the matching process, we may divide the similarity computation into two stages. In the first stage, we compute only the first terms in Eqs. (4) and (5). Only those clusters that have the highest results will be used in the second stage. In the second stage, we order the images according to the second terms. This approach takes advantage of the fact that the number of neurons, J, is generally much smaller than the number of images, M. Thus in the first stage we only need to compute J similarities instead of M. In the second stage we may need ten or twenty more computations of the second term. Since the user is generally interested only in the top-rank images, this two-stage scheme may achieve adequate results.

4. Experimental result

Before conducting experiments for the proposed method we notice that there is no generally recognized benchmark procedure for the semantic image retrieval because it is hard to automatically decide the real semantics of a piece of information. In general, there are two types of schemes in evaluating semantics, namely human evaluation and automatic evaluation. In human evaluation we should present the retrieval results for a standard set of queries to a set of human subjects and ask for their judgements. This scheme is considered the most accurate since human

interpretation is the ultimate judge of the semantics. However, it is time-consuming and hard to control due to the wide variety of human perceptions. On the other hand, automatic evaluation uses a pre-determined evaluation process and measurement to generate the evaluation results. Although this evaluation process is generally fast and consistent, the evaluation measurement is always questionable. In this work, we propose an automatic evaluation process and measurement to make our method comparable with others. We also conducted a human evaluation process to verify the validity of the proposed method. Finally, the quality of extracted features is also evaluated against a similar system, PicSOM (Laaksonen et al., 2000).

We test the method with a set of web pages which were collected manually according to the Yahoo! web site directory. The Yahoo! directory hierarchy was used because it has been a standard test bed for categorization and semantics development of web pages, and many other works have used the Yahoo! hierarchy in their experiments. One advantage of the Yahoo! hierarchy is that it is constructed by human linguists and domain experts, thus making it "semantically correct". That is, web pages that have been assigned to the same category should generally be semantically relevant. Moreover, the relationships among directories are also carefully revealed and assigned so that directories in the same hierarchy obey their inherent semantic structures. All categories are arranged in a hierarchical manner in the Yahoo! hierarchy, with 14 top-level categories in the Yahoo! directory. We use the "Arts and Humanities" category as our source of web pages. This category contains 26 sub-categories, which we denote as level-1 categories. Table 1 lists all the level-1 categories in the Art category. The '@' character appended in a category name denotes a reference link to other category. Such categories are also considered as physical categories in Art hierarchy in our experiments. Each level-1 category contains several sub-categories which are denoted as level-2 categories. Notice that the levels mentioned here should not refer to the image semantic levels as we discussed in Section 2.2. A level-1 or level-2 category contains two parts of hyperlinks as shown in Fig. 5. The first part is labeled by 'Categories' and contains hyperlinks which link to lower level categories, i.e. the level-$(n+1)$ category, where n is the current level. The second part is labeled 'Site Listings' and includes the instantiation hyperlinks which link to corresponding web pages. In this work we collect web pages which are linked by all instantiation hyperlinks in all level-1 and level-2 categories. However, we discard those pages which have no suitable images. The statistics of those selected web sites are listed in Table 2. There are total 7736 web pages in the corpus. All pages were preprocessed to extract the images and identify their ETs. The ETs used in the experiments consist of ET_{Normal} and $ET_{Caption}$, as described in Section 3.2.2. We then transformed each image

Table 1
The list of level-1 categories in 'Art' hierarchy

Art history	Education
Art Weblogs@	Employment
Artists	Events
Arts Therapy@	Humanities
Awards	Institutes
Booksellers@	Museums, galleries, and centers
Censorship	News and media
Chats and forums	Organizations
Crafts	Performing arts

Table 2
The statistics of web pages in the corpus

Number of level-1 categories	27
Number of level-1 web pages	1785
Number of level-2 categories	436
Number of level-2 web pages	5951

