Narrowing the Semantic Gap—Improved Text-Based Web Document Retrieval Using Visual Features

Rong Zhao, *Member, IEEE,* and William I. Grosky

***Abstract*—In this paper, we present the results of our work that seek to negotiate the gap between low-level features and high-level concepts in the domain of web document retrieval. This work concerns a technique, latent semantic indexing (LSI), which has been used for textual information retrieval for many years. In this environment, LSI determines clusters of co-occurring keywords—sometimes called concepts—so that a query which uses a particular keyword can then retrieve documents perhaps not containing this keyword, but containing other keywords from the same cluster. In this paper, we examine the use of this technique for content-based web document retrieval, using both keywords and image features to represent the documents. Two different approaches to image feature representation, namely, color histograms and color anglograms, are adopted and evaluated. Experimental results show that LSI, together with both textual and visual features, is able to extract the underlying semantic structure of web documents, thus helping to improve the retrieval performance significantly, even when querying is done using only keywords.**

***Index Terms*—Anglogram, color, content-based retrieval, feature, histogram, image, latent semantic indexing (LSI), multimedia, semantics, World Wide Web.**

I. Introduction

INFORMATION is increasingly becoming ubiquitous and all pervasive, with the world wide web as its primary repository. With the popularity of multimedia technology, contents of the world wide web have been a lot more versatile than a few years ago. However, although more information is available on the web, the efficient and effective retrieval and management of these web documents are still very challenging research issues. When navigating in such a vast collection of linked multimedia documents, users can easily get lost in its depths. Some users know what they are looking for and try to satisfy their needs by following appropriate links. These users may or may not find something of interest, but may easily miss other, more relevant documents far from their current browsing paths. Other users who know what they want can express their needs to a software mediator called a *search engine*, which, ostensibly, helps them find the appropriate documents. Still other users may not be able to articulate exactly what they want, but will know that a document satisfies their needs when they see it; they then would like to examine other similar documents.

Manuscript received April 27, 2001; revised February 26, 2002. The associate editor coordinating the review of this paper and approving it for publication was Prof. Alberto Del Bimbo.

R. Zhao is with the Department of Computer Science, State University of New York, Stony Brook, NY 11794-4400 USA (email: rzhao@cs.sunysb.edu).

W. I. Grosky is with the Multimedia Information Systems Laboratory, the Department of Computer and Information Science, University of Michigan, Dearborn, MI 48128-1491 USA (email: wgrosky@umich.edu).

Publisher Item Identifier S 1520-9210(02)04859-9.

Search engines are still in their infancy. Although the exact nature of many of the algorithms they use for finding appropriate pages is proprietary [26], there is some research that suggests that the algorithms they use are not very accurate [15]. Existing search engines and their users are typically at cross purposes. While these systems normally retrieve documents based on low-level features, users usually have a more abstract notion of what will satisfy them when conducting a query for certain information. For instance, most search engines use low-level syntactic properties to characterize the semantics of web documents. These properties usually reduce to various sophisticated variations of simple keyword counts. There are research prototypes that take link information into account [30], but they still do not overcome the so-called *semantic gap* [14]. This is a term that has been applied to content-based image retrieval, but which certainly has relevance to web searching. It corresponds to the mismatch between users' requests and the way automated search engines try to satisfy these requests. Sometimes, the user has in mind a concept so abstract that he himself does not know what he wants exactly until he sees it. At that point, he may want documents similar to what he has just seen or can envision. Again, however, the notion of similarity is typically based on high-level abstractions, such as activities or events described in the document, or some evoked emotions, among others. Standard definitions of similarity using low-level features generally will not produce quality results.

In reality, the correspondence between user-based semantic concepts and system-based low-level features is always many-to-many. A certain word can be interpreted in different ways within different contexts; while the same concept is usually associated with a set of different terms, and people may have different preferences about which one to use. Generally speaking, these problems exist no matter what features are used in the system. Making the scenario even more complicated, it is very likely that a web document does not present a fixed semantics, but multiple semantics that vary over time. It is not only that different users may have different opinions about the similarity between web documents, but also, the same user may have different ideas under different circumstances. Even though user-based similarity is a very subjective issue based on high-level concepts, so far all existing management systems or search engines can only rely on some statistical measures based on low-level features.

In this paper, we attempt to find a solution to negotiating this semantic gap in web document retrieval. We will present the results of our study that seeks to transform low-level features to a higher level of meaning. This work concerns a technique called *latent semantic indexing* (*LSI*) [10], which has been used

for textual information retrieval for many years. In this environment, LSI is used to determine clusters of co-occurring keywords, sometimes, called *concepts*, so that a query which uses a particular keyword can then retrieve documents perhaps not containing this keyword, but containing other keywords from the same concept cluster. We will examine the use of this technique for content-based web document retrieval, with both keywords and image features to represent document contents. Two different approaches to image feature representation, namely, *color histograms* and *color anglograms*, are adopted and evaluated. Experimental results show that LSI, together with both textual and visual features, is able to extract the underlying semantic structure of web documents, thus helping to improve the retrieval performance significantly, even when querying is done using only keywords.

The remainder of this paper is organized as follows. In Section II, related works on capturing semantics and image feature extraction and representation are briefly discussed. Section III introduces the theoretical background of LSI, and also outlines its usage in both textual information retrieval and image retrieval. Section IV describes how to use LSI to conduct semantic-based retrieval of web documents. In Section V, we present an improved approach that integrates LSI with both keywords and image features in the documents. Conclusions are drawn in Section VI, along with some proposed future work.

II. Related Works

The motivation of this work is our belief that LSI is able to extract the underlying semantic structure of web documents, and that this semantic structure can be represented by integrating textual and image features of the web documents. This study addresses the following questions: *How can the semantics of a web document be derived given a set of features? How can image features be used to characterize the semantics of a multimedia web document? When can we determine that two web documents have similar or overlapping semantics? How can semantic-based web document retrieval help us in navigating the web more efficiently?* We will briefly review existing research in these fields in this section and then present and validate our approach with experimental results in Section IV and V.

A. Capturing Semantics

As previously mentioned, LSI has often been applied to full-text document collections [4], [10]. However, there has not been much work on using this technique for image collections [1], [6], [18], [20], [25]. The only work of which we are familiar that intentionally uses such a dimensional reduction technique [20] attempts to find a better way to search images on the web. In this work, LSI is applied to analyzing text that appears close to a given image. An image feature vector is then comprised of two components, one component representing visual features and the other representing the textual information transformed by using LSI. Since the LSI is just used on text, this approach is not able to find different image features that co-occur with the same set of textual keywords.

In our previous study of applying LSI to content-based image retrieval [40]–[42], we experimented with different visual features, such as global color histograms, subimage color histograms, and color anglograms, and the results showed that LSI is effective in finding the semantics of images, thus helping to improve the retrieval performance. We believe that LSI also discovers that certain sets of different image features co-occur with the same set of keywords, resulting in the formation of general concept clusters comprising various textual and image features. This idea was validated by the experiments of integrating visual features with textual annotations that we conducted in [42].

The promising results of using LSI in textual and image retrieval inspired us to negotiate the semantic gap in web document retrieval. We intend to use both textual keywords and image features in an attempt to discover the latent semantic structure of web documents and to correlate keywords with image features.

