42

VIDEO SHOT DETECTION USING COLOR ANGLOGRAM AND LATENT SEMANTIC INDEXING: FROM CONTENTS TO SEMANTICS

Rong Zhao
Department of Computer Science
State University of New York
Stony Brook, NY 11794-4400, USA
rzhao@cs.sunysb.edu

William I. Grosky
Department of Computer and Information Science
University of Michigan-Dearborn
Dearborn, MI 48128-1491, USA
wgrosky@umich.edu

1. INTRODUCTION

The emergence of multimedia technology coupled with the rapidly expanding image and video collections on the World Wide Web have attracted significant research efforts in providing tools for effective retrieval and management of visual information. Video data is available and used in many different application domains such as security, digital library, distance learning, advertising, electronic publishing, broadcasting, interactive TV, video-on-demand entertainment, and so on. As in the old saying, "a picture is worth a thousand words". If each video document is considered a set of still images, and the number of images in such a set might be in hundreds or thousands or even more, it's not so hard to imagine how difficult it could be if we try to find certain information in video documents. The sheer volume of video data available nowadays presents a daunting challenge in front of researchers – How can we organize and make use of all these video documents both effectively and efficiently? How can we represent and locate meaningful information and extract knowledge from video documents? Needless to say, there's an urgent need for tools that can help us index, annotate, browse, and search video documents. Video retrieval is based on the availability of a representation scheme of video contents and how to define such a scheme mostly depends on the indexing mechanism that we apply to the data. Apparently, it is totally impractical to index video documents manually due to the fact that it is too time consuming. However, state of the art computer science hasn't been mature enough to provide us with a method that is both automatic and able to cope with these problems with the intelligence comparable to that of human beings. Therefore, existing video management techniques for video

collections and their users are typically at cross-purposes. While they normally retrieve video documents based on low-level visual features, users usually have a more abstract and conceptual notion of what they are looking for. Using low-level features to correspond to high-level abstractions is one aspect of the *semantic gap* [22] between content-based analysis and retrieval methods and the concept-based users [9, 28, 38, 39, 40].

In this chapter, we attempt to find a solution to negotiating this semantic gap in content-based video retrieval, with a special focus on video shot detection. We will introduce a novel technique for spatial color indexing, *color anglogram*, which is invariant to rotation, scaling, and translation. We will present the results of our study that seeks to transform low-level visual features to a higher level of meaning when we apply this technique to video shot detection. This chapter also concerns another technique to further our exploration in bridging the semantic gap, *latent semantic indexing* (*LSI*), 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. In this chapter, we examine the use of this technique for video shot detection, hoping to uncover the semantic correlation between video frames. Experimental results show that LSI, together with color anglogram, is able to extract the underlying semantic structure of video contents, thus helping to improve the shot detection performance significantly.

The remainder of this chapter is organized as follows. In Section 2, related works on visual feature indexing and their application to video shot detection are briefly reviewed. Section 3 describes the color anglogram technique. Section 4 introduces the theoretical background of latent semantic indexing. Comparison and evaluation of various experimental results are presented in Section 5. Section 6 contains the conclusions, along with proposed future work.

2. RELATED WORKS

The development of video shot detection techniques has a fairly long history already and has become one of the most important research areas in content-based video analysis and retrieval. The detection of boundaries between video shots provides a basis for almost all of the existing video segmentation and abstraction methods [24]. However, it is quite difficult to give a precise definition of a video shot transition since many factors such as camera motions may change the video content significantly. Usually, a shot is defined to be a sequence of frames that was (or appears to be) continuously captured by the same camera [16]. Ideally, a shot can encompass camera motions such as pans, tilts, or zooms, and video editing effects such as fades, dissolves, wipes and mattes [8, 24]. Basically, video shot transitions can be categorized into two types: *abrupt/sharp shot transitions*, which is also called *cuts*, where a frame from one shot is followed by a frame from a different shot, and *gradual shot transitions*, such as cross dissolves, fade-ins and fade-outs, and various other editing effects. Methods to cope with these two types of shot transitions have been proposed by many researchers. These methods fall into one of two domains, either uncompressed or compressed, depending on whether it is applied to raw video stream or compressed video data. According to [8], methods working on uncompressed video are, in general, more reliable but require higher storage and computational resources, compared with techniques in the compressed domain. In this chapter, our discussion will focus only on methods in the uncompressed domain. For more details of compressed domain shot detection algorithms and evaluation of their performance, please refer to [8, 16, 17, 24, 26].