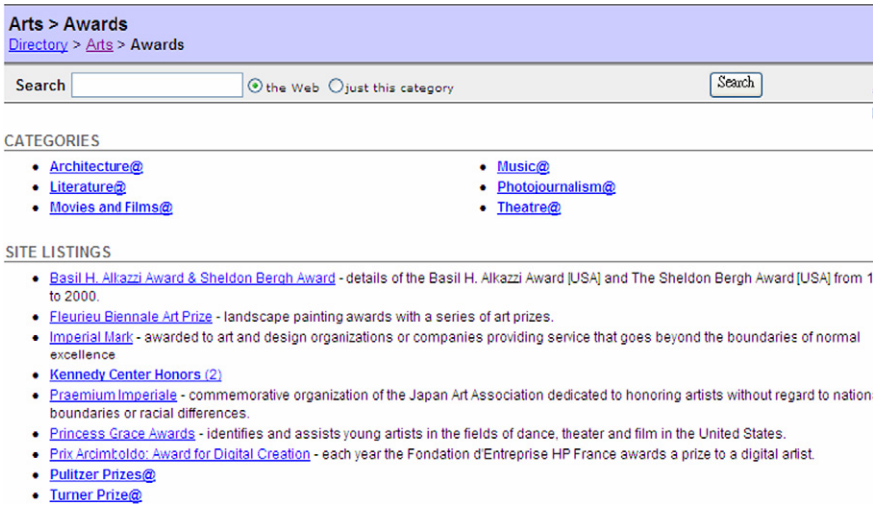


Fig. 5. The sub-categories and site listings of category 'Arts > Awards'.

to an image vector as in Section 3.2.3. We extracted 44,782 images from the 7736 pages and discarded 17,215 unqualified images, as described in Section 3.2.1. This reduced the number of images used as the training images to 27,567. The ETs and EKs of each image were extracted as described in Section 3.2. A total number of 58,729 distinct EKs had been identified.

To train the image vectors we constructed a self-organizing map which consists of 900 neurons in 30×30 grid format. The number of neurons is determined by a simple rule of thumb that the dimension of the map should be close to the average number of items associated with a neuron. According to this rule we should set the dimension of the map to about $M^{\frac{1}{3}}$, which is 30 in our experiment. The initial gain is set to 0.4 and the maximum training time is set to 500 in the training process. Both parameters are determined experimentally to obtain better results. After the training, we label each image to a neuron, as described in Section 3.3. A part of the labeling results is shown in Fig. 6, with only one of the neurons shown due to space limitations. It is clear that those images in the same cluster vary in visual features. However, their ETs all contain similar words such those associated with the same neuron in the KCM. This makes our method different from other content-based approaches. An example of the retrieval results when we use 'landscape image' as query words is shown in Fig. 7. The ranks increase from left to right, and up to down. This example shows that 'semantically similar' images may be retrieved, even when they differ in various aspects of visual features.

To evaluate the proposed method, we performed two types of tasks. The first task is to draw traditional precision versus recall curves over all queries. The second task is to use a user-defined measure for evaluation. We first describe the precision versus recall curve approach (Baeza-Yates & Ribeiro-Neto, 1999). Here the recall is the percentage of correct documents being retrieved over all correct documents corresponding to a query, i.e. $recall = \frac{|R_a|}{|R|}$ where R_a denotes the set of correct documents being retrieved and R is the set of correct documents for a query q. The precision is the percentage of correct documents in the retrieved documents in responding to q, i.e. $precision = \frac{|R_a|}{|A|}$ where A denotes the set of retrieved documents for q. To simplify this task, the images relevant to a query image are defined as those images that are in the same category as the query image, and all other images are considered as irrelevant. All images are used as queries and the average precision versus recall curve is drawn as shown in Fig. 8.