There have been various papers concerned with transforming web pages into concepts [7], [16], [39]. These papers show how to transform the set of pages returned by a standard search engine into a more browsable representation through the mediation of clustering, each cluster corresponding to one of the concepts.

An important aspect of our study is to bring multimedia information into the definition of web document semantics. We characterize content-based retrieval systems that try to capture user semantics into two classes, namely, *system-based* and *user-based*. System-based approaches either try to define various semantics globally, based on formal theories or consensus among domain experts, or to use other techniques, not based on user-interaction, to get from low-level features to high-level semantics. User-based approaches, on the other hand, are adaptive to user behavior and try to construct individual profiles. An important component of most user-based approaches is the technique of relevance feedback [3], [28], which has not been generally used on the web yet, especially for images.

Some of the examples of system-based approaches can be found in [9], [20], [27], and [32]. Reference [27] is the first paper that concerns retrieving images, in this case, graphic objects, based on user semantics. A methodology for composing features which evoke certain emotions is discussed in [9], whereas [20] uses textual information close to an image on a web page to derive information regarding the image contents. [32] explores a heterogeneous clustering methodology that overcomes the drawback of single-feature matching when dealing with images that are considered similar by computation but actually having different semantics.

Approaches that depend on some form of user interaction include [8], [21], and [29]. Mediated by user interaction, the system discussed in [8] defines a set of queries that correspond to a user concept. Reference [21] is a system that learns how to combine various features in the overall retrieval process through user feedback. Reference [29] introduces an exploration paradigm based on an advanced user interface simulating three-dimensional (3-D) space. In this space, thumbnail images having the same user semantics are displayed close to each other, and thumbnails that are far from the user's semantic view are smaller in size than thumbnails that are closer to the user's semantic

view. Users can also convert images that are close to each other into a concept and replace the given set of thumbnails by a concept icon.

B. Image Feature Extraction and Representation

During the past few years, many different image representations have been developed, and various content-based image retrieval systems have been proposed. The most widely used image features are color, texture, shape, and spatial layout. Since none of these low-level features is powerful enough by itself to represent the image contents on the object level, researchers have been focusing on combining different features, or different feature representations, and developing integrated similarity measures.

In the query by image content (QBIC) system [23], image objects are indexed with color histograms, color moments, and shape descriptors. Other research groups have also tried to combine color and shape features for improving the performance of image retrieval. In [19], the color in an image is represented by color histograms in (R, G, B) space, while a histogram of the directions of the edge points is used to represent the general shape information. A composite feature descriptor is proposed in [22] based on a clustering technique, combining the information of both shape and color clusters. In [2], a system that uses a so-called *blobworld* representation to retrieve images is proposed. This approach attempts to recognize the nature of images as combinations of these blobs.

Due to the uncontrolled nature of images, how to extract image objects automatically and precisely is still beyond the reach of state-of-the-art computer vision. Moreover, each image object may appear differently, depending on viewpoint, occlusion, and deformation.

Though it is more meaningful to represent the spatial distribution of color information based on image objects or regions, various fixed image-partitioning techniques have also been proposed because of their simplicity and acceptable performance. In [33], an image is divided into five partially overlapped, fuzzy regions, with each region indexed by three moments of the color distribution. The Color-WISE system [32] partitions an image into 64 blocks with each block indexed by its dominant hue and saturation values.

Instead of partitioning an image into regions, there are other approaches for the representation of spatial color distribution. A histogram refinement technique is described in [24] by partitioning histogram bins based on the spatial coherence of pixels. A pixel is coherent if it is a part of some sizable similarly-colored region, and incoherent otherwise. In [17], a statistical method is proposed to index an image by *color correlograms*, which are tables containing color pairs, where the kth entry for $\langle i, j \rangle$ specifies the probability of locating a pixel of color j at a distance k from a pixel of color i in the image.

We note that neither the histogram refinement nor the color correlogram can recognize the nature of images on object level. As for meaningful region-based image representations, two image objects are usually considered similar only if the corresponding regions they occupy overlap. Along with the position dependence of similar image objects, the fixed image partition strategy does not allow image objects to be rotated within an image.

Many image features can be represented as a set of points. These points can be tagged with labels to capture any necessary semantics. Each of the individual points representing some feature of an image object is called a *feature point*. The entire image object is represented by a set of labeled feature points. We note that capturing the various spatial relationships among these points is an important aspect of our work, which sets our work apart from other approaches.

Effective semantic representation and retrieval requires labeling the feature points of each image object. The introduction of such feature points and associated labels effectively converts an image object into an equivalent symbolic representation, called a *point feature map*. By representing an image object as a point feature map, we capture not only image features but also the spatial relationships of these features.

The representation we use for a point feature map is called an *anglogram* [36], [42]. This is constructed by computing a Delaunay triangulation for each set of feature points in the point feature map labeled with a similar feature, and then computing a feature point histogram by counting the two largest angles produced by the triangulation. We have shown the efficacy of using this technique to represent both shape [35] and color [36], [37], and have validated the anglogram approach by applying it in a medical imaging environment [38]. We have also compared its performance with that of some other existing approaches such as those introduced in [17] and [31]. Another major advantage of this method is that it is invariant to rotation, translation, and scaling.

III. Latent Semantic Indexing in Content-Based Retrieval

LSI was introduced to overcome a fundamental problem that plagues existing textual retrieval techniques. The problem is that users want to retrieve documents on the basis of conceptual content, while individual keywords provide unreliable evidence about the conceptual meaning of a document. There are usually many ways to express a given concept. Therefore, the literal terms used in a user query may not match those of a relevant document. In addition, most words have multiple meanings and are used in different contexts. Hence, the terms in a user query may literally match the terms in documents that are not of any interest to the user at all.

In information retrieval these two problems are addressed as *synonymy* and *polyzemy*. The concept *synonymy* is used to describe the fact that there are many ways to refer to the same object. Users in different contexts, or with different needs, knowledge, or linguistic habits will describe the same concept using different terms. The prevalence of synonyms tends to decrease the *recall* performance of the retrieval. By *polyzemy* we refer to the fact that most words have more than one distinct meaning. In different contexts or when used by different people, the same term takes on a varying referential significance. Thus, the use of a term in a query may not necessarily mean that a document containing the same term is relevant at all. Polyzemy is one factor underlying poor *precision* performance of the retrieval [10].

LSI tries to overcome the deficiencies of term-matching retrieval. It is assumed that there exists some underlying latent semantic structure in the data that is partially obscured by the randomness of word choice. Statistical techniques are used to estimate this latent semantic structure, and to get rid of the obscuring noise.

The LSI technique makes use of the *singular value decomposition (SVD)*. We take a large matrix of term-document association and construct a semantic space wherein terms and documents that are closely associated are placed near to each other. The SVD allows the arrangement of the space to reflect the major associative patterns in the data, and ignore the smaller, less important influences. As a result, terms that did not actually appear in a document may still end up close to the document, if that is consistent with the major patterns of association in the data. Position in the transformed space then serves as a new kind of semantic indexing. Retrieval proceeds by using the terms in a query to identify a point in the semantic space, and documents in its neighborhood are returned as relevant results to the query.