Usually, a similarity measure between successive video frames is defined based on various visual features. When one frame and the following frame are sufficiently dissimilar, an abrupt transition (cut) may be determined. Gradual transitions are found by using measures of cumulative differences and more sophisticated thresholding mechanisms.

In [37], *pair-wise pixel comparison*, which is also called *template matching*, was introduced to evaluate the differences in intensity or color values of corresponding pixels in two successive frames. The simplest way is to calculate the absolute sum of pixel differences and compare it against a threshold. The main drawback of this method is that both the feature representation and the similarity comparison are closely related to the pixel position. Therefore, methods based on simple pixel comparison are very sensitive to object and camera movements and noises.

In contrast to pair-wise comparison which is based on global visual features, *block-based* approaches use local characteristics to increase the robustness to object and camera movements. Each frame is divided into a number of blocks that are compared against their counterparts in the successive frame. Typically, the similarity or dissimilarity between two frames can be measured by using a likelihood ratio, as proposed in [25, 37]. A shot transition is identified if the number of changed blocks is above the given threshold. Obviously, this approach provides a better tolerance to slow and small motions between frames.

To further reduce the sensitivity to object and camera movement and thus provide a more robust shot detection technique, histogram comparison was introduced to measure the similarity between successive frames. In fact, histogram-based approaches have been widely used in content-based image analysis and retrieval. Beyond the basic histogram comparison algorithm, several researchers have proposed various approaches to improve its performance, such as histogram equalization [1], histogram intersection [32], histogram on group of frames [14], and normalized $?^2$ test [27]. However, experimental results show that approaches which enhance the difference between two frames across a cut may also magnify the difference due to object and camera movements [37]. Due to such a trade-off, for instance, the overall performance of applying $?^2$ test is not necessarily better than that of the linear histogram comparison, even though it is more time consuming.

Another interesting issue is which color space to use when we consider color-based techniques such as color histogram comparison. As we know, the HSV color space reflects human perception of color patterns. In [18] the performance of several color histogram based methods using different color spaces, including RGB, HSV, YIQ, etc. were evaluated. Experimental results showed that HSV performs quite well with regard to classification accuracy and it is one of those that are the least expensive in terms of computational cost of conversion from the RGB color space. Therefore, the HSV color space will be used in the experimental study in the following sections of this chapter.

The reasoning behind any of these approaches is that two images (or frames) with unchanging background and unchanging (although moving) objects will have minor difference in their histogram [26]. In addition, histograms are invariant to rotation and can minimize the sensitivity to camera movements such as panning and zooming. Besides, they are not sensibly affected by histogram dimensionality [16]. Performance of precision is quite impressive, as is shown in [4, 6]. Finally, histogram comparison doesn't require intensive computation. Although it has these attractive characteristics, theoretically, histogram-based similarity measures may lead to incorrect classifications, since the whole process depends on the distribution and does not take spatial

properties into account. Therefore, the overall distribution of features, and thus its histogram, may remain mostly unchanged, even if pixel positions have been changed significantly.

There are many other shot detection approaches, such as clustering-based [13, 21, 31], feature-based [36], and model-driven [1, 5, 23, 35] techniques. For details of these methods please refer to surveys such as [8, 16, 26, 30].

3. COLOR ANGLOGRAM

In this section, we will introduce our *color anglogram* approach, which is a spatial color indexing technique based on Delaunay triangulation. We will first give some Delaunay triangulation-related concepts in computational geometry, and then present the geometric triangulation-based *anglogram* representation for encoding spatial correlation, which is translation, scale, and rotation invariant.

Let $P = \{ p_1, p_2, \ldots, p_n \}$ 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's at least as close to p_i as to any other site:

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

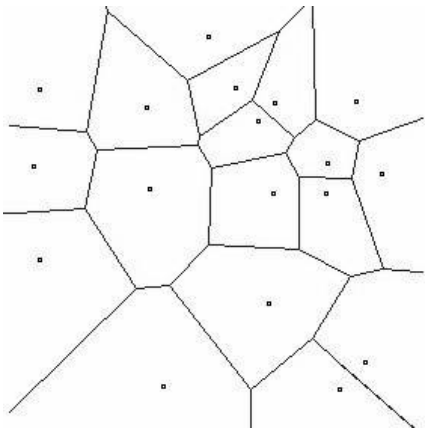
Some points x's do not have a unique nearest site. The set of all points that have more than one nearest site form the *Voronoi diagram* $V(P)$ for the set of sites.