To perform the second type of evaluation, we make some assumptions about the semantics of images. Since

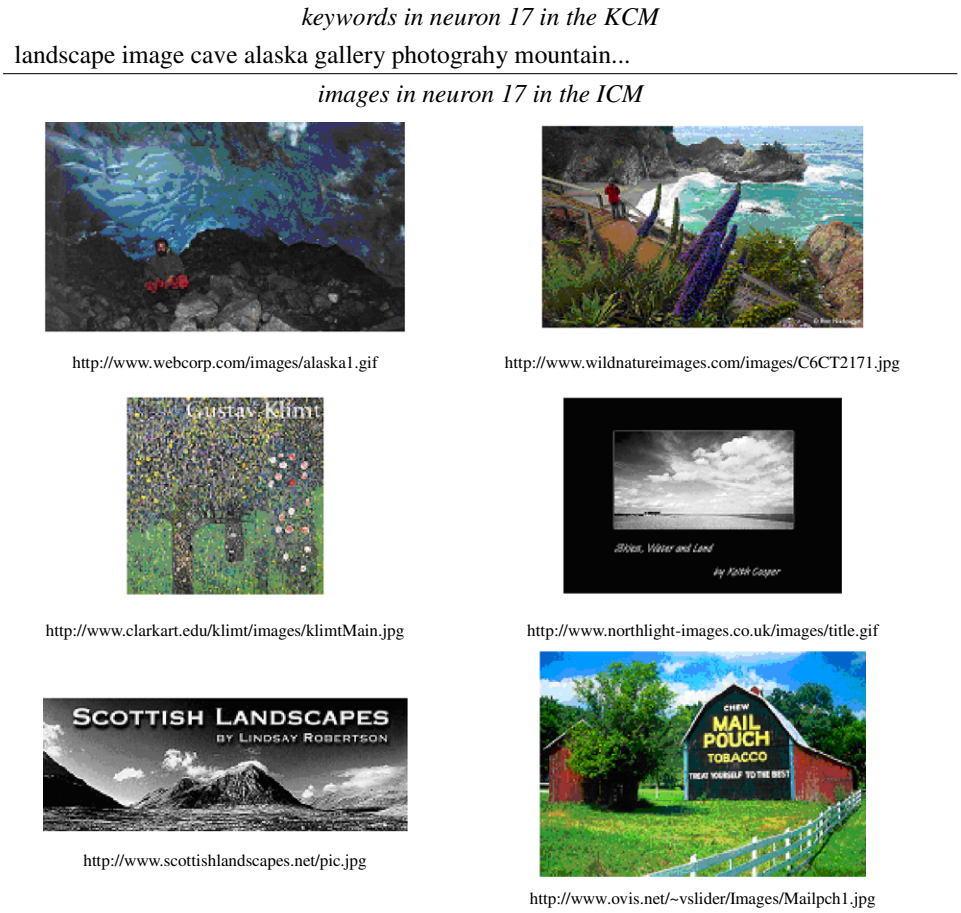


Fig. 6. The contents of neuron 17 in the ICM and KCM, respectively. We do not show the full contents due to space limitation.

Fig. 7. The retrieval result of query words 'landscape image'.

we select web pages according to the Yahoo! category hierarchy, we assume that all web pages in the same category are semantically relevant and relate to the same topic. Another assumption is that web pages in a level-2 category are also semantically relevant to those web pages in its associated level-1 category. However, although categories in adjacent levels are semantically relevant, their relevance should be less than that in the same category. For example, all web pages in category "Art Historians" should be relevant. It is the same for category "Art History". Since the "Art History" category is the parent category of "Art Historians" category in the Yahoo! hierarchy, the second assumption states that two web pages are more relevant when they are in the same category (both in Art History or in Art Historians) than in different categories (one in Art History and the other in Art Historians). We can thus define five types of relationships between two categories. The first type is the equivalent relationship such that the two categories are identical. The second type is the parent-child relationship such that one category is the parent of the other category. The third type is the sibling relationship such that the two different categories have the same parent. The fourth type is the cross-level relationship such that the two categories are in different levels and they do not have the parent-child relationship. The last type is the equi-level relationship such that the two categories are both in the level 2 but with different parents or they are both level-1 categories. With these assumptions and relationships, we may design an evaluation measure for our method, as follows:

$$E = \sum_{1 \leqslant i \leqslant M} \mathrm{CSD}(\mathbf{q}, \mathbf{x}_i) R(\mathbf{x}_i), \tag{6}$$

where CSD is a function which returns the categorical semantic distance (CSD) between its two arguments, as defined in Table 3. We define the CSD according to the relationships between categories in the Yahoo! hierarchy. The value for each type of relationship is determined according to the length of the optimal path between two categories with a relationship such as that shown in Fig. 9. For example, when two level-2 categories have the same parent, the length of the optimal path between these categories is 2. In addition, $R(\mathbf{x}_i)$ is the rank of $\mathbf{x}_i$ when $\mathbf{q}$ is presented as query.