LSI is based on the fact that the term-document association can be formulated by using the vector space model, in which each document is represented as a vector, where each vector component reflects the importance of a particular term in representing the semantics of that document. The vectors for all the documents in a database are stored as the columns of a single matrix. LSI is a variant of the vector space model in which a low-rank approximation to the vector space representation of the database is employed. That is, we replace the original matrix by another matrix that is as close as possible to the original matrix but whose column space is only a subspace of the column space of the original matrix. Reducing the rank of the matrix is a means of removing extraneous information or noise from the database it represents. According to [4], LSI has achieved average or above average performance in several experiments with the TREC collections.

A. The Vector Space Model

In the vector space model, a vector is used to represent each item or *document* in a collection. Each component of the vector reflects a particular keyword associated with the given document. The value assigned to that component reflects the importance of the term in representing the semantics of the document.

A database containing a total of d documents described by t terms is represented as a $t \times d$ *term-by-document matrix* A. The d vectors representing the d documents form the columns of the matrix. Thus, the matrix element a_{ij} is the weighted frequency at which term i occurs in document j. The columns of A are called the *document vectors*, and the rows of A are the *term vectors*. The semantic content of the database is contained in the column space of A, meaning that the document vectors span that content. We can exploit geometric relationships between document vectors to model similarity and differences in content. Meanwhile, we can also compare term vectors geometrically in order to identify similarity and differences in term usage.

A variety of schemes are available for weighting the matrix elements. The element a_{ij} of the term-by-document matrix A is often assigned such values as $a_{ij} = l_{ij}g_i$. The factor g_i is called the *global weight*, reflecting the overall value of term i as an indexing term for the entire collection. Global weighting schemes range from simple normalization to advanced statistics-based approaches [11]. The factor l_{ij} is a local weight that reflects the importance of term i within document j itself. Local weights range in complexity from simple binary values to functions involving logarithms of term frequencies. The latter functions have a smoothing effect in that high-frequency terms having limited discriminatory value are assigned low weights.

B. Singular Value Decomposition (SVD)

The SVD is a dimension reduction technique which gives us reduced-rank approximations to both the column space and row space of the vector space model. The SVD also allows us to find a rank-k approximation to a matrix A with minimal change to that matrix for a given value of k[4]. The decomposition is defined as follows:

$$A = U\Sigma V^T$$

where

- U $t \times t$ orthogonal matrix having the left singular vectors of A as its columns;
- V $d \times d$ orthogonal matrix having the right singular vectors of A as its columns;
- Σ $t \times d$ diagonal matrix having the singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r$ of the matrix A in order along its diagonal, where $r \leq \min(t, d)$.

This decomposition exists for any given matrix A [13].

The rank r_A of the matrix A is equal to the number of nonzero singular values. It follows directly from the orthogonal invariance of the *Frobenius* norm that $||A||_F$ is defined in terms of those values

$$\|A\|_F = \left\|U\Sigma V^T\right\|_F = \left\|\Sigma V^T\right\|_F = \|\Sigma\|_F = \sqrt{\sum_{j=1}^{r_A} \sigma_j^2}. \quad (1)$$

The first r_A columns of matrix U are a basis for the column space of matrix A, while the first r_A rows of matrix V^T are a basis for the row space of matrix A. To create a rank-k approximation A_k to the matrix A, where $k \leq r_A$, we can set all but the k largest singular values of A to be zero. A classic theorem about the SVD by Eckart and Young [12] states that the distance between the original matrix A and its rank-k approximation is minimized by the approximation A_k. The theorem further shows how the norm of that distance is related to singular values of matrix A. It is described as

$$\|A - A_k\|_F = \min_{rank(X) \leq k} \|A - X\|_F = \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_{r_A}^2}. \quad (2)$$

Here $A_k = U_k \Sigma_k V_k^T$, where U_k is the $t \times k$ matrix the columns of which are the first k columns of matrix U, V_k is the $d \times k$ matrix the columns of which are the first k columns of matrix V, and Σ_k is the $k \times k$ diagonal matrix the diagonal elements of which are the k largest singular values of matrix A. Using the SVD to find the approximation A_k guarantees that the approximation is the best that can be achieved for any given choice of k.

C. Similarity Measure

In the vector space model, a user queries the database to find relevant documents, using the vector space representation of those documents. The query is also a set of terms, with or without weights, represented by using a vector just like the documents. The matching process is to find the documents most similar to the query in the use and weighting of terms. In the vector space model, the documents selected are those geometrically closest to the query in the transformed semantic space.

One common measure of similarity is the cosine of the angle between the query and document vectors. If the term-by-document matrix A has columns $a_j, j = 1, 2, \ldots, d$, those d cosines are computed according to the following formula:

$$\cos\theta_j = \frac{a_j^T q}{\|a_j\|_2 \|q\|_2} = \frac{\sum_{i=1}^{t} a_{ij} q_i}{\sqrt{\sum_{i=1}^{t} a_{ij}^2} \sqrt{\sum_{i=1}^{t} q_i^2}} \tag{3}$$

for $j = 1, 2, \ldots, d$, where the Euclidean vector norm $\|x\|_2$ is defined by

$$\|x\|_2 = \sqrt{x^T x} = \sqrt{\sum_{i=1}^{t} x_i^2} \tag{4}$$

for any t-dimensional vector x.

The LSI technique has been successfully applied to textual information retrieval, in which it shows distinctive power of finding the latent correlation between terms and documents [4], [5], and [10]. This inspired us to apply LSI to content-based image retrieval. In a previous study [40], [42], we made use of the power of LSI to reveal the underlying semantic nature of image contents, and thus to find the correlation between image features and the semantics of the image or its objects. We explored this approach further by correlating low-level feature groups and high-level semantic clusters. Experimental results show that integrating LSI with content-based techniques helps improve the retrieval performance significantly.

Now we attempt to extend the power of LSI to the domain of web document retrieval. Conventional web document retrieval techniques are based on low-level features such as keywords in the contents or links of the documents because it is fairly easy to extract these features automatically and analyze them statistically. However, they are not the real means by which human users understand and retrieve web documents. Higher level concepts or topics, with certain domain knowledge, are the major criteria to conduct any kind of search. How to automate the extraction and analysis of this data is still beyond the state-of-the-art in information retrieval or artificial intelligence, which leaves those low-level features the only means available with which to initiate a search. To find a remedy for this problem, we attempt to use LSI to bridge this semantic gap in web retrieval. Some preliminary experiments have been conducted and the results are very promising. The next section details the effectiveness of using LSI with textual features. In Section V, we introduce a more effective approach in which LSI is integrated with both textual features and visual elements.

IV. Semantic-Based Web Document Retrieval Using Textual Contents

This section presents our approach to discover the co-occurrence between keywords in web documents. Considering the problems of *synonymy* and *polyzemy* discussed in the previous section, many different keywords could be used to refer to the same concept, while most keywords have more than one distinct meaning and are used in different contexts to represent different semantics. Our motivation is to discover the correlation between keywords and thus construct more meaningful concept clusters. Retrieval will then be conducted based on these high-level concept clusters, rather than those low-level keywords.