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. In 1934, 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*.

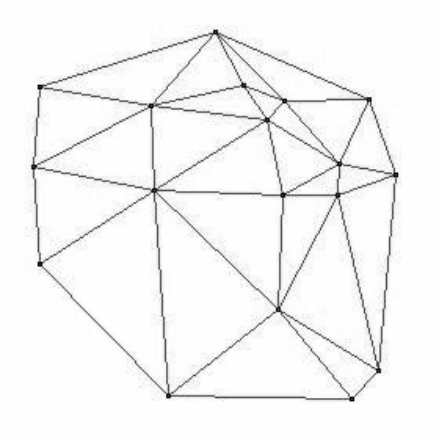
For example, Figure 1(a) shows the Voronoi diagram for a number of 18 sites. Figure 1(b) shows the corresponding Delaunay triangulation for the sites shown in Figure 1(a), and Figure 1(c) shows the Voronoi diagram in Figure 1(a) superimposed on the corresponding Delaunay triangulation in Figure 1(b). We note that it is not immediately obvious that using straight lines in the dual 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 1(c).

The proof of Delaunay's theorems and properties are beyond the scope of this chapter, but can be found in [29]. Among various algorithms for constructing the Delaunay triangulation of a set of N points, we note that there are O($NlogN$) algorithms [11, 15] for solving this problem.

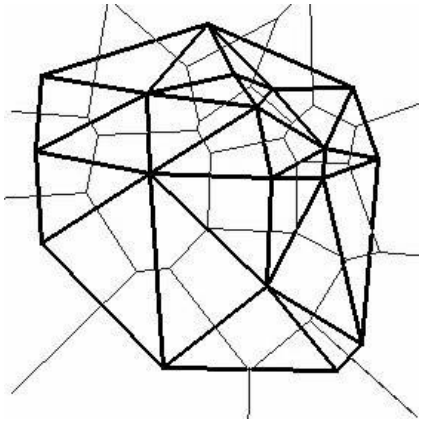
Spatial layout of a set of points can be coded through such an *anglogram* that is computed by discretizing and counting the angles produced by the Delaunay triangulation of a set of unique feature points in the context, given the selection criteria of what the bin size will be, and of which angles will 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 of a set of N points.



(a) Voronoi Diagram (n = 18)



(b) Delaunay Triangulation



(c) Delaunay Triangulation and Voronoi Diagram

Figure1. A Delaunay Triangulation Example

The *color anglogram* technique is based on the Delaunay triangulation computed on visual features of images. To construct color anglograms, 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 large angles or the two small angles in the Delaunay triangles. Finally, the image will be indexed by using the concatenated feature point histogram for each feature value. Figure 2(a) shows a pyramid image of size 192×128. By dividing the image into 256 blocks, Figure 2(b) and Figure 2(c) show the image approximation using dominant hue and saturation values to represent each block, respectively. Figure 2(d) presents the corresponding point feature map perceptually. Figure 2(e) is the Delaunay triangulation of the set of feature points labeled with saturation value 5, and Figure 2(f) shows the corresponding anglogram obtained by counting the two largest angles of each triangle. A sample query with *color anglogram* is shown in Figure 3.

4. LATENT SEMANTIC INDEXING

Latent Semantic Indexing (*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 *polysemy*. 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 *polysemy* 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. Polysemy is one factor underlying poor *precision* performance of the retrieval [7].

Latent semantic indexing 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.

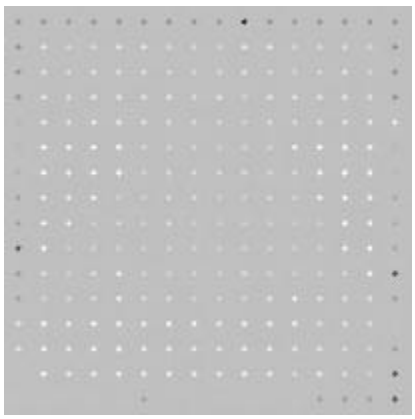
The LSI technique makes use of the *Singular Value Decomposition* (*SVD*). We take a large matrix of term-document association and construct a semantic space wherein terms and documents that are closely associated are placed near to each other. The singular value decomposition allows the arrangement 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.