The evaluation measure E defined in Eq. (6) has the minimal value when the ranks of retrieved images optimally reflect their semantic relationships defined in the Yahoo! hierarchy. That is, the images in the same category as the query image should be at the top of the ranking. Images that have the parent-child relationships to the query image

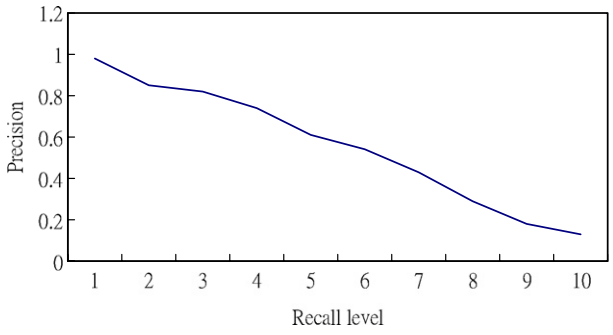


Fig. 8. The average precision versus recall curve over all images.

Table 3
The return values of the categorical semantic distance function in different relationships

Relationship	Value
Equivalent relationship	0
Parent-child relationship	1
Sibling relationship	2
Cross-level relationship	3
Equi-level relationship	4

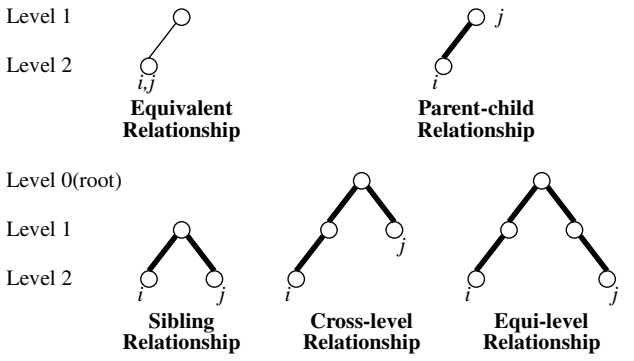


Fig. 9. We determine the categorical semantic distance between two categories by the length of the optimal path between them. Symbols *i* and *j* denote the two categories to be measured.

should follow in the ranking. After these images are those images that have the sibling relationships to the query image, and so on. With this measurement, we can effectively evaluate the results for a semantic image retrieval scheme.

To evaluate this method, we used every image in the corpus as queries, and then measured the evaluation measure for each query. Table 4 lists the statistics of all the evaluation measures. We compute the average evaluation measure over all queries and obtain the value 735.4. Since it is hard to compare this result with results from other studies, we also compute the theoretical average of E for reference. As described above, images in categories with different relationships to the query image should be ranked according to their categories for optimal evaluation measure. This ranking is called an optimal ranking and results

Table 4
The statistics of the E over all image queries

Range for E	Number of images in range
More than 10000	0
9000–9999	11
8000–8999	5
7000–7999	93
6000–6999	87
5000–5999	348
4000–4999	1365
3000–3999	1469
2000–2999	2876
1000–1999	5478
Less than 1000	15835

Table 5
The average E values for 10 degrees of distortion

Degree of distortion (%)	Average E	Number of image queries
10	874	12,792 (46.4%)
20	2377	22,353 (81.1%)
30	4821	16,869 (97.5%)
40	7692	27,503 (99.8%)
50	11,092	27,567 (100%)
60	20,312	27,567 (100%)
70	33,785	27,567 (100%)
80	46,872	27,567 (100%)
90	61,396	27,567 (100%)
100	78,743	27,567 (100%)

in minimal E. To compute the reference values of E, we emulate different ranking results and compute E for each emulation. In total, we emulate 10 different rankings. Each ranking is distorted from the optimal ranking in degrees ranging from 10% to 100%. A 10% distortion means that we randomly disarrange 10% of the images in the optimal ranking, and other degrees are similar. We compute the average E value over 100 times of distortion for each degree, and Table 5 lists the average E values for these 10 different distorted rankings. We also list the number of image queries that have E values agreeing with corresponding degrees of distortion. It is clear that our method can produce good results since a large portion of image queries can result in small E values.

