

PART II:

**CONTENT-BASED RETRIEVAL
AND IMAGE DATABASE TECHNIQUES**

Chapter II

Bridging the Semantic Gap in Image Retrieval

Rong Zhao and William I. Grosky
Wayne State University, USA

The emergence of multimedia technology and the rapidly expanding image and video collections on the internet have attracted significant research efforts in providing tools for effective retrieval and management of visual data. Image retrieval is based on the availability of a representation scheme of image content. Image content descriptors may be visual features such as color, texture, shape, and spatial relationships, or semantic primitives.

Conventional information retrieval was based solely on text, and those approaches to textual information retrieval have been transplanted into image retrieval in a variety of ways. However, “a picture is worth a thousand words”. Image contents are much more versatile compared with texts, and the amount of visual data is already enormous and still expanding very rapidly. Hoping to cope with these special characteristics of visual data, content-based image retrieval methods have been introduced. It has been widely recognized that the family of image retrieval techniques should become an integration of both low-level visual features addressing the more detailed perceptual aspects and high-level semantic features underlying the more general conceptual aspects of visual data. Neither of these two types of features is sufficient to retrieve or manage visual data in an effective or efficient way. Although efforts have been devoted to combining these two aspects of visual data, the gap between them is still a huge barrier in front of researchers. Intuitive and heuristic approaches do not provide us with satisfactory performance. Therefore, there is an urgent need of finding the latent correlation between low-level features and high-level concepts and merging them from a different perspective. How to find this new perspective and bridge the gap between visual features and semantic features has been a major challenge in this research field. Our paper addresses these issues.

INTRODUCTION

The emergence of multimedia technology and the rapidly expanding image and video collections on the Internet have attracted significant research efforts in providing tools for effective retrieval and management of visual data. Image retrieval is based on the availability of a representation scheme of image content. Image content descriptors may be visual features such as color, texture, shape, and spatial relationships, or semantic primitives.

Conventional information retrieval was based solely on text, and those approaches to textual information retrieval have been transplanted into image retrieval in a variety of ways. However, “a picture is worth a thousand words”. Image contents are much more versatile compared with texts, and the amount of visual data is already enormous and still expanding very rapidly. Hoping to cope with these special characteristics of visual data, content-based image retrieval methods have been introduced. It has been widely recognized that the family of image retrieval techniques should become an integration of both low-level visual features addressing the more detailed perceptual aspects and high-level semantic features underlying the more general conceptual aspects of visual data. Neither of these two types of features is sufficient to retrieve or manage visual data in an effective or efficient way (Smeulders, et al., 2000). Although efforts have been devoted to combining these two aspects of visual data, the gap between them is still a huge barrier in front of researchers. Intuitive and heuristic approaches do not provide us with satisfactory performance. Therefore, there is an urgent need of finding the latent correlation between low-level features and high-level concepts and merging them from a different perspective. How to find this new perspective and bridge the gap between visual features and semantic features has been a major challenge in this research field.

Image Retrieval

Image retrieval is an extension to traditional information retrieval. Approaches to image retrieval are somehow derived from conventional information retrieval and are designed to manage the more versatile and enormous amount of visual data which exist.

Low-level visual features such as color, texture, shape and spatial relationships are directly related to perceptual aspects of image content. Since it is usually easy to extract and represent these features and fairly convenient to design similarity measures by using the statistical properties of these features, a variety of content-based image retrieval techniques have been proposed in the past few years. High-level concepts, however, are not extracted directly from visual contents, but they represent the relatively more important meanings of objects and scenes in the images that are perceived by human beings. These conceptual aspects are more closely related to users' preferences and subjectivity. Concepts may vary significantly in different circumstances. Subtle changes in the semantics may lead to dramatic conceptual differences. Needless to say, it is a very challenging task to extract and manage meaningful semantics and to make use of them to achieve more intelligent and user-friendly retrieval.

Challenges

High-level conceptual information is normally represented by using text descriptors. Traditional indexing for image retrieval is text-based (Jorgensen, 1998). In certain content-based retrieval techniques, text descriptors are also used to model perceptual aspects (Kelly & Cannon, 1995; Gimel'farb & Jain, 1996). Unfortunately, the inadequacy of text description is an obvious and very problematic issue. Meanwhile, image contents are much more

complicated than textual data stored in traditional databases, yet there is an even greater demand for retrieval and management tools for visual data, since visual information is a more capable medium of conveying ideas and is more closely related to human perception of the real world. Image retrieval techniques should provide support for user queries in an effective and efficient way, just as conventional information retrieval does for textual retrieval.

However, the dynamic and versatile characteristics of image content require expensive computations and sophisticated methodologies in the areas of computer vision, image processing, data visualization, indexing, and similarity measurement. In order to manage image data effectively and efficiently, several schemes for data modeling and image representation have been proposed (Ahmad & Grosky, 1997; Aslandogan, et al., 1995; Chang & Liu, 1984; Chang & Wu, 1992; Chang, Shi, & Yan, 1987; Gudivada, 1995; Gudivada, 1997; Huang & Jean, 1994; Pentland, Picard, & Sclaroff, 1994; Smith & Chang, 1998; Tao & Grosky, 1999). Due to the lack of any unified framework for image representation and retrieval, certain methods may perform better than others under certain query situations. Therefore, these schemes and retrieval techniques have to be somehow integrated and adjusted on the fly to facilitate effective and efficient image data management.

Research Goals

The motivation of our research is to improve several aspects of content-based image retrieval by finding the latent correlation between low-level visual features and high-level semantics and integrating them into a unified vector space model. To be more specific, the significance of this approach is to design and implement an effective and efficient framework of image retrieval techniques, using a variety of visual features such as color, texture, shape and spatial relationships. Latent semantic indexing, an information retrieval technique, is incorporated with content-based image retrieval. By using this technique, we aim to extract the underlying semantic structure of image content and hence to bridge the gap between low-level features and high-level concepts. Improved retrieval performance and more efficient indexing structure can also be achieved.

The remaining part of this chapter is organized as follows. The next section provides a brief overview of related techniques of content-based image retrieval and a special look at the relationship between low-level features and high-level semantics. Then we introduce the feature extraction techniques applied in this research, followed by the theoretical background of latent semantic indexing and a brief description of its application in textual information retrieval. Then we present the experimental results of our semantic-based image retrieval technique. The last section summarizes the chapter and highlights some of our proposed future work.

EXISTING TECHNIQUES

Visual contents, which include, but are not limited to, color, texture, shape and spatial constraints, is an integral part of multimedia information systems. In the past few years, content-based image retrieval has seen a great deal of emphasis in the context of multimedia databases (Grosky, 1997).