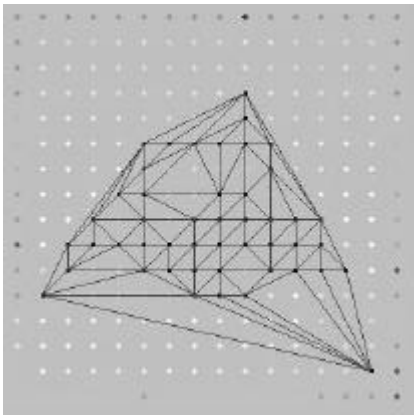
(a) A Sample Image

(b) Hue Component

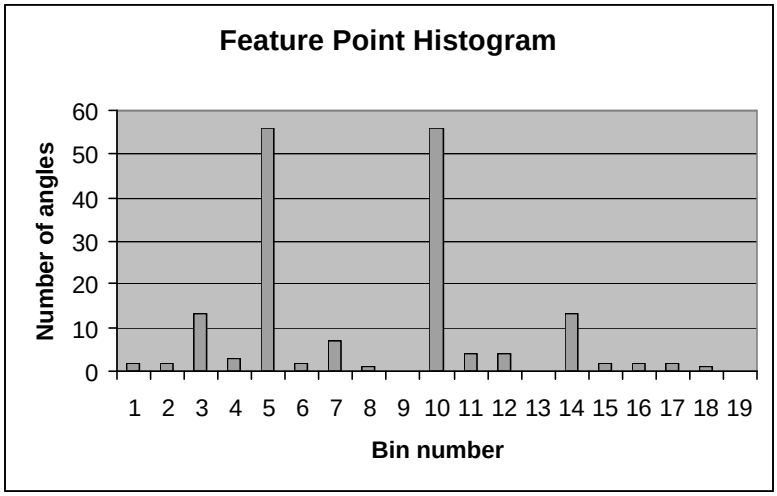
(c) Saturation Component



(d) Point Feature Map of Saturation



(e) Delaunay Triangulation of Saturation Value 5



(f) Anglogram of Saturation Value 5

Figure 2. A Color Anglogram Example

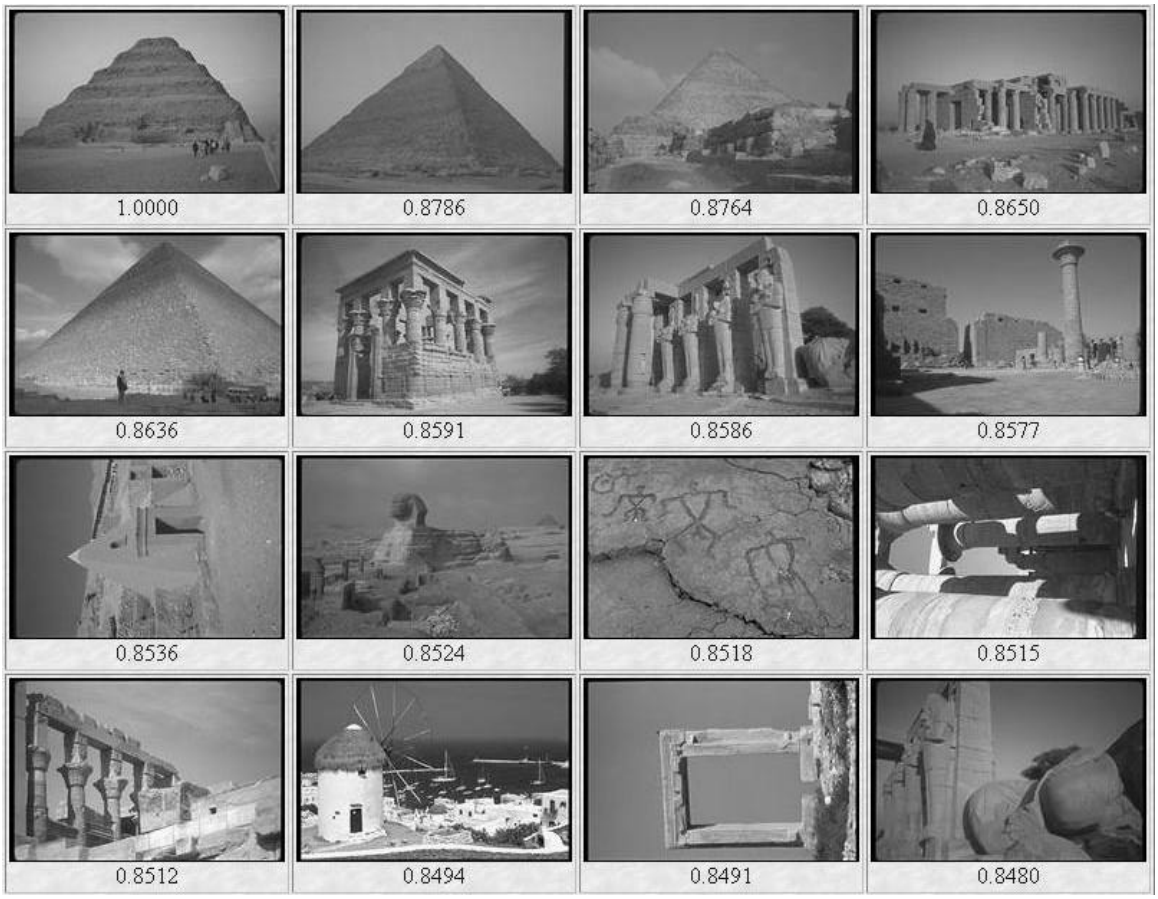


Figure 3. A Sample Query Result of Color Anglogram

Latent semantic indexing 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. Latent semantic indexing 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 [2], latent semantic indexing has achieved average or above average performance in several experiments with the TREC collections.

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* ´ *d term-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-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 [10]. 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.

The *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. 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 [2]. The decomposition is defined as follows,

$$A = U\,S\,V^T$$

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 S is the $t \times d$ diagonal matrix having the singular values $s_1 \geq s_2 \geq \ldots \geq s_r$ of the matrix A in order along its diagonal, where $r \leq min(t, d)$. This decomposition exists for any given matrix A [19].

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,

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

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 $k \leq r_A$, 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 [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

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

Here $A_k = U_k\,S_k\,V_k^T$, 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 S_k is the $k \times k$ diagonal matrix whose diagonal elements 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.