The two evaluation tasks described above use some kind of mathematical measurements to provide objective judgements on the quality of the retrieval result. They do provide some clues in appraising the goodness of the proposed method. However, the assessments come from the end users give the final judgement to the retrieval performance of an image retrieval system. Thus we decided to perform a human evaluation process. This process uses another set of images which are collected randomly from the Web using 20 query words/phrases listed in Table 6. For each query result we collected the top 1000 web pages that contain at least one qualified image, making a 20000-pages corpus. These pages are also preprocessed and mined as described in Section 3. We asked some volunteers to test our system and collected their opinions. There are 97 subjects who take the information retrieval course participate in the test. The 20 queries in Table 6 are used to retrieve relevant images. The subjects are asked to select the relevant ones from the top 50 images in the ranking for each query. We then accumulate their selections to obtain the final accuracy rate of our method. In all 97000 images in the ranking for all subjects, the subjects report 81783 correct ones. An accuracy rate of 84.3% is achieved.

We also compare the proposed method with the PicSOM system (Laaksonen et al., 2000) using the three scalar measurements derived from the observed probability used in PicSOM. These measurements are used for evaluating feature extraction methods, which play a crucial role in our method. To obtain these measurements, we let images

Table 6
The 20 query words/phrases used in human evaluation process

Mountains	Fishes	Lakes	Iris flowers
Deserts	Notebook computers	Laser printers	LCD screens
Keyboards	RAM	Bill Clinton	Michelangelo Buonarroti
Claude Monet	Ludwig Van Beethoven	Tom Cruise	Planet Jupiter
Black holes	Star wars	Mars mission	Submarines

of web pages in the same level-1 category belong to the same class, and obtain $\eta_{local} = 0.17$, $\eta_{global} = 0.73$, and $\eta_{half} = 0.84$. For comparison, the average values for the three measurements reported in Laaksonen et al. (2000) are 0.067, 0.269, and 0.651, respectively. The result shows that our scheme outperforms traditional CBIR systems that use visual features.

5. Conclusions

In this work we propose a novel approach for semantic-based image retrieval, using a text mining approach to discover semantically related images. Unlike other semantic-based image retrieval approaches, we avoid direct analysis of images and rely on their environmental texts to discover their semantics. The reason for this approach is that it is hard to extract semantics directly from images, requiring human intervention to provide such semantic information. Our approach reduces this need, although our approach relies on the precise segmentation of the environmental texts. In this work we design several criteria to extract environmental texts which we believe may convey the most semantic information of the image. These criteria include the <img> tag in the HTML source, captions, nearby texts, etc. The environmental texts of every image are further clustered according to their semantic similarities through the self-organizing map algorithm, which is a part of the text mining process. The result of the text mining process contains two maps which reveal the semantic relationships among images and keywords, respectively. We then use these maps to perform image retrieval tasks.

In this work there are two types of retrieval tasks, namely retrieval by keywords and retrieval by examples. The first type uses the relationships among keywords revealed by the text mining process to retrieval relevant images of the user-specified query words. In the query-by-examples mode, the user presents an example image and the system retrieves the relevant images according to the image relationships discovered by the text mining process. The approach was tested on a set of web pages which were obtained from a part of the Yahoo! hierarchy since that provides sufficient semantic information which is useful in later evaluation process. We perform two types of evaluation process. One is the average precision versus recall curve and the other is a semantic evaluation measurement. Both evaluation schemes produced promising results.