To validate our semantic-based retrieval technique, we choose news service sites for our experiments. We are interested in this application domain because of the following reasons. First of all, news headlines are often used as their URL anchors and document titles in most news service sites. The topic or major concepts in a piece of news can usually be represented easily and clearly by a group of keywords in the headline. Second, news service sites such as *cnn.com*, *abcnews.com*, and *msnbc.com* often have extensive coverage of the same news topic during a certain period of time. Therefore, documents on these sites can be used to cross examine the semantic structures of similar documents and corresponding concept clusters discovered by using our retrieval method. Finally, news documents often include some multimedia components, such as images, audio clips, or video clips, which are closely related to the topic of the news story. In the next section, making use of this special characteristic, we are going to present how to discover the correlation between keywords and visual features and to use this correlation to improve retrieval performance of multimedia web documents.

For our experiment, we use a collection of 24 documents on *cnn.com*. The first eight documents are about a California high school shooting incident, the next eight about President Bush's tax cut plan, and the last eight are miscellaneous documents about other topics. The headlines of these news stories, i.e., titles of the corresponding web documents, are listed in Fig. 1. Keywords, in our experiment, are the most meaningful words found within those titles. It is usually a noun referring to some roles of an affair, or some objects in an event. For instance, in the news story of "Bush urges Congress to restructure Medicare," there are three keywords, namely, *Bush*, *Congress*, and *Medicare*.

It is worth noticing that the meaning of a keyword is very much determined by its context. A meaningful keyword in one context may be of no use to the same user when it appears in some other contexts. For example, a user may be interested in looking for all the news stories related to President Bush's tax cut plan. In this context, the word "Bush" will guide the user to potentially relevant documents. However, the same keyword may be of much less importance to him if it appears in news of other issues involving the President, for example, in "Bush urges Congress to restructure Medicare." To validate that LSI is capable of distinguishing the different semantics of keywords in different contexts, we selected intentionally 8 miscellaneous documents which contain keywords that also appear in the first 16 documents, such as *school*, *California*, *Bush*, *Congress*, etc.

1. School shooting arraignment postponed
2. Pennsylvania girl arrested in classmate's shooting
3. Judge delays arraignment in school shooting
4. Investigators: Teen 'reloaded 4 times'
5. Detroit high school shooting injures 3
6. Teenager suspect faces charges as adult under new law
7. California students arrested after 'hit list,' rifle found
8. Friends of suspect won't be coming back to school Wednesday
9. House set to vote on tax rate reductions
10. Bush to give tax-cut pep talk on eve of vote
11. Bush prepares to campaign once more for tax cuts
12. House plunges ahead with tax-relief bill
13. Bush takes his tax-cut idea to the heartland
14. Bush to plead 'fiscal sanity' before Congress
15. Bush prepares budget speech for Congress
16. Bush, Democrats outline budget differences
17. Bush moves to increase federal spending on education
18. Study: Kids rate bullying and teasing as 'big problem'
19. Coach fired for wielding meat cleaver at school
20. California unveils rail system improvement plan
21. Bush officials weighing help for steel industry
22. Bush talks tough on North Korea
23. Bush urges Congress to restructure Medicare
24. Review: Web help at tax time

Fig. 1. Document titles in the experiment of semantic-based retrieval using keywords. (Source: *cnn.com*.)

In our experiment, first the keywords are extracted from each document title, and each document is then represented by a feature vector whose components are the keywords. Next, a keyword-document matrix, $\mathbf{A} = [\mathbf{V_1}, \ldots, \mathbf{V_{24}}]$, which is 55×24, is constructed using these feature vectors. Each row corresponds to one of the keywords and each column is the entire feature vector of the corresponding document.

An SVD is performed on the keyword-document matrix. The result comprises three matrices, namely, $\mathbf{U}$, Σ, and $\mathbf{V}$, where $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V^T}$. The dimensions of $\mathbf{U}$, Σ, and $\mathbf{V}$ are 55×55, 55×24, and 24×24, respectively. The rank of matrix Σ, and thus the rank of matrix $\mathbf{A}$, is 24. Therefore, the first 24 columns of $\mathbf{U}$ spans the column space of $\mathbf{A}$ and all the 24 rows in $\mathbf{V^T}$ spans the row space of $\mathbf{A}$. Σ is a diagonal matrix of which the diagonal elements are the singular values of $\mathbf{A}$. To reduce the dimensionality of the transformed space, we use a rank-k approximation, $\mathbf{A_k}$, of the matrix $\mathbf{A}$, where $k = 16$ (this optimal value of k was pragmatically determined). This is defined by $\mathbf{A_k} = \mathbf{U_k}\Sigma_\mathbf{k}\mathbf{V_k^T}$. The dimension of $\mathbf{A}_k$ is the same as $\mathbf{A}$, 55 by 24. The dimensions of $\mathbf{U}_k$, Σ_k, and $\mathbf{V}_k$ are 55×16, 16×16, and 24×16, respectively.

The query process is to compute the distance between the transformed feature vector of the query document, $\mathbf{q}$, and that of each of the 24 candidate documents in the database $\mathbf{d}$. The distance is defined as $dist(\mathbf{q}, \mathbf{d}) = \mathbf{q^T d}/||\mathbf{q}||||\mathbf{d}||$, where $||\mathbf{q}||$ and $||\mathbf{d}||$ are the norms of those vectors.

The measures of *recall* and *precision* are used in evaluating the performance of this semantic-based retrieval technique. Consider an information request I and its set R of relevant documents. Let $|R|$ be the number of documents in this set. Assume that a given retrieval method generates a document answer set A and let $|A|$ be the number of documents in this set. Also, let $|R_a|$ be the number of documents in the intersection of the sets R and A. Then *recall* is defined as

$$\text{Recall} = \frac{|R_a|}{|R|}$$

which is the fraction of the relevant documents that has been retrieved, and *precision* is defined as

$$\text{Precision} = \frac{|R_a|}{|A|}$$

which is the fraction of the retrieved documents that are considered as relevant.

With different sizes of the document answer set, $|A|$, we evaluated our method by using each of the first 16 documents as query documents. In Table I, we compare the recall and precision of using LSI with those without LSI, i.e., using straight keyword matching. Results show that LSI does improve the retrieval performance significantly.

We notice that recall and precision have the following inherent weaknesses. First, the proper estimation of recall requires detailed knowledge of all the documents in the collection. When the collection is considerably large, it will be very difficult or even impossible to have a proper estimate at all. Second, recall and precision are related measures capturing different aspects of the set of retrieved documents. In many situations, improvement of one leads to the deterioration of the other. Therefore, we introduce our own measure in the next section when evaluating the performance of semantic-based retrieval using both keywords and image features.

V. Semantic-Based Web Document Retrieval Using Textual and Visual Contents

In this section, we attempt to uncover the semantic correlation between keywords in the document title and image features in the same web document, hoping to use this correlation to improve the retrieval of multimedia web documents. As mentioned in the previous section, we select documents from news service sites for conducting out experiments. News documents often include multimedia components that are closely related to the topic or major concepts of the news story. In particular, we find that many documents on *cnn.com* have some images around or near the headline of the news story. For our experiment, we consider one image in each document, which is selected with regard to the position, size, and format of the image.

We use a collection of 20 documents on *cnn.com*. This collection consists of four semantic categories of five documents each. The categories are the Bush inaugural, the Kursk submarine accident, the Clinton impeachment, and the space-station MIR. Document titles are listed in Fig. 2, and their images are shown in Fig. 3.