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-document matrix A has columns $a_j, j = 1, 2, \ldots, d$, those d cosines are computed according to the following formula

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

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

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

for any t-dimensional vector x.

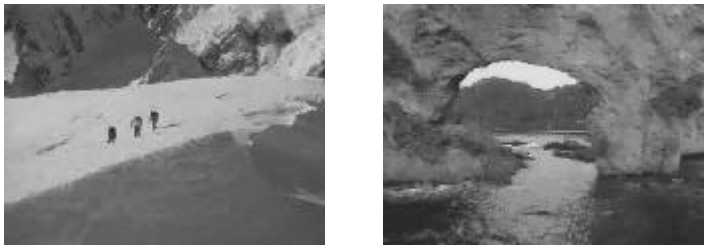
The latent semantic indexing technique has been successfully applied to textual information retrieval, in which it shows distinctive power of finding the latent correlation between terms and documents [2, 3, 7]. This inspired us to apply LSI to content-based image retrieval. In a previous study [38, 39], 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. Then in [40] we further extended the power of LSI to the domain of Web document retrieval by applying it to both textual and visual contents of Web documents and the experimental results verified that latent semantic indexing helps improve the retrieval performance by uncovering the semantic structure of Web documents.

5. EXPERIMENTAL RESULTS

In this section we are going to discuss and evaluate the experimental results of applying color anglogram and latent semantic indexing to video shot detection. We will also compare the performance of these methods with that of some existing shot detection techniques. Our data set consisted of 8 video clips of which the total length is 496 seconds. A total of 255 abrupt shot transitions (cuts) and 60 gradual shot transitions were identified in these clips. Almost all of the possible editing effects, such as cuts, fades, wipes, and dissolves, can be found in these clips. These clips contain a variety of categories ranging from outdoor scenes and news story to TV commercials and movies trailers. All these video clips were converted into AVI format using a software decoder/encoder. A sample clip (outdoor scene) is presented in Figures 4 and 5. Figure 4 shows the 4 abrupt transitions and Figure 5 shows the gradual transition.