Since the proposed research incorporates multiple visual feature extraction techniques, the use of latent semantic indexing to find the correlation of features and semantics, and the integration of visual features and textual annotations, in the following sections a

brief review of these subjects is provided and a few related works are analyzed. Other issues in image retrieval, such as high dimensional indexing, visualization and browsing, human interaction and knowledge engineering, are out of the scope of this chapter. For detailed information of those topics please refer to (Baeza-Yates & Ribeiro-Neto, 1999; Del Bimbo, 1999; Grosky, Jain, & Mehrotra, 1997; Rui, Huang, & Chang, 1998; Smeulders, et al., 2000).

Content-Based Image Retrieval

Visual feature extraction is the basis of any content-based image retrieval technique. Widely used features include color, texture, shape and spatial relationships. Because of perception subjectivity and the complex composition of visual data, there does not exist a single best representation for any given visual feature. Multiple approaches have been introduced for each of these visual features and each of them characterizes the feature from a different perspective.

Color is one of the most widely used visual features in content-based image retrieval. It is relatively robust and simple to represent. Various studies of color perception and color spaces have been proposed (Rui, Huang, & Chang, 1998; Smeulders, et al., 2000). The color histogram is the most commonly used representation technique, statistically describing the combined probabilistic property of the three color channels. Swain and Ballard proposed the *histogram intersection* measure that has been a fairly standard metric for analyzing histogram based features (Swain & Ballard, 1991).

Texture refers to the patterns in an image that present the properties of homogeneity that do not result from the presence of a single color or intensity value. It is a powerful discriminating feature, present almost everywhere in nature. However, it is almost impossible to describe texture in words, because it is virtually a statistical and structural property. There are three major categories of texture-based techniques (Gimel'farb & Jain, 1996), namely, *probabilistic/statistical*, *spectral*, and *structural* approaches. The well known *Tamura features* were introduced in (Tamura, 1978), which include *coarseness*, *contrast*, *directionality*, *line-likeness*, *regularity* and *roughness*.

Shape representation is normally required to be invariant to *translation*, *rotation*, and *scaling*. In general, shape representations can be categorized into either *boundary-based* or *region-based*. The former uses only the outer boundary characteristics of the entities while the latter uses the entire region. Well known methods include Fourier descriptors and moment invariants (Rui, Huang, & Chang, 1998). In our shape-based image retrieval approach, we introduce the *anglogram* method, a geometrically computed representation technique based on Delaunay triangulation (Tao & Grosky, 1999). This representation is relatively efficient and invariant to translation, rotation, and scaling.

Spatial color indexing has attracted more and more interest in the content-based image retrieval field. One of the earliest image retrieval projects was QBIC (Niblack, et al., 1993). Other research groups (Belongie, et al., 1998; Jain & Vilara, 1996; Mehre, Kankanhalli, & Lee, 1998) have also tried to combine color and shape features for improving the performance of image retrieval.

Features vs. Semantics

Needless to say, human beings are much better than computers at extracting and making use of semantic information from images. We believe that complete image understanding should start from interpreting image objects and their relationships. Unfor-

tunately, this goal is still beyond the reach of state-of-the-art in computer vision. As of now, most of the existing approaches are still based on manual annotations of semantic information related to the objects of concern or extraction of low-level statistical features directly from visual contents. The semantic information associated with image objects can be represented as a class object or embedded inside a class (Grosky & Jiang, 1994; Jiang, Grosky, & Zamorano, 1996).

Grouping low-level image features into meaningful image objects and then automatically attaching correlated semantic descriptions to image objects is still a challenging problem in image retrieval. One of the earliest attempts to automate this procedure was made in (Krey, et al., 1997; Roper, et al., 1997). Following the approach of syntactical pattern recognition, they propose a graph grammar to handle object recognition. This grammar consists of three different object types: goals, terminals, and non-terminals. Terminal nodes are represented by the inputs of the color, texture, and contour modules. The non-terminal nodes are composed of color, texture, and contour segments. Thus it follows that the non-terminal nodes are divided into different object classes: the primitive objects, which are just supported by the color, texture, and contour segments (specifying the grammar about the primitive objects) and the complex objects that rest on the primitive objects. The goals are recognized image objects and various scene descriptions.

VISUAL FEATURES

In this section we present the feature extraction techniques that are applied in this research work. We propose to integrate a variety of visual features with the latent semantic indexing technique for image retrieval. These visual features include global and subimage color histograms, as well as anglograms. Anglograms can be used for shape-based and color-based representations, as well as for the representation of spatial relationships of image objects. Thus, a unified framework of image retrieval techniques is going to be generated in our proposed study.

Color Histogram

Color histogram is the most traditional and the most widely used way to represent color patterns in an image. It is a relatively efficient representation of color content and it is fairly insensitive to variations originated by camera rotation or zooming (Del Bimbo, 1999; Smeulders, et al., 2000). Also, it is fairly insensitive to changes in image resolution when images have quite large homogeneous regions, and insensitive to partial occlusions as well.

In our study, the *HSV* color histogram is generated for each image on either the whole image level or the subimage level. On whole image level, a two-dimensional global histogram of both the hue component and saturation component is computed. Since the human perception of color depends mostly on hue and saturation, we ignore the intensity value component in our preliminary research, in order to simplify the computation. Each image is first 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 taken to be the representing feature vector of each image. This is a vector of 100 elements, $V = [f_1, f_2, f_3, \dots, f_{100}]^T$, where each element corresponds to one of the bins in the hue-saturation histogram.

On the subimage level, each image is decomposed into 5 subimages, which is

illustrated by the sample image in Figure 1. Such an approach was used in (Stricker & Dimai, 1996), and is a step toward identifying the *semcons* (Grosky, Fotouhi, & Jiang, 1998) appearing in an image. Considering that it is very common to have the major object located in central position in the image, we have one subimage to capture the central region in each image, and the other four subimages cover the upper-left, upper-right, lower-left, and lower-right areas in the image. For each pixel of the resulting subimage, hue and saturation are extracted and each quantized into a 10-bin histogram. Then the two histograms h and s are again combined into one $h \infty s$ histogram with 100 bins, which is taken to be the representing feature vector of each image. This is a vector of 100 elements, $V = [f_1, f_2, f_3, \dots, f_{100}]^T$, where each element again corresponds to one of the bins in the hue-saturation histogram. Since the global and subimage color histograms are formulated as a feature vector, it is very easy to use them as the input for latent semantic indexing.

Anglogram

In this section, we first provide some theoretical background of Delaunay triangulation, and then introduce the triangulation-based *anglogram* method for encoding spatial correlation, which is invariant to translation, scale, and rotation.

Let $P = \{p_1, p_2, \dots, p_n\}$ be a set of points in the two-dimensional Euclidean plane, namely the *sites*. Partition the plane by labeling each point in the plane to its nearest site. All those points labeled as p_i form the *Voronoi region* $V(p_i)$. $V(p_i)$ consists of all the points x , which are at least as close to p_i as to any other site:

$$V(p_i) = \{x: |p_i - x| \leq |p_j - x|, \forall j \neq i\}$$

Some of the points do not have a unique nearest site, however. The set of all points that have more than one nearest site form the *Voronoi diagram* $V(P)$ for the set of sites.