In our experiment, 43 keywords are extracted from the title of these documents and a textual feature vector $\mathbf{K} = [k_1, k_2, k_3, \ldots k_{43}]^T$ is then constructed. To extract and represent the images, we apply two different approaches. Our first approach is to use global color histograms. Each image is first

TABLE I
RESULTS OF EXPERIMENT OF SEMANTIC-BASED RETRIEVAL USING KEYWORDS. (A) COMPARISON OF RECALL. (B) COMPARISON OF PRECISION

Recall	Straight Keyword Matching	Latent Semantic Indexing
$\|A\| = 10$	0.51	0.60
$\|A\| = 15$	0.63	0.75
$\|A\| = 20$	0.76	0.88

(a)

Precision	Straight Keyword Matching	Latent Semantic Indexing
$\|A\| = 10$	0.41	0.48
$\|A\| = 15$	0.34	0.40
$\|A\| = 20$	0.30	0.35

(b)

1. Bush, in first address as president
2. Education, tax cuts top Bush's Washington agenda
3. Campaign promises could prove troublesome for Bush
4. Bush's to-do list: Set tone for next four years
5. George W. Bush: The 43rd President
6. Rescue mission for crippled Russian sub enters second day
7. Russian official says chances not good for rescue of trapped crew aboard sunken nuclear sub
8. Kursk salvage raises questions
9. Russia to start recovering Kursk bodies
10. Russian navy begins attempt to evacuate sailors from sunken sub
11. Clinton acquitted; president apologizes again
12. Clinton apologizes to nation
13. Clinton's evolving apology for the Lewinsky affair
14. Clinton will not address impeachment in State of the Union
15. Clinton says 'presidents are people, too'
16. MIR prepares for risky plunge
17. Mir positioned for fiery descent
18. A Mir risk
19. Mir demise causes international high anxiety
20. New Zealand issues Mir warning

Fig. 2. Document titles in the experiment of semantic-based retrieval using both keywords and image features. (Source: *cnn.com*.)

converted from the *RGB* color space to the *HSV* color space. For each pixel of the resulting image, hue and saturation are extracted and each quantized into a 10-bin histogram. Then, the two histograms h and s are combined into one $h \times s$ histogram with 100 bins, which is the representing image feature vector of each document to which the image belongs. This is a vector of 100 elements, $\mathbf{F} = [f_1, f_2, f_3, \ldots f_{100}]^T$.

Then, we combine the textual feature vector $\mathbf{K}$ and image feature vector $\mathbf{F}$ into a new feature vector $\mathbf{V}$, so that each document is represented by such a feature vector. Next, a feature-document matrix, $\mathbf{A} = [\mathbf{V_1}, \ldots, \mathbf{V_{20}}]$, which is 143×20, is constructed using these feature vectors. Each row corresponds to one of the feature elements and each column is the entire feature vector of the corresponding document.

An SVD is performed on the feature-document matrix. The result comprises three matrices, namely, $\mathbf{U}$, Σ, and $\mathbf{V}$, where $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V^T}$. The dimensions of $\mathbf{U}$, Σ, and $\mathbf{V}$ are 143×143, 143×20, and 20×20, respectively. To reduce the dimensionality of the transformed space, we use a rank-k approximation, $\mathbf{A_k}$, of the matrix $\mathbf{A}$, where $k = 12$. This is defined by $\mathbf{A_k} = \mathbf{U_k}\Sigma_\mathbf{k}\mathbf{V_k^T}$. The dimension of $\mathbf{A_k}$ is the same as $\mathbf{A}$, 143×20. The dimensions of $\mathbf{U_k}$, Σ_k, and $\mathbf{V_k}$ are 143×12, 12×12, and 20×12, respectively.

Defining a keyword-based query from each document, in turn, we find the average sum of the positions of all of the five correct answers. Note that in the best case, where the five correct matches occupy the first five positions, this average sum would be 15, whereas in the worst case, where the five correct matches occupy the last five positions, this average sum would be 90. A measure that we use of how good a particular method is defined as

$$measure - of - goodness = \frac{18 - \frac{average - sum}{5}}{15}. \quad (5)$$

Fig. 3. Images associated with the web documents in Fig. 2. (Source: *cnn.com*.) (a) Bush inaugural. (b) Kursk submarine accident. (c) Clinton impeachment. (d) The MIR.

We note that in the best case, this measure is equal to one, whereas in the worst case, it is equal to zero [42].

The following *normalization* process will assign equal emphasis to each component of the feature vector. Different components within the vector may be of totally different physical quantities. Therefore, their magnitudes may vary drastically and thus bias the similarity measurement significantly. One component may overshadow the others just because its magnitude is relatively too large. For the feature-document matrix $\mathbf{A} = [\mathbf{V_1}, \mathbf{V_2} \ldots, \mathbf{V_{20}}]$, we have $\mathbf{A_{i,,j}}$ which is the ith component in vector $\mathbf{V_j}$. Assuming a Gaussian distribution, we can obtain the mean μ_i and standard deviation σ_i for the ith component of the feature vector across the whole collection of documents. Then we normalize the original feature document matrix into the range of [−1,1] as follows:

$$Ai, j = \frac{Ai, j - \mu_i}{\sigma_i}. \tag{6}$$

It can easily be shown that the probability of an entry falling into the range of [−1, 1] is 68%. In practice, we map all the entries into the range of [−1, 1] by forcing the out-of-range values to be either −1 or 1. We then shift the entries into the range of [0,1] by using the following formula:

$$Ai, j = \frac{Ai, j + 1}{2}. \tag{7}$$

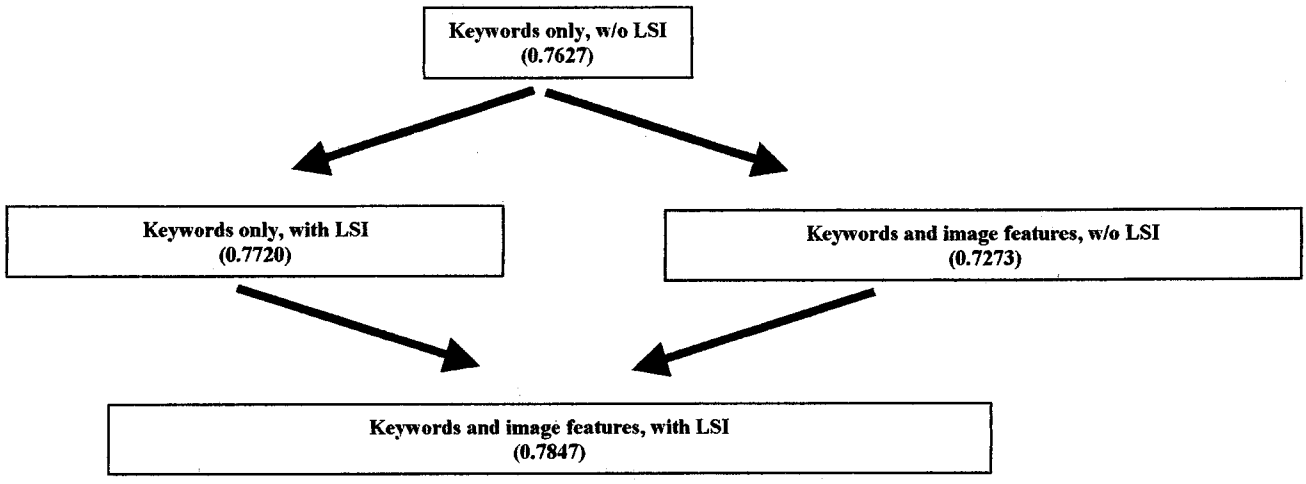

Fig. 4. Results of semantic-based retrieval using keywords and image features (global color histogram).