Our shot detection evaluation platform is shown in Figure 6. This system supports video playback and frame-by-frame browsing and provides a friendly interface. It allows us to compare the performance of various shot detection techniques, such as pair-wise comparison, global and local color histogram, color histogram with $?^2$, color anglogram, and latent semantic indexing, and a combination of these techniques. Several parameters, such as number of blocks, number of frames, and various thresholds used in feature extraction and similarity comparison can be adjusted. Statistical results of both abrupt transitions and gradual transitions, together with their locations and strength, are presented in both chart and list format. The results of using various shot

detection methods on a sample video clip are shown in Figure 7. Our evaluation platform also measures the computational cost of each shot detection process in terms of processing time.



(a) Cut #1, Frame #26 and Frame #27

(b) Cut #2, Frame #64 and Frame #65

(c) Cut #3, Frame #83 and Frame #84

(d) Cut #4, Frame #103 and Frame #104

Figure 4. Abrupt Shot Transitions of a Sample Video Clip

(a) Frame #36 (b) Frame #37 (c) Frame #38

(d) Frame #39 (e) Frame #40 (f) Frame #41

Figure 5. Gradual Shot Transition of a Sample Video Clip

Two thresholds, T_1 and T_2 are involved in the similarity comparison process, which are similar to those in [20, 37]. If the distance between two consecutive frames f_i and f_{i+1} is above T_1, a shot transition is identified between frames f_i and f_{i+1}. If the distance between f_i and f_{i+1} is below T_2, the two frames are considered to be within the same shot. If the distance falls in the range between these two thresholds, further examination will be necessary to determine if the distance results from a gradual transition or not. A certain number of frames, N, can be specified by the user, which allows the system to analyze the accumulative similarity of frames f_{i-N}, f_{i-N+1}, ..., and f_i. This measure will be compared with frame f_{i+1} and thus to determine if a transition exists between f_i and f_{i+1}.

In our experiment, we compared the performance of our color anglogram approach with that of the color histogram method, due to the fact that color histogram provides one of the best performance among existing techniques. Then, we applied the latent semantic indexing technique to both color histogram and color anglogram, and evaluated their shot detection performance. The complete process of visual feature extraction and similarity comparison is outlined as follows.

Each frame is converted into the *HSV* color space. For each pixel of the frame, 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 frame. This is a vector of 100 elements, $\mathbf{F} = [f_1, f_2, f_3, \ldots f_{100}]^T$.

To apply the latent semantic indexing technique, a feature-frame matrix, $\mathbf{A} = [\mathbf{F_1},\ldots,\mathbf{F_n}]$, where n is the total number of frames of a video clip, is constructed using the feature vector of each frame. Each row corresponds to one of the feature elements and each column is the entire feature vector of the corresponding frame.

Singular Value Decomposition is performed on the feature-frame matrix. The result comprises three matrices, $\mathbf{U}$, S, and $\mathbf{V}$, where $\mathbf{A} = \mathbf{U}\mathrm{S}\mathbf{V}^{\mathbf{T}}$. The dimensions of $\mathbf{U}$, S, and $\mathbf{V}$ are 100×100, $100 \times n$, and $n \times n$, respectively. For our data set, the total number of frames of each clip, n, is greater than 100. 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 $\mathbf{A_k} = \mathbf{U_k S_k V_k^T}$. The dimension of $\mathbf{A}_k$ is the same as $\mathbf{A}$, 100 by n. The dimensions of $\mathbf{U}_k$, S_k, and $\mathbf{V}_k$ are 100×12, 12×12, and $n \times 12$, respectively.