Construct the dual graph G for a Voronoi Diagram $V(P)$ as follows: the nodes of G are the sites of $V(P)$, and two nodes are connected by an arc if their corresponding Voronoi polygons share a Voronoi edge. Delaunay proved that when the dual graph is drawn with straight lines, it produces a planar triangulation of the Voronoi sites P , so called the *Delaunay triangulation* $D(P)$. Each face of $D(P)$ is a triangle, so called the *Delaunay triangle*.

For example, Figure 2 shows the Voronoi diagram for 18 sites. Figure 3 shows the corresponding Delaunay triangulation for the sites shown in Figure 2, and Figure 4 shows the Voronoi diagram in Figure 2 superimposed on the corresponding Delaunay triangulation in Figure 3. We note that it is not immediately obvious that using straight lines in the dual graph would avoid crossings in the dual. The dual segment between two sites does not necessarily cross the Voronoi edge shared between their Voronoi regions, as illustrated in Figure 4.

The proof of Delaunay's theorems and properties are beyond the scope of this chapter, but can be found in (O'Rourke, 1994). Among various algorithms for constructing the Delaunay triangulation of a set of N points, we note that some of them have a worst-case complexity of $O(N \log N)$ (Dwyer, 1987; Fortune, 1987).

The spatial layout of a set of feature points can be coded through such a triangulation-based process, namely, an *anglogram*. One discretizes and counts the angles produced by the Delaunay triangulation of a set of unique feature points in some context, given the selection criteria of what the bin size will be and of which angles will contribute to the final angle histogram. An important property of our proposed anglogram for encoding spatial correlation is its invariance to translation, scale, and rotation. An $O(\max(N, \#bins))$ algorithm is necessary to compute the anglogram corresponding to the Delaunay triangulation.

lation of a set of N points.

A *shape anglogram* can be used for image object indexing, while the *color anglogram* can be used as a spatial color representation technique.

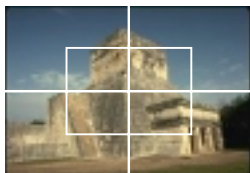
In the *shape anglogram* approach, those objects that will be used to index the image are identified, and then a set of high-curvature points along the object boundary are obtained as the feature points. The Delaunay triangulation is performed on these feature points and thus the feature point histogram is computed by discretizing and counting the number of either the two largest angles or the two smallest angles in the Delaunay triangles.

To apply the *color anglogram* approach, color features and their spatial relationship are extracted and then coded into the Delaunay triangulation. Each image is decomposed into a number of non-overlapping blocks. Each individual block is abstracted as a unique feature point labeled with its spatial location and feature values. The feature values in our experiment are dominant or average hue and saturation in the corresponding block. Then, all the normalized feature points form a point feature map for the corresponding image. For each set of feature points labeled with a particular feature value, the Delaunay triangulation is constructed and then the feature point histogram is computed by discretizing and counting the number of either the two largest angles or the two smallest angles in the Delaunay triangles. Finally, the image will be indexed by using the concatenated feature point histogram for each feature value. An example is shown in Figure 5. Figure 5(a) shows a pyramid image of size 192×128 . By dividing the image evenly into 256 blocks, Figure 5(b) and Figure 5(c) show the image approximation using dominant hue and saturation values to represent each block, respectively. Figure 5(d) shows the corresponding point feature map perceptually. Figure 5(e) shows the resulting Delaunay triangulation of a set of feature points labeled with saturation 5, and Figure 5(f) shows the corresponding *anglogram* (feature point histogram) obtained by counting only the two largest angles out of each individual Delaunay triangle. A sample query with color anglogram is shown in Figure 6.

LATENT SEMANTIC INDEXING

In this section we describe an approach to automatic information indexing and retrieval, namely, *Latent Semantic Indexing (LSI)*. It is introduced to overcome a fundamental problem that plagues existing retrieval techniques that try to match words of queries with words of documents. The problem is that users want to retrieve on the basis of conceptual content, while individual words 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's query may not match those of a relevant document. In addition,

Figure 1: Subimage Decomposition



most words have multiple meanings and are used in different contexts. Hence, the terms in a user's 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 *polysemy*. *Synonymy* is used to describe the fact that there are many ways to refer to the same object. The prevalence of synonyms tends to decrease the *recall* performance. *Polysemy* refers to the fact that most words have more than one distinct meaning. Polysemy is a factor underlying poor *precision* performance (Deerwester, et al., 1990).

Latent semantic indexing tries to overcome the deficiencies of term-matching retrieval by treating the unreliability of observed term-document association data as a statistical problem. It is assumed that there exists some underlying latent semantic structure in the data that is partially obscured by the randomness of word choice with respect to retrieval. LSI is used to estimate this latent semantic structure, and to get rid of the obscuring noise.

Latent semantic indexing is based on the fact that the term-document association can be formulated by using the vector space model. A low-rank approximation to the vector space representation of the document collection 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

Figure 2: Voronoi Diagram of 18 Sites

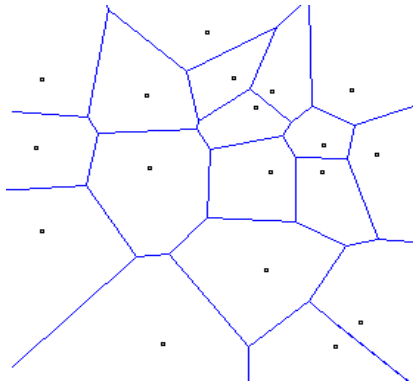


Figure 3: Delaunay Triangulation for the Sites Shown in Figure 2

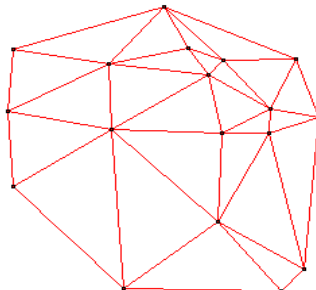
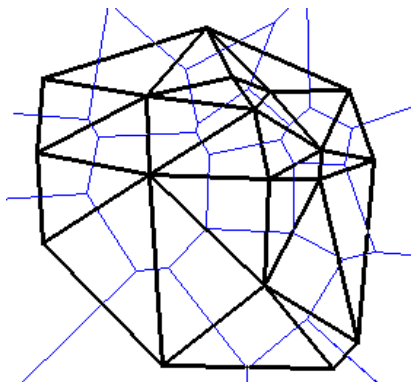


Figure 4: Delaunay Triangulation and Voronoi Diagram Together for the Sites Shown in Figure 2.



represents. According to (Berry, Drmac, & Jessup, 1999), latent semantic indexing has achieved average or above average performance in several experiments with the TREC collections.