After this normalization process, each component of the feature-document matrix is a value between zero and one, and thus will not bias the importance of any component in the computation of similarity.

One of the common and effective methods for improving full-text retrieval performance is to apply different weights to different components [11]. We apply these techniques to our experiment. The raw frequency in each component of the feature-document matrix, with or without normalization, can be weighted in a variety of ways. Both global weight and local weight are considered in our approach. A *global weight* indicates the overall importance of that component in the feature vector across the whole document collection. Therefore, the same global weighting is applied to an entire row of the matrix. A *local weight* is applied to each element indicating the relative importance of the component within its vector. The value for any component $\mathbf{A_{i,j}}$ is thus $L(i,j)G(i)$, where $L(i,j)$ is the local weighting for feature component i in document j, and $G(i)$ is the global weighting for that component.

Common local weighting techniques include *term frequency*, *binary*, and *log of term frequency*, whereas common global weighting methods include *Normal*, *GfIdf*, *Idf*, and *Entropy*. Based on previous research, it has been found that $\log(1 + \text{term frequency})$ helps to dampen effects of large differences in frequency and thus has the best performance as a local weight, whereas *Entropy* is the appropriate method for global weighting [11].

The entropy method is defined by having a component global weight of

$$1 + \sum_j \frac{p_{ij} \log (p_{ij})}{\log(number\_of\_documents)} \tag{8}$$

where

$$pij = \frac{f}{gfi}$$

is the probability of that component; tf_{ij} is the raw frequency of component $\mathbf{A_{i,j}}$; and gf_i is the global frequency, i.e., the total number of times that component i occurs in the whole collection.

The global weights give less emphasis to those components that occur frequently or in many images. Theoretically, the entropy method is the most sophisticated weighting scheme, taking the distribution property of feature components over the image collection into account.

We conducted experiments for these four cases:

1) keywords only, no LSI;
2) keywords only, with LSI;
3) keywords and image features (global color histogram), no LSI;
4) keywords and image features (global color histogram), with LSI.

The results are shown in Fig. 4, where the number in parenthesis is the measure-of-goodness of the particular method. These results are based on actual user feedback. It can be noticed that LSI improves the retrieval performance, and is even better when integrated with both textual and visual features, even though just adding visual features without using LSI worsens the retrieval performance. This validates our beliefs that LSI can help discover the correlation between textual features and visual features, and that visual features provide a helping hand when we are retrieving multimedia documents.

Our next approach performs similar experiments utilizing our previously formulated method of color anglograms to represent the spatial color features of images [37], [42]. This is a novel spatial color indexing scheme based on the point feature map obtained by dividing an image evenly into a number of nonoverlapping blocks with each individual block abstracted as a unique feature point labeled with its spatial location, dominant/average hue, and dominant/ average saturation. Fig. 5(a) shows a pyramid image of size 192×128. By dividing the image into 256 blocks, Fig. 5(b) and 5(c) shows the image approximation using dominant hue and saturation values to represent each block, respectively. Fig. 5(d) presents the corresponding point feature map perceptually. Fig. 5(e) is the Delaunay triangulation of the set of feature points labeled with saturation value 5, and Fig. 5(f) shows the corresponding anglogram obtained by counting the two largest angles of each triangle.

For our experiments, we first normalize the images to size 192×128, and then divide each of the images into 64 blocks.

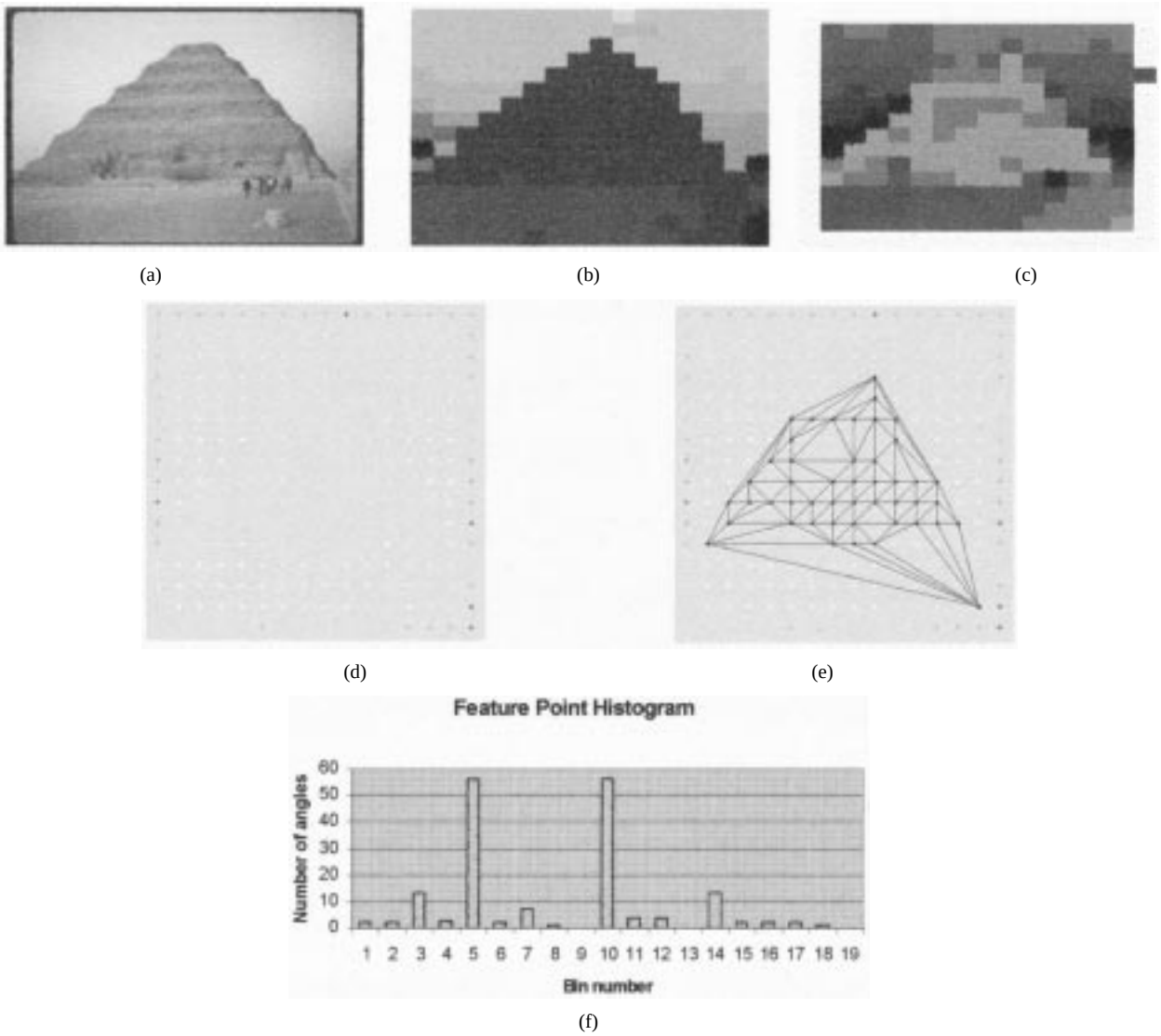

Fig. 5. (a). Pyramid image (b). Hue component (c). Saturation component (d). Point feature map (e). Delaunay triangulation of saturation 5 (f). Anglogram of saturation 5.