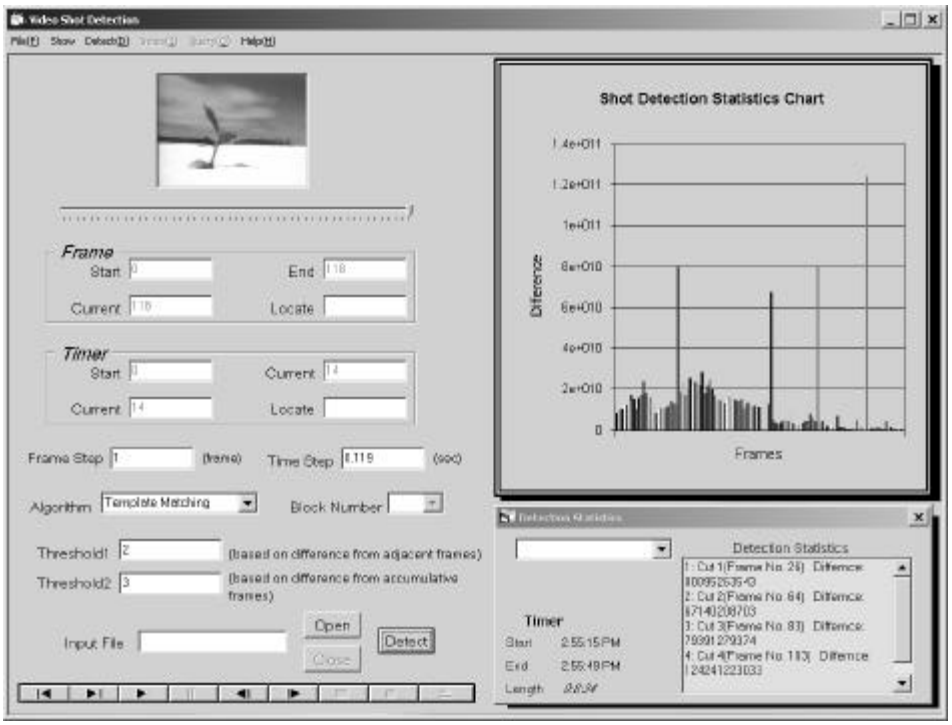


Figure 6. Video Shot Detection Evaluation System

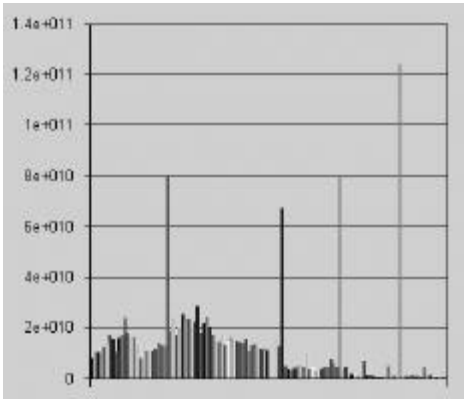
The following *normalization* process will assign equal emphasis to each frame of the feature vector. Different components within the vector may be of totally different physical quantities. Therefore, their magnitudes may vary drastically and thus bias the similarity measurement significantly. One component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

$$A_{i,j} = \frac{A_{i,j} - m_i}{s_i}.$$

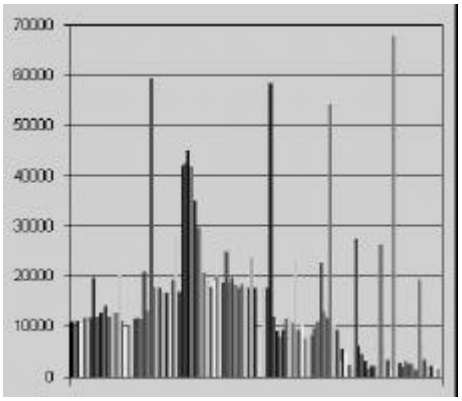
It can easily be shown that the probability of an entry falling into the range of [-1, 1] is 68%. In practice, we map all the entries into the range of [-1, 1] by forcing the out-of-range values to be either –1 or 1. We then shift the entries into the range of [0, 1] by using the following formula

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

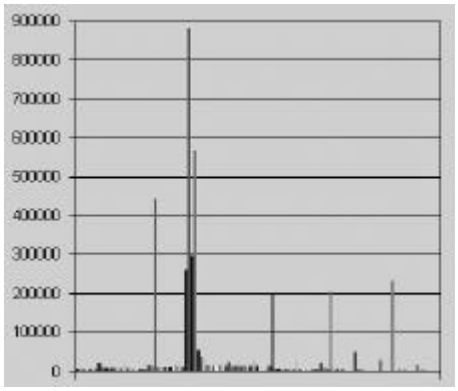
After this normalization process, each component of the feature-frame matrix is a value between 0 and 1, and thus will not bias the importance of any component in the computation of similarity.