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 concept, keyword, or term associated with the given document. The value assigned to that component reflects the importance of the term in representing the semantics of the document. Typically, the value is a function of the frequency with which the term occurs in the document or in the document collection as a whole (Dumais, 1991).

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.

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 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 (Dumais, 1991). 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.

Singular Value Decomposition

Singular Value Decomposition (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 (Berry, Drmac, & Jessup, 1999). The decomposition is defined as follows,

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

where U is the $t \times t$ orthogonal matrix having the left singular vectors of A as its columns, V is the $d \times d$ orthogonal matrix having the right singular vectors of A as its columns, and Σ is the $t \times d$ diagonal matrix having the singular values $\sigma_1 \square \sigma_2 \square \dots \square \sigma_r$ of the matrix A in order along its diagonal, where $r = \min(t, d)$. This decomposition exists for any given matrix A (Golub & Van Loan, 1996).

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,



Figure 5(a) A Pyramid Image



Figure 5(b) Hue Component



Figure 5(c) Saturation Component

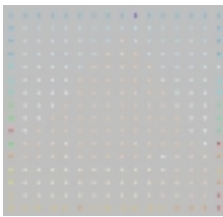


Figure 5(d) Point Feature Map

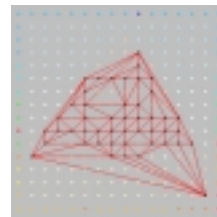


Figure 5(e) Delaunay Triangulation of Saturation 5

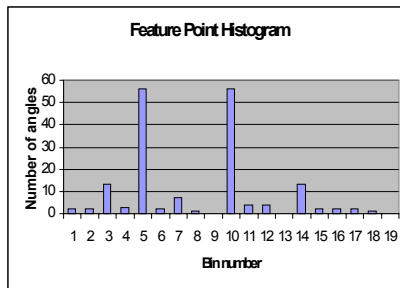


Figure 5(f) Anglogram of Saturation 5

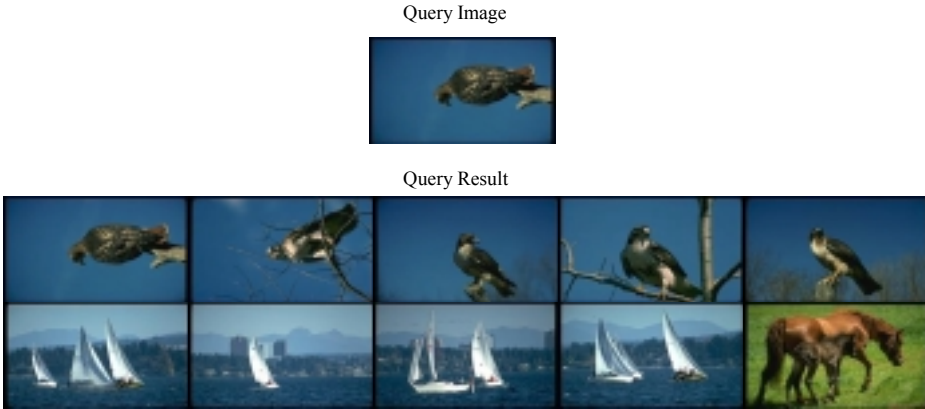


Figure 6 Query with Color Anglogram

$$\|A\|_F = \|U\Sigma V^T\|_F = \|\Sigma V^T\|_F = \|\Sigma\|_F = \sqrt{\sum_{j=1}^{r_A} \sigma_j^2}$$

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 $k < r_A$, we can set all but the k largest singular values of A to be zero. A classic theorem about the singular value decomposition by Eckart and Young (1936) 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

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

Here $A_k = U_k \Sigma_k V_k^T$, where U_k is the $t \times k$ matrix whose columns are the first k columns of matrix U , V_k is the $d \times k$ matrix whose columns are the first k columns of matrix V , and Σ_k is the $k \times k$ diagonal matrix whose diagonal elements are the k largest singular values of matrix A .

How to choose the rank that provides optimal performance of latent semantic indexing for any given database remains an open question and is normally decided via empirical testing (Berry, Dumais, & O'Brien, 1995). For very large databases, the number of dimensions used usually ranges between 100 and 300 (Letsche & Berry, 1997). Normally, it is a choice made for computational feasibility as opposed to accuracy. Using the *SVD* to find the approximation A_k , however, guarantees that the approximation is the best that can be achieved for any given choice of k .

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 $a_j, j = 1, 2, \dots, d$, those d cosines are computed according to the following formula

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

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

for any t -dimensional vector x .

The latent semantic indexing technique has been successfully applied to information retrieval, in which it shows distinctive power of finding the latent correlation between terms and documents. This inspires us to apply this technique to image retrieval. We aim to reveal the underlying semantic nature of image contents, and thus to find the correlation between visual features and semantics of visual documents or objects.

FINDING LATENT CORRELATION BETWEEN VISUAL FEATURES AND SEMANTICS

Existing management systems for image collections and their users are typically at cross-purposes. While these systems normally retrieve images based on low-level features, users usually have a more abstract notion of what will satisfy them. Using low-level features to correspond to high-level abstractions is one aspect of the *semantic gap* (Gudivada & Raghavan, 1995) between content-based system organization and the concept-based user. Sometimes, the user has in mind a concept so abstract that he himself doesn't know what he wants until he sees it. At that point, he may want images 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 taking place in the image or evoked emotions. Standard definitions of similarity using low-level features generally will not produce good results.

In reality, the correspondence between user-based semantic concepts and system-based low-level features is many-to-many. That is, the same semantic concept will usually be associated with different sets of image features. Also, for the same set of image features, different users could easily find dissimilar images relevant to their needs, such as when their relevance depends directly on an evoked emotion.

In this section, we present the results of a series of experiments that seeks to transform low-level features to a higher level of meaning using latent semantic indexing (Zhao & Grosky, 2000). We examined the use of this technique for content-based image retrieval to find the correlation between visual features and semantics.

Semantic-Based Image Retrieval Using Latent Semantic Indexing with Global and Subimage Color Histograms

Here we present the improvement that latent semantic analysis, normalization, and

weighting can give to two simple and straightforward image retrieval techniques, both of which use standard color histograms. For our experiments, we use a database of 50 JPEG images, each of size 192×128 . This image collection consists of ten semantic categories of five images each. The categories consist of: ancient towers, ancient columns, birds, horses, pyramids, rhinos, sailing scenes, skiing scenes, sphinxes, and sunsets. These images are shown below, in Figure 7.

Our first approach uses global color histograms. Each image is first 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 feature vector of each image. This is a vector of 100 elements, $V = [f_1, f_2, f_3, \dots, f_{100}]^T$, where each element corresponds to one of the bins in the hue-saturation histogram.