We provide some discussions about the restrictions and further development here. It is clear that our method can only be applied for documents with both images and texts, and Web pages are ideal for this method. However, collections of skeleton images such as medical images or photo albums would not be suitable for our method unless some kind of annotations were added to these images manually or automatically. Actually, automatic annotation of images is another application of our text mining process, and in the near future will be integrated into this work to overcome the skeleton image problem.

In the extraction of ETs it is difficult to be precise, and so both the training results and the retrieval results may be deteriorated. A plausible remedy is to use a semi-automatic process that allows users to segment ETs on their own. Another approach would be to use a relevance feedback process to learn and modify the criteria in the extraction of ETs. Both approaches will be developed in our future work.

References

Al-Khatib, W., Day, F., Ghafoor, A., & Berra, P. B. (1999). Semantic modeling and knowledge representation in multimedia databases. *IEEE Transactions on Knowledge and Data Engineering, 11*(1), 64–80.

Allan, J., Carbonell, J., Doddington, G., Yamron, J., & Yang, Y. (1998). Topic detection and tracking pilot study: Final report. In *Proc. DARPA broadcast news transcription and understanding workshop, Lansdowne, VA* (pp. 194–218).

Baeza-Yates, R., & Ribeiro-Neto, B. (1999). *Modern information retrieval* (1st ed.). ACM Press.

Cavazza, M., Green, R. J., & Palmer, I. J. (1998). Multimedia semantic features and image content description. In *Proc. 1998 multimedia modeling, Lausanne, Switzerland* (pp. 39–46).

Chang, S., Smith, J., Beigi, M., & Benitez, A. (1997). Visual information retrieval from large distributed online repositories. *Communications of ACM, 40*, 63–71.

Chang, S. F., Chen, W., & Sundaram, H. (1998). Semantic visual templates: linking visual features to semantics. In *Proc. IEEE international conference on image processing (ICIP'98), Chicago, IL* (pp. 531–535).

Chen, Z., Liu, W., Zhang, F., Li, M., & Zhang, H. (2001). Web mining for web image retrieval. *Journal of the American Society of Information Science, 52*(1), 831–839.

Colombo, C., DelBimbo, A., & Pala, P. (1999). Semantics in visual information retrieval. *IEEE Multimedia, 6*(3), 38–53.

Cox, T. F., & Cox, M. A. A. (1994). *Multidimensional Scaling*. London, UK: Chapman & Hall.

Dagan, I., Feldman, R., & Hirsh, H. (1996). Keyword-based browsing and analysis of large document sets. In *Proc. 5th annual symposium on document analysis and information retrieval (SDAIR), Las Vegas, NV* (pp. 191–208).

Deerwester, S., Dumais, S., Furnas, G., & Landauer, K. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science, 40*(6), 391–407.

De Marsicoi, M., Cinque, L., & Levialdi, S. (1997). Indexing pictorial documents by their content: A survey of current techniques. *Image and Vision Computing, 15*, 119–141.

Doermann, D. (1998). The indexing and retrieval of document images: a survey. *Computer Vision and Image Understanding, 70*(3), 287–298.

Dori, D., & Hel-Or, H. (1998). Semantic content based image retrieval using object-process diagrams. In *Proc. 7th international workshop on structural and syntactic pattern recognition (SSPR98), Sydney, Australia* (pp. 15–30).

Eakins, J. P. (1996). Automatic image content retrieval: Are we going anywhere? In *Proc. 3rd international conference on electronic library and visual information research, Milton Keynes, UK* (pp. 123–135).

Feldman, R., & Dagan, I. (1995). Kdt – knowledge discovery in texts. In *Proc. 1st annual conference on knowledge discovery and data mining (KDD), Montreal, Canada* (pp. 112–117).

Feldman, R., Dagan, I., & Hirsh, H. (1998). Mining text using keyword distributions. *Journal of Intelligent Information Systems, 10*, 281–300.

Feldman, R., Klosgen, W., & Zilberstein, A. (1997). Visualization techniques to explore data mining results for document collections. In *Proc. 3rd annual conference on knowledge discovery and data mining (KDD), Newport Beach, CA* (pp. 16–23).