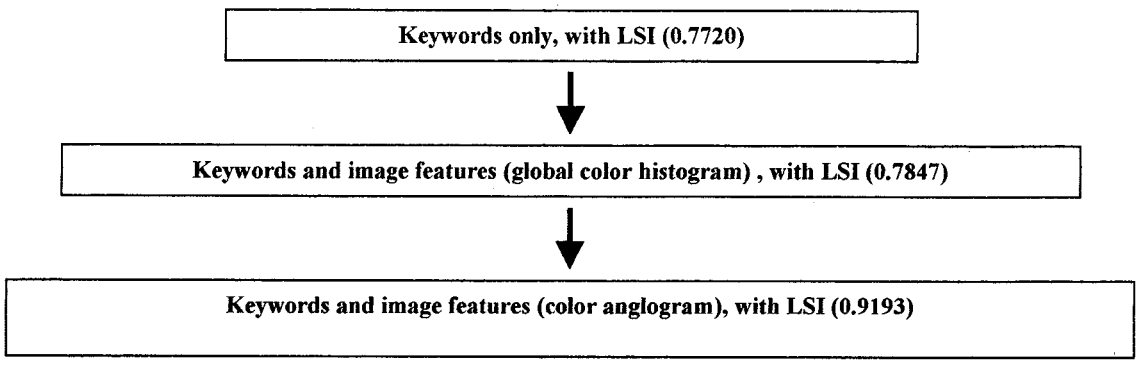

Fig. 6. Results of semantic-based retrieval using both keywords and image features (color anglogram).

We have ten quantized average hue values and ten average saturation values. We count the two largest angles of each triangle, and have an anglogram bin of 5°. Our vector representation of an image thus has 720 elements: 36 bins for each of ten hue values and 36 bins for each of ten saturation values. In this case, the feature-document matrix is 763×20. Of each feature vector, the first 720 elements are visual elements, i.e., the color anglogram of each image, and the last 43 elements are textual features corresponding to the keywords. We reduce the dimensionality of the feature-document matrix to $k = 12$. We follow the same query process as in the previous section. Using the measure of goodness, we show the results in Fig. 6.

From these results, one notices that our color anglogram method is better than the global color histogram in capturing image features, which is consistent with our previous results [34], [37], [42]. One also notices that LSI improves the retrieval performance further when integrated with this method. Once again our experiments validate that finding the semantics of web documents and integrating textual and visual features are promising approaches to more meaningful retrieval of multimedia web documents.

VI. Conclusions

In this paper, we have presented an approach to negotiating the gap between low-level features and high-level concepts in web documents. We have examined the use of LSI for content-based web retrieval, using both keywords and image features of the documents. Two different approaches to image feature representation, namely, color histograms and color anglograms, are adopted and evaluated. This paper has shown the use of image features in improving text retrieval, whereas our previous papers [40], [42] have shown the use of textual keywords to improve image retrieval.

First of all, experimental results show that LSI is able to correlate the semantically similar keywords to construct concept clusters, and it is also able to correlate keywords with image features in the web documents. Using LSI to discover the underlying semantic structure of web documents is a promising approach to understanding the documents on a higher level, which better reflects human perception. Second, we validated that integrating textual features with visual features, together with LSI, can represent the contents of multimedia web document better than using keywords only. Therefore, it helps improve the retrieval performance significantly. Finally, once again our spatial feature representation technique, the color anglogram, proves to be powerful in capturing and representing image features in an effective and efficient way.

We propose to use the anglogram technique for representing the structural features of web documents, where we call it a *structure anglogram*. Similar to the color anglogram approach on images, each web document is divided into blocks. The dominant tag in each block, i.e., the tag whose rendering takes up most of the area in the given block is then used as the relevant feature. A feature vector can be constructed by computing the anglogram of the feature points in the document in a process similar to that used in the color anglogram. Then we propose to integrate textual, visual, and structural features all together to further improve the retrieval performance. One of the strengths of the LSI technique is that it is a vector-based method which easily integrates different features into one feature vector and treats them just as similar components. Hence, ostensibly, we can expand the feature vector by adding even more features without any concern.

Currently we are also experimenting with various clustering techniques for web documents, and comparing the performance with that of using LSI. We are also planning to apply the LSI technique to web document prefetching and web mining, as well as studying how to make use of the characteristics of XML to better extract semantics of web documents from their markup presentation. We firmly believe that semantic-based retrieval is a promising approach to significantly unleashing the power of both the World Wide Web and multimedia technology.

References

[1] Y. H. Ang, Z. Li, and S. H. Ong, "Image retrieval based on multidimensional feature properties," *Proc. SPIE, Storage and Retrieval for Image and Video Databases III*, vol. 2420, pp. 47–57, 1995.

[2] S. Belongie, C. Carson, H. Greenspan, and J. Malik, "Color- and texture-based image segmentation using EM and its application to content-based image retrieval," in *Proc. ICCV*, Mumbai, India, 1998, pp. 675–682.

[3] A. B. Benitez, M. Beigi, and S. F. Chang, "Using relevance feedback in content-based image metasearch," *IEEE Internet Comput.*, vol. 2, pp. 59–69, July-Aug. 1998.

[4] M. Berry, Z. Drmac, and E. Jessup, "Matrices, vector spaces, and information retrieval," *SIAM Rev.*, vol. 41, no. 2, pp. 335–362, 1999.

[5] M. Berry, S. T. Dumais, and G. W. O'Brien, "Using linear algebra for intelligent information retrieval," *SIAM Rev.*, pp. 573–595, 1995.

[6] J. Bigum, "Unsupervised feature reduction in image segmentation by local transforms," *Pattern Recognit. Lett.*, vol. 14, pp. 573–583, 1993.

[7] C. H. Chang and C. C. Hsu, "Enabling concept-based relevance feedback for information retrieval on the WWW," *IEEE Trans. Knowl. Data Eng.*, vol. 11, no. 4, pp. 595–609, 1999.

[8] S. F. Chang, W. Chen, and H. Sundaram, "Semantic visual templates: Linking visual features to semantics," in *Proc. IEEE ICIP*, Chicago, IL, 1998.

[9] C. Colombo, A. Del Bimbo, and P. Pala, "Semantics in visual information retrieval," *IEEE Multimedia*, vol. 6, pp. 38–53, July-Sept. 1999.

[10] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *J. Amer. Soc. Inf. Sci.*, vol. 41, no. 6, pp. 391–407, 1990.

[11] S. Dumais, "Improving the retrieval of information from external sources," *Behav. Res. Meth., Instrum., Comput.*, vol. 23, no. 2, pp. 229–236, 1991.

[12] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, pp. 211–218, 1936.

[13] G. H. Golub and C. Van Loan, *Matrix Computation*. Baltimore, MD: Johns Hopkins Univ. Press, 1996.