We then generate the feature-image-matrix, $\mathbf{A} = [\mathbf{V}_1, \dots, \mathbf{V}_{50}]$, which is 100×50 . Each row corresponds to one of the elements in list of features and each column is the entire feature vector of the corresponding image.

Singular value decomposition is then performed on the feature-image-matrix. The result comprises three matrices, \mathbf{U} , \mathbf{S} and \mathbf{V} , where $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$. The dimensions of \mathbf{U} are 100×100 , \mathbf{S} is 100×50 , and \mathbf{V} is 50×50 . The rank of matrix \mathbf{S} , and thus the rank of matrix \mathbf{A} , in our case is 50. Therefore, the first 50 columns of \mathbf{U} spans the column space of \mathbf{A} and all the 50 rows in \mathbf{V}^T spans the row space of \mathbf{A} . \mathbf{S} is a diagonal matrix of which the diagonal elements are the singular values of \mathbf{A} . To reduce the dimensionality of the transformed latent semantic space, we use a rank- k approximation, A_k , of the matrix \mathbf{A} , for $k = 34$, which worked better than other values tried. This is defined by $A_k = U_k S_k V_k^T$. The dimension of A_k is the same as \mathbf{A} , 100×50 . The dimensions of U_k , S_k , and V_k are 100×34 , 34×34 , and 50×34 , respectively.

The query process in this approach is to compute the distance between the transformed feature vector of the query image, q , and that of each of the 50 images in the database, d . This distance is defined as $dist(q, d) = \mathbf{q}^T \mathbf{d} / \|\mathbf{q}\| \|\mathbf{d}\|$, where $\|\mathbf{q}\|$ and $\|\mathbf{d}\|$ are the norms of those vectors. Using each image as a query, 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 240. A measure that we use of how good a particular method is defined as,

$$measure\text{-of-goodness} = \frac{48 - \text{average-sum} / 5}{45}.$$

We note that in the best case, this measure is equal to 1, whereas in the worst case, it is equal to 0.

This approach was then compared to one without using latent semantic indexing. We also wanted to see whether the standard techniques of normalization and term weighting from text retrieval would work in this environment.

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 image matrix $\mathbf{A} = [\mathbf{V}_1, \mathbf{V}_2, \dots, \mathbf{V}_{50}]$, we have $A_{i,j}$ which is the i^{th} component in vector \mathbf{V}_j . Assuming a Gaussian distribution, we can obtain the mean μ_i and standard deviation σ_i for the i^{th} component of the feature vector

across the whole image database. Then we normalize the original feature image matrix into the range of $[-1, 1]$ as follows,

$$A_{i,j} = \frac{A_{i,j} - \mu_i}{\sigma_i}.$$

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

$$A_{i,j} = \frac{A_{i,j} + 1}{2}.$$

After this normalization process, each component of the feature image matrix is a value between 0 and 1, 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 (Dumais, 1991). We apply these techniques to our image environment. The raw frequency in each component of the feature image 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 image 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 relevant importance of the component with 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 image 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 (Dumais, 1991) it has been found that log of 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.

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

$$1 + \sum_j \frac{p_{ij} \log(p_{ij})}{\log(\text{number_of_images})}$$

where

$$p_{ij} = \frac{tf_{ij}}{gf_i}$$

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 and it takes the distribution property of feature components over the image collection into account.

We conducted similar experiments for these four cases:

1. Global color histograms, no normalization, no term weighting, no latent-semantic indexing (raw data)
2. Global color histograms, normalized and term-weighted, no latent semantic indexing
3. Global color histograms, no normalization, no term-weighting, with latent semantic indexing
4. Global color histograms, normalized and term-weighted, with latent semantic indexing

Results are shown in Table 1, where each table entry is a measure-of-goodness of the corresponding technique. Using normalized and weighted data or using latent semantic indexing with the raw data improves performance, while using both techniques is even better. We note that the improvements under LSI do not seem very large. This is an artifact of the small size and nature of our database and the fact that any of the techniques mentioned work well. It is, however, an indication that LSI is a technique worthy of further study in this environment.

Our next approach uses sub-image matching in conjunction with color histograms. Each image is first converted from the RGB color space to the HSV color space. Each image is decomposed into 5 overlapping subimages, as shown in Figure 1. For the 50 images in our case, 250 subimages will be used in the following feature extraction process. 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.

We then generate the feature-subimage-matrix, $\mathbf{A} = [\mathbf{V}_1, \dots, \mathbf{V}_{250}]$, which is 100×250 . Each row corresponds to one of the elements in the feature vector and each column is the whole feature vector of the corresponding subimage.

Singular value decomposition is then performed on the feature-subimage-matrix. The result comprises three matrices, \mathbf{U} , \mathbf{S} and \mathbf{V} , where $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$. The dimensions of \mathbf{U} are 100×100 , \mathbf{S} is 100×250 , and \mathbf{V} is 250×250 . To reduce the dimensionality of the transformed latent semantic space, we use a rank- k approximation, A_k , of the matrix A , for $k = 55$.

Figure 7(a) Ancient Towers



Figure 7(b) Ancient Columns

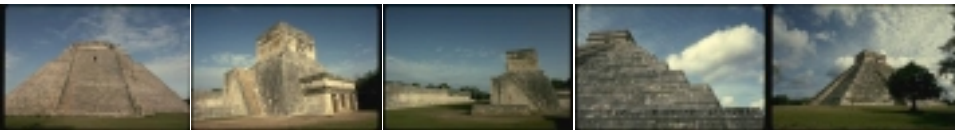


Figure 7(c) Birds

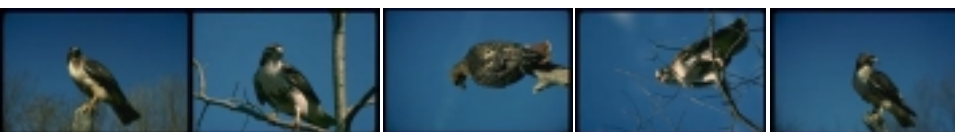


Figure 7(d) Horses



Figure 7(e) Pyramids



Figure 7(f) Rhinos



Figure 7(g) Sailing Scenes



Figure 7(h) Skiing Scenes



Figure 7(i) Sphinxes

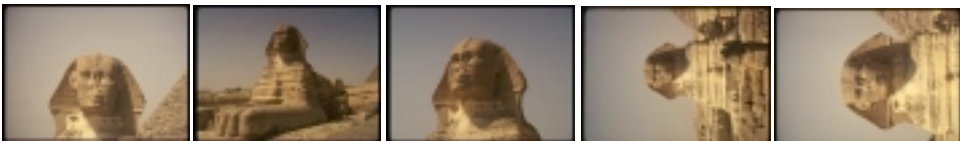


Figure 7(j) Sunsets



The first step of the query process in this approach is to compute the distance between the transformed feature vector of each subimage of the query image, \mathbf{q} , and that of each of the 250 images in the database, \mathbf{d} . This distance is defined as $dist(\mathbf{q}, \mathbf{d}) = \mathbf{q}^T \mathbf{d} / \|\mathbf{q}\| \|\mathbf{d}\|$, where $\|\mathbf{q}\|$ and $\|\mathbf{d}\|$ are the norms of those vectors.