Frankel, C., Swain, M., & Athitsos, V. (1996). *Webseer: an image search engine for the world-wide web*. Tech. Rep. 94-14, Computer Science Department, University of Chicago, Chicago, IL.

Gudivada, V. N., & Raghavan, V. V. (1995). Content-based image retrieval system. *IEEE Computer, 28*(9), 18–22.

Gupta, A., & Jain, R. (1997). Visual information retrieval. *Communications of the ACM, 40*(5), 71–79.

Harmandas, V., Sanderson, M., & Dunlop, M. (1997). Image retrieval by hypertext links. In *Proc. 20th international ACM SIGIR conference on research and development in information retrieval* (pp. 296–303).

Hermes, T., Klauck, C., Kreyβ, J., & Zhang, J. (1995). Image retrieval for information systems. In *Proc. SPIE, Vol. 2420 storage and retrieval for image and video databases III* (pp. 394–405).

Honkela, T., Kaski, S., Lagus, K., & Kohonen, T. (1996). *Newsgroup exploration with websom method and browsing interface*. Tech. Rep. A32, Laboratory of Computer and Information Science, Helsinki University of Technology, Espoo, Finland.

Jolliffe, I. T. (1986). *Principal component analysis*. Berlin: Springer-Verlag.

Kaski, S., Honkela, T., Lagus, K., & Kohonen, T. (1998). Websom – self-organizing maps of document collections. *Neurocomputing, 21*, 101–117.

Kohonen, T. (1997). *Self-organizing maps*. Berlin: Springer-Verlag.

Kohonen, T. (1998). Self-organization of very large document collections: state of the art. In *Proc. 8th international conference on artificial neural networks, London, UK, Vol. 1* (pp. 65–74).

Laaksonen, J., Koskela, M., Laakso, S., & Oja, E. (2000). Picsom: content-based image retrieval with self-organizing maps. *Pattern Recognition Letters, 21*(13-14), 1199–1207.

Meghini, C., Sebastiani, F., & Straccia, U. (1997). The terminological image retrieval model. In *Proc. 9th international conference on image analysis and processing (ICIAP'97), Florence, Italy, Vol. 2* (pp. 156–163).

Mukherjea, S., & Cho, J. (1999). Automatically determining semantics for world wide web multimedia information retrieval. *Journal of Visual Languages and Computing, 10*, 585–606.

Ojala, T., Rautiainen, M., Matinmikko, E., & Aittola, M. (2001). Semantic image retrieval with hsv correlograms. In *Proc. 12th Scandinavian conference on image analysis, Bergen, Norway* (pp. 621–627).

Ritter, H., & Kohonen, T. (1989). Self-organizing semantic maps. *Biological Cybernetics, 61*, 241–254.

Rowe, N., & Frew, B. (1998). Automatic caption localization for photographs on world-wide web pages. *Information Processing and Management, 34*, 95–107.

Sclaroff, S., Cascia, M. L., Sethi, S., & Taycher, L. (1999). Unifying textual and visual cues for content-based image retrieval on the world wide web. *Computer Vision and Image Understanding, 75*(1–2), 86–98.

Srihari, R. K., & Burhans, D. T. (1994). Visual semantics: extracting visual information from text. In *Proc. 12th national conference on artificial intelligence, Seattle, WA* (pp. 793–798).

Wu, Q., Sitharama Iyengar, S., & Zhu, M. (2001). Web image retrieval using self-organizing feature map. *Journal of the American Society for Information Science and Technology, 52*(1), 868–875.

Zhao, R., & Grosky, W. I. (2002). Narrowing the semantic gap: Improved text-based web document retrieval using visual features. *IEEE Transactions on Multimedia, 4*(2), 189–200.

Zhuang, Y., Mehrotra, S., & Huang, T. S. (1999). A multimedia information retrieval model based on semantic and visual content. In *Proc. 5th international conference for young computer scientists (ICYCS), Nanjing, China*.