[14] V. N. Gudivada and V. V. Raghavan, "Content-based image retrieval systems," *IEEE Comput.*, vol. 28, pp. 18–22, Sept. 1995.

[15] D. Hawking, "Results and challenges in web search evaluation," in *Proc. 8th IWWWC*, Toronto, ON, Canada, 1999.

[16] M. A. Hearst and J. O. Pedersen, "Reexamining the cluster hypothesis: scatter/gather on retrieval results," in *Proc. 19th Annu. Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, Zürich, Switzerland, 1996, pp. 76–84.

[17] J. Huang, S. R. Kumar, M. Mitra, W. J. Zhu, and R. Zabih, "Spatial color indexing and applications," *Int. J. Comput. Vis.*, vol. 35, no. 3, pp. 245–268, 1999.

[18] J. Huang, S. R. Kumar, and R. Zabih, "An automatic hierarchical image classification scheme," in *Proc. 6th ACM ICM*, Bristol, U.K., 1998, pp. 219–228.

[19] A. K. Jain and A. Vilaya, "Image retrieval using color and shape," *Pattern Recognit.*, vol. 29, no. 8, pp. 1233–1244, 1996.

[20] M. La Cascia, S. Sethi, and S. Sclaroff, "Combining textual and visual cues for content-based image retrieval on the WWW," in *Proc. IEEE Workshop Content-Based Access of Image and Video Libraries*, Santa Barbara, CA, 1998, pp. 24–28.

[21] T. P. Minka and R. W. Picard, "Interactive learning with a society of models," *Pattern Recognit.*, vol. 30, no. 4, pp. 565–581, 1997.

[22] B. M. Mehtre, M. S. Kankanhalli, and W. F. Lee, "Content-based image retrieval using a composite color-shape approach," *Inf. Process. Manage.*, vol. 34, no. 1, pp. 109–120, 1998.

[23] W. Niblack, R. Barder, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Yaubin, "The QBIC project: Querying images by content using color, texture, and shape," *Proc. SPIE, Storage and Retrieval for Image and Video Databases*, vol. 1908, pp. 173–181, 1993.

[24] G. Pass and R. Zabih, "Histogram refinement for content-based image retrieval," in *IEEE Workshop Applications of Computer Vision*, Sarasota, FL, 1996, pp. 96–102.

[25] A. Pentland, R. W. Piccard, and S. Sclaroff, "Photobook: Content-based manipulation of image databases," *Int. J. Comput. Vis.*, vol. 18, no. 3, pp. 233–254, 1996.

[26] G. Pringle, L. Allison, and D. L. Dowe, "What is a tall poppy among web pages?," in *Proc. 7th IWWWC*, Brisbane, Australia, 1998.
[27] F. Rabitti and P. Stanchev, "GRIM_DBMS: A graphical image database system," in *Visual Database Systems*, T. Kunii, Ed. Amsterdam, The Netherlands: North-Holland, 1989, pp. 415–430.
[28] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: A powerful tool in interactive content-based image retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 644–655, 1998.
[29] S. Santini and R. Jain, "Integrated browsing and querying for image databases," *IEEE Multimedia*, pp. 26–39, July-Sept. 2000.
[30] S. Schechter, M. Krishnam, and M. D. Smith, "Using path profiles to predict HTTP requests," in *Proc. 7th IWWWC*, Brisbane, Australia, 1998.
[31] I. K. Sethi, I. Coman, B. Day, F. Jiang, D. Li, J. Segovia-Juarez, G. Wei, and B. You, "Color-WISE: A system for image similarity retrieval using color," *Proc. SPIE , Storage and Retrieval for Image and Video Databases*, vol. 3312, pp. 140–149, 1998.
[32] G. Sheikholeslami, W. Chang, and A. Zhang, "Semantic clustering and querying on heterogeneous features for visual data," in *Proc. 6th ACM ICM*, Bristol, U.K, 1998, pp. 3–12.
[33] M. Stricker and A. Dimai, "Color indexing with weak spatial constraints," *Proc. SPIE Storage and Retrieval for Image and Video Databases*, vol. 2670, pp. 29–40, 1996.
[34] Y. Tao and W. I. Grosky, "Delaunay Triangulation for image object indexing: A novel method for shape representation," in *Proc. IS&T/SPIE Symp. Storage and Retrieval for Image and Video Databases VII*, San Jose, CA, Jan. 23–29, 1999, pp. 631–642.
[35] ——, "Object-Based Image Retrieval Using Point Feature Maps," in *Proc. 8th IFIP 2.6 Working Conf. Database Semantics (DS8)*, Rotorua, New Zealand, Jan. 5–8, 1999.
[36] ——, "Spatial color indexing: A novel approach for content-based image retrieval," in *Proc. ICMCS*, Florence, Italy, 1999, pp. 530–535.
[37] ——, "Spatial color indexing using rotation, translation, and scale invariant anglograms," *Multimedia Tools Applicat.*, vol. 15, no. 3, pp. 247–268, Dec. 2001.
[38] Y. Tao, W. I. Grosky, L. Zamorano, Z. Jiang, and J. Gong, "Segmentation and representation of lesions in MRI brain images," *Proc. SPIE, Medical Imaging*, pp. 930–939, 1999.
[39] O. Zamir and O. Etzioni, "Web document clustering: A feasibility demonstration," in *Proc. 21st Annu. Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, Melbourne, Australia, 1998, pp. 46–54.
[40] R. Zhao and W. I. Grosky, "From features to semantics: some preliminary results," in *Proc. IEEE ICME*, New York, 2000, pp. 679–682.
[41] ——, "Bridging the semantic gap in image retrieval," in *Distributed Multimedia Databases: Techniques and Applications*, T. K. Shih, Ed. Hershey, PA: Idea Group, 2001, pp. 14–36.
[42] ——, "Negotiating the semantic gap: from feature maps to semantic landscapes," *Pattern Recognit.*, vol. 35, no. 3, pp. 593–600, 2002.

Rong Zhao (M'02) received the B.Eng. degree in computer science and technology from Tsinghua University, Beijing, China, in 1996, and the Ph.D. degree in computer science from Wayne State University, Detroit, MI, in 2001.

He is currently a Research Assistant Professor in the Department of Computer Science at the State University of New York at Stony Brook. He was with the Visual Information Management Group in the Imaging Science and Technology Laboratory of the Eastman Kodak Company in the summer of 1999. His current research interests are semantic-based multimedia data mining, semantic-based Web data mining, information retrieval, databases and digital library, pattern recognition, computer vision, image processing, and user interfaces.

William I. Grosky received the B.S. degree in mathematics from the Massachusetts Institute of Technology, Cambridge, in 1965, the M.S. degree in applied mathematics from Brown University, Providence, RI, in 1968, and the Ph.D. degree in engineering and applied science from Yale University, New Haven, CT, in 1971.

He is currently Professor and Chair of the Department of Computer and Information Science at University of Michigan, Dearborn. Prior to joining the University of Michigan in 2001, he was Professor and Acting Chair of the Department of Computer Science at Wayne State University, Detroit, MI. Before joining Wayne State University in 1976, he was an Assistant Professor in the Department of Information and Computer Science at Georgia Institute of Technology, Atlanta. His current research interests include databases, data mining, information retrieval, and the semantic web, as they apply to multimedia information.