With respect to the query image and each of the 50 database images, we now have the distances between each pair of subimages by the previous step. These distance values $dist(\mathbf{q}_i, \mathbf{d}_i)$ are then combined into one distance value between these two images in an approach similar to the computation of Euclidean distance. Given a query image \mathbf{q} , with corresponding subimages q_1, \dots, q_5 , and a candidate database image \mathbf{d} , with corresponding subimages d_1, \dots, d_5 , we define,

$$dist(q, d) = \frac{1}{5} \sqrt{\sum_{i=1}^5 [dist(q_i, d_i)]^2}$$

This approach was again compared to one without using latent semantic indexing. Each image is decomposed into five subimages which are then represented by their hue-saturation histograms \mathbf{V} . Then the cosine measure between corresponding subimages is computed and used as the similarity metric. We thus have the distance between the query image and each of the 50 database images. These similarity values are then combined into one similarity measure between these two images. Given a query image \mathbf{q} and a candidate image \mathbf{d} in the database, we define,

$$dist(q, d) = \frac{1}{5} \sum_{i=1}^5 sim(q_i, d_i)$$

Using each image as a query, we again find the average sum of the positions of all of the five correct answers. Now, without using latent semantic indexing, using the measure previously introduced, the result is 0.9452, while the use of latent semantic indexing brings this measure to 0.9502.

We also did a similar experiment where $dist(q, d)$ weighted the center subimage twice as much as the peripheral subimages. Using the same measure, the results of these experiments are 0.9475 for the experiment without using latent semantic indexing and 0.9505 for that using latent semantic indexing. Therefore, latent semantic indexing does improve the retrieval performance for both global and subimage color histogram based retrieval. Comparison of global and subimage results shows that subimage provides better performance than global color histogram either with or without latent semantic indexing.

Semantic-Based Image Retrieval Using Latent Semantic Indexing with Color Anglograms

Our next approach performs similar experiments utilizing our previously formulated approach of color anglograms (Tao & Grosky, 2000). In our experiments, we divide the images into 64 non-overlapping blocks, have 10 quantized hue values and 10 quantized saturation values, count the two largest angles for each Delauney triangle, and have an anglogram bin of 5. Our vector representation of an image thus has 720 elements: 36 hue bins for each of 10 hue ranges and 36 saturation bins for each of 10 saturation ranges. We use the same approach to querying as described in the previous section.

We conducted similar experiments for these four cases:

1. Color anglograms, no normalization, no term weighting, no latent-semantic indexing

(raw data)

2. Color anglograms, normalized and term-weighted, no latent semantic indexing
3. Color anglograms, no normalization, no term-weighting, with latent semantic indexing
4. Color anglograms, normalized and term-weighted, with latent semantic indexing

with the results shown in Table 1.

From these results, one notices that our anglogram method is better than global color histogram, which is consistent with our previous results (Tao & Grosky, 1999; Tao & Grosky, 2000). One also notices that latent semantic indexing improves the performance of this method. However, it seems that normalization and weighting has a negative impact on query performance. We more thoroughly examined the impact of these techniques and derived the data shown in Table 2.

The impact of normalization is worse than that of weighting. Normalization is a compacting process which transforms the original feature image matrix (the anglogram elements) to the range $[0, 1]$. Now, the feature image matrix in this case is a sparse matrix with many 0's, some small integers, and a relatively small number of large integers. We believe that these large integers represent the discriminatory power of the anglogram and that the compacting effect of normalization weakens their significance. Local log-weighting also has a compacting effect. Since both the local and global weighting factors lie between 0 and 1, the transformed matrix always has smaller values than the original one, even though no normalization is applied. Thus, normalization and weighting don't help improve the performance, but actually makes it worse.

Utilizing Image Annotations

We conducted various experiments to determine whether image annotations could improve the query results of our various techniques and the results indicate that they can.

For both the global color histogram and color anglogram representation, we appended an extra 15 elements to each of these vectors (called *category bits*) to accommodate the

Table 1 Results for Global Color Histogram and Color Anglogram Representations

	Global Color Histogram	Color Anglogram
Raw Data	0.9257	0.9508
Raw Data with LSI	0.9377	0.9556
Normalized and Weighted Data	0.9419	0.9272
Normalized and Weighted Data with LSI	0.9446	0.9284

Table 2 More Detailed Results for Color Anglogram Representation

	Color Anglogram
Raw Data with LSI	0.9556
Normalized Data with LSI	0.9476
Weighted Data with LSI	0.9529
Normalized and Weighted Data with LSI	0.9284

following 15 keywords associated with these images: *sky, sun, land, water, boat, grass, horse, rhino, bird, human, pyramid, column, tower, sphinx, and snow*. Thus, the feature vector for the global histogram representation now has 115 elements (100 visual elements and 15 textual elements), while the feature vector for the color anglogram representation now has 735 elements (720 visual elements and 15 textual elements). Each image is annotated with appropriate keywords and the area coverage of each of these keywords. For instance, one of the images is annotated with *sky(0.55), sun(0.15), and water(0.30)*. This is a very simple model for incorporating annotation keywords. One of the strengths of the LSA technique is that it is a vector-based method that helps us to integrate easily different features into one feature vector and to treat them just as similar components. Hence, ostensibly, we can apply the normalization and weighting mechanisms introduced in the previous sections to the expanded feature image matrix without any concern.

For the global color histogram representation, we start with an image feature matrix of size 115×50 . Then, using the SVD, we again compute the rank 34 approximation to this matrix, which is also 115×50 . For each query image, we fill bits 101 through 115 with 0's. We also fill the last 15 rows of the transformed image feature matrix with all 0's. Thus, for the querying, *we do not use any annotation information*. We also note, that as before, we apply normalization and weighting, as this improves the results, which are shown in Table 3. The first two results are from Table 1, while the last result shows how our technique of incorporating annotation information improves the querying process.

For the color anglogram representation, we start with an image feature matrix of size 735×50 . Then, using the SVD, we again compute the rank 34 approximation to this matrix, which is also 735×50 . For each query image, we fill bits 721 through 735 with 0's. We also fill the last 15 rows of the transformed image feature matrix with all 0's. Thus, for the querying, *we do not use any annotation information*. We also note that as before, we do not apply normalization and weighting, as this improves the results, which are shown in Table 4. The first two results are from Table 1, while the last result shows how our technique of incorporating annotation information improves the querying process.

Note that annotations improve the query process for color anglograms, even though we do not normalize the various vector components, nor weight them. This is quite surprising, given that the feature image vector consists of 720 visual elements, which are relatively large

Table 3 Global Color Histograms with Annotation Information

	Global Color Histogram
Normalized and Weighted Data	0.9419
Normalized and Weighted Data with LSI	0.9446
Normalized and Weighted Data with LSI and Annotation Information	0.9465

Table 4 Global Color Histograms with Annotation Information

	Color Anglogram
Raw Data	0.9508
Raw Data with LSI	0.9556
Raw Data with LSI and Annotation Information	0.9590

integers, and only 15 annotation elements, which are in the range [0,1].

CONCLUSION AND FUTURE WORK

In this chapter we proposed image retrieval schemes that incorporate multiple visual feature extraction represented by color histograms and color anglograms. Features are extracted on both whole image level and subimage level to better capture salient object descriptions. To negotiate the gap between low-level visual features and high-level concepts, latent semantic indexing is applied and integrated with these content-based retrieval techniques in a vector space model. Correlation between visual features and semantics are explored. Annotations are also fused into the feature vectors to improve the efficiency and effectiveness of the retrieval process.

The length of an image feature vector depends on the number of histogram bins used, which is fixed beforehand. The time complexity of computing all the vectors depends on their length as well as on the image block size used, which is also fixed beforehand. The efficiency and scalability of our technique is bounded by the computationally expensive procedure of finding the singular value decomposition of the image-feature matrix. There have been some papers which have addressed approximation techniques for this computation when the underlying image collections undergoes insertions and deletions (O'Brian, 1994).

Our research provides the following contributions. First, the latent semantic indexing technique is applied to image retrieval and used to uncover the underlying semantic structure of visual contents. The proposed technique is a unified yet open-ended framework that is able to accommodate virtually any vector feature model. Preliminary experiments confirmed that this approach does improve the retrieval performance by linking low-level features and high-level semantics, and better reflects human perception of visual contents. Secondly, the anglogram method, together with latent semantic indexing, provides a robust and efficient indexing scheme for both capturing the spatial relationship of salient image regions and describing object-level concepts. Experiments show that combining the color anglogram and latent semantic indexing achieves the best performance in the comparison of various approaches. Finally, since it is obvious that neither visual features nor textual annotations are sufficient to capture the overall contents of visual data, we propose a seamless integration of visual features and textual annotation, taking advantage of using our vector space model and latent semantic indexing. The combined feature vector, on which latent semantic indexing will be performed afterwards, is normalized and weighted. Preliminary results reveal that it is a very promising approach to further bridging the semantic gap and achieving better retrieval performance. Relevance feedback will also be helpful when incorporated into our proposed scheme.

The results presented in the previous section are quite interesting and are certainly worthy of further study. Our hope is that latent semantic analysis will find that different image features co-occur with similar annotation keywords, and consequently lead to improved techniques of semantic image retrieval. We are currently experimenting with the integration of shape anglograms, color anglograms, and structural anglograms with latent semantic indexing and developing a unified framework to accommodate multiple features and their representation. We will further test and benchmark this integrated image retrieval framework over various large image databases, along with tuning the latent semantic indexing scheme to achieve optimal performance with highly reduced dimensionality. We will further our study of image semantics and incorporation of textual annotations and

explore the correlation between visual feature groups and semantic clusters. We also consider applying various clustering techniques and use the cluster identifier in place of annotation information. Analyzing the patterns of user interaction, either in the query process or in the browsing process, is another interesting research topic. Making use of relevance feedback to infer user preference should also be incorporated to elevate the retrieval performance. Finally, considering that the image archives on the internet are normally associated with other sources of information such as captions, titles, labels, and surrounding texts, we also propose to extend the application of the latent semantic indexing technique to analyze the structure of different types of visual and hypermedia documents. Results of some preliminary experiments show that the integration of latent semantic indexing and the anglogram technique works well in retrieving web documents. For more detail please refer to (Zhao & Grosky, 2001).

REFERENCES

- Ahmad, I., & Grosky, W.I. (1997). Spatial Similarity-based Retrievals and Image Indexing By Hierarchical Decomposition. Proceedings of the International Database Engineering and Application Symposium (IDEAS'97), Montreal, Canada, pp. 269-278.
- Aslandogan, Y. A., Their, C., Yu, C. T., & Liu, C., (1995), Design, Implementation and Evaluation of SCORE, Proceedings of the 11th IEEE International Conference on Data Engineering, Taipei, Taiwan, pp. 280-287.
- Baeza-Yates, R., & Ribeiro-Neto, B., (1999), Modern Information Retrieval, Addison Wesley, New York, NY.
- Belongie, S., Carson, C., Greenspan, H., & Malik, J., (1998), Color- and Texture-based Image Segmentation Using EM and Its Application to Content-Based Image Retrieval, Proceedings of the International Conference on Computer Vision (ICCV '98).
- Berry, M., Drmac, Z., & Jessup, E., (1999), Matrices, Vector Spaces, and Information Retrieval, SIAM Review, Vol. 41, No. 2, pp. 335-362.
- Berry, M., Dumais, S. T., & O'Brien, G. W., (1995), Using Linear Algebra for Intelligent Information Retrieval, SIAM Review, pp. 573-595.
- Chang, S. K., & Liu, S. H., (1984), Picture Indexing and Abstraction Techniques for Pictorial Databases. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 6, No. 4, pp. 475-484.
- Chang, C. C., & Wu, T. C., (1992), Retrieving the Most Similar Symbolic Pictures from Pictorial Databases, Information Processing and Management, Vol. 28, No. 5, pp. 581-588.
- Chang, S. K., Shi, Q. Y., & Yan, C. W., (1987), Iconic Indexing by 2-D Strings, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 9, No. 3, pp. 413-428.
- Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R., (1990), Indexing by Latent Semantic Analysis, Journal of the American Society for Information Science, Volume 41, Number 6, pp. 391-407.
- Del Bimbo, A., (1999), Visual Information Retrieval, Morgan Kaufmann, San Francisco, CA.
- Dumais, S., (1991), Improving the Retrieval of Information from External Sources, Behavior Research Methods, Instruments, and Computers, Vol. 23, Number 2, pp. 229-236.
- Dwyer, R. A., (1987), A Faster Divide-and-Conquer Algorithm for Constructing Delaunay

- Triangulations, Algorithmic, Vol. 2, No. 2, pp. 127-151.
- Eckart, C., & Young, G., (1936), The Approximation of One Matrix by Another of Lower Rank, *Psychometrika*, pp. 211-218.
- Fortune, S., (1987), A Sweep-line Algorithm for Voronoi Diagrams, *Algorithmic*, Vol. 2, No. 2, pp. 153-174.
- Grosky, W. I., Fotouhi, F., & Jiang, Z., (1998), Using Metadata for the Intelligent Browsing of Structured Media Objects, *Managing Multimedia Data: Using Metadata to Integrate and Apply Digital Data*, A. Sheth and W. Klas (Eds.), McGraw Hill Publishing Company, New York, pp. 67-92.
- Gimel'farb, G. L., & Jain, A. K., (1996), On Retrieving Textured Images from an Image Database, *Pattern Recognition*, Vol. 29, No. 9, pp.1461-1483.
- Grosky, W. I., Jain, R., & Mehrotra, R., (1997), *The Handbook of Multimedia Information Management*, Prentice Hall, Inc., Upper Saddle River, NJ.
- Golub, G. H., & Van Loan, C., (1996), *Matrix Computation*, Johns Hopkins Univ. Press, Baltimore, MD.
- Grosky, W. I., & Jiang, Z., (1994), Hierarchical Approach to Feature Indexing, *Image and Vision Computing*, Vol. 12, No. 5, pp. 275-283.
- Grosky, W. I., (1997), Managing Multimedia Information in Database Systems, *Communications of the ACM*, Vol. 40, No. 12, pp. 73-80.
- Gudivada, V. N., (1995), On Spatial Similarity Measures for Multimedia Applications, *Proceedings of IS&T/SPIE: Storage and Retrieval for Image and Video Databases III*, San Jose, California, pp. 363-372.
- Gudivada, V. N., (1997), q-String: A Geometry-Based Representation for Efficient and Effective Retrieval of Images By Spatial Similarity, *IEEE Transactions on Knowledge and Data Engineering*.
- Gudivada, V. N., & Raghavan, V., (1995), Design and Evaluation of Algorithms for Image Retrieval by Spatial Similarity, *ACM Transactions on Information Systems*, Vol. 13, No. 1, pp. 115-144.
- Huang, P. W., & Jean, Y. R., (1994), Using 2D C+-Strings as Spatial Knowledge Representation for Image Database Systems, *Pattern Recognition*, Vol. 27, No. 9, pp. 1249-1257.
- Jain, A. K., & Vilaya, A., (1996), Image Retrieval Using Color and Shape, *Pattern Recognition*, Vol. 29, No. 8, pp. 1233-1244.
- Jiang, Z., Grosky, W. I., & Zamorano, L., (1996), Immersive Database - Concepts and Preliminary Study, *Journal of Medicine and Virtual Reality*, Vol. 1, No. 2, pp. 20-26.
- Jorgensen, C., (1998), Attributes of Images in Describing Tasks, *Information Processing and Management*, Vol. 34, No. 2/3, pp. 161-174.
- Kelly, P. M., & Cannon, M., (1995), Query by Image Example: the CANDID Approach, *Proceedings of IS&T/SPIE: Storage and Retrieval for Image and Video Databases III*, San Jose, California, pp. 238-248.
- Krey, J., Röper, B. M., Alshuth, P., Hermes, T., & Herzog, O., (1997), Video Retrieval by Still Image Analysis with ImageMiner, *Proceedings of IS&T SPIE's Symposium on Electronic Image: Science & Technologies*, San Jose, California, pp. 36-44.
- Letsche, T., & Berry, M., (1997), Large-scale Information Retrieval with Latent Semantic Indexing, *Information Science*, pp. 105-137.
- Mehre, B. M., Kankanhalli, M. S., & Lee, W. F., (1998), Content-Based Image Retrieval Using A Composite Color-Shape Approach, *Information Processing & Management*, Vol. 34, No. 1, pp. 109-120.

- Niblack, W., Barder, R., Equitz, W., Flickner, M., Glasman, E., Petkovic, D., Yanker, P., Faloutsos, C., & Yaubin, G., (1993), The QBIC Project: Querying Images by Content Using Color, Texture, and Shape, *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, Vol. 1908, pp. 173-181.
- O'Brian, G. W., (1994), *Information Management Tools for Updating and SVD-Encoded Indexing Scheme*, Masters Thesis, The University of Knoxville, Knoxville, Tennessee.
- O'Rourke, J., (1994), *Computational Geometry in C*, Cambridge University Press, Cambridge, England.
- Pentland, A., Picard, R. W., & Sclaroff, S., (1994), Photobook: Tools for Content-Based Manipulation of Image Databases, *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, Vol. 2185, pp. 34-47.
- Röper, M., Hermes, T., Alshuth, P., & Herzog, O., (1997), Video Retrieval with the ImageMiner System, *Journal of Electronic Imaging*.
- Rui, Y., Huang, T. S., & Chang, S. F., (1998), Image Retrieval: Past, Present, and Future, *Journal of Visual Communication and Image Representation*.
- Smith, J. R., & Chang, S. F., (1998), Integrated Spatial and Point Feature Map Query, *ACM Multimedia Systems Journal*.
- Stricker, M., & Dimai, A., (1996), Color Indexing with Weak Spatial Constraints, *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, Vol. 2670, pp. 29-40.
- Swain, M. J., & Ballard, D. H., (1991), Color Indexing, *International Journal of Computer Vision*, Vol. 7, No. 1, pp. 11-32.
- Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., & Jain, R., (2000), Content-Based Image Retrieval at the End of the Early Years, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 12.
- Tao, Y., & Grosky, W. I., (1999), Object-Based Image Retrieval Using Point Feature Maps, *Proceedings of The 8th IFIP 2.6 Working Conference on Database Semantics (DS8)*, Rotozur, New Zealand.
- Tao, Y., & Grosky, W. I., (1999), Delaunay Triangulation for Image Object Indexing: A Novel Method for Shape Representation, *Proceedings of IS&T/SPIE's Symposium on Storage and Retrieval for Image and Video Databases VII*, San Jose, California, pp. 631-642.
- Tao, Y., & Grosky, W. I., (1999), Spatial Color Indexing: A Novel Approach for Content-Based Image Retrieval, *Proceedings of International Conference on Multimedia Computing and Systems*, Florence, Italy.
- Tao, Y., & Grosky, W. I., (2000), Spatial Color Indexing Using Rotation, Translation, and Scale Invariant Anglograms, *Multimedia Tools and Applications*.
- Tamura, H., (1978), .H.,HTextural Features Corresponding to Visual Perception, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 8, No. 6.
- Zhao, R., & Grosky, W. I., (2000), From Features to Semantics: Some Preliminary Results, *Proceedings of the IEEE International Conference on Multimedia & Expo*, New York, New York.
- Zhao, R., & Grosky, W. I., (2001), Narrowing the Semantic Gap – Improved Text-Based Web Document Retrieval Using Visual Features, *IEEE Transactions on Multimedia*, Submitted.