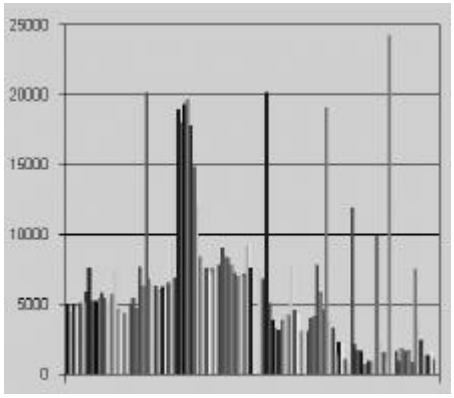
(a) Pairwise



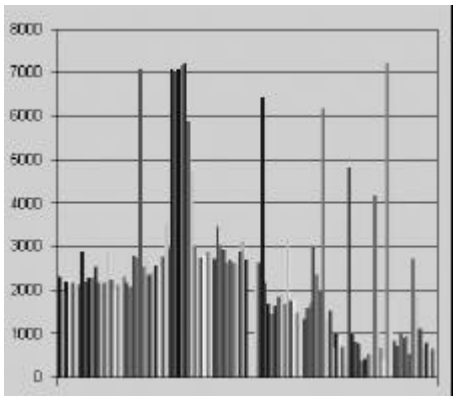
(b) Color Histogram



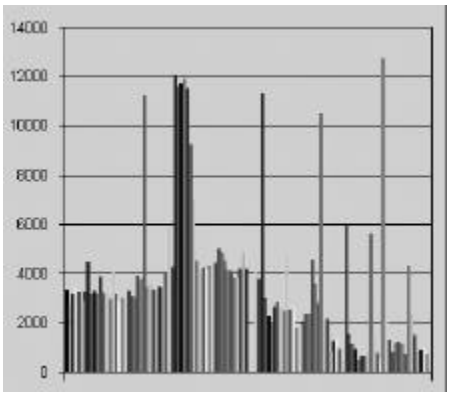
(c) Color Histogram with $?^2$



(d) Color Anglogram



(e) Color Histogram with LSI



(f) Color Anglogram with LSI

Figure 7. Shot Detection Result of a Sample Video Clip

One of the common and effective methods for improving full-text retrieval performance is to apply different weights to different components [10]. We apply these techniques to our experiment. The raw frequency in each component of the feature-frame 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 all the frames. 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 frame *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 + 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 [10].

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

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

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 all the frames.

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

We applied color histogram to shot detection and evaluated the results of using it with and without latent semantic indexing. The experimental results are presented in Table 1. The measures of *recall* and *precision* are used in evaluating the shot detection performance. 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

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

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

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

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

Table 1. Evaluations of Experimental Results

	Abrupt Shot Transition		Gradual Shot Transition	
	Precision	Recall	Precision	Recall
Color Histogram	70.6%	82.7%	62.5%	75.0%
Color Histogram with LSI	74.9%	83.1%	65.8%	80.0%
Color Anglogram	76.5%	88.2%	69.0%	81.7%
Color Anglogram with LSI	82.9%	91.4%	72.6%	88.3%

It can be noticed that better performance is achieved by integrating color histogram with latent semantic indexing. This validates our beliefs that LSI can help discover the correlation between visual features and higher level concepts, and thus help uncover the semantic correlation between frames within the same shot.

For our experiments with color anglogram, we still use the hue and saturation values in the *HSV* color space, as what we did in the color histogram experiments. We divide each frame into 64 blocks and compute the average hue value and average saturation value of each block. The average hue values are quantized into 10 bins, so are the average saturation values. Therefore, for each quantized hue (saturation) value, we can apply Delaunay triangulation on the point feature map. We count the two largest angles of each triangle in the triangulation, and categorize them into an number of anglogram bins each of which is 5°. Our vector representation of a frame thus has 720 elements: 36 bins for each of the 10 hue values and 36 bins for each of the 10 saturation values. In this case, for each video clip the dimension of its feature-frame matrix is $720 \times n$, where n is the total number of frames. As is discussed above, we reduce the dimensionality of the feature-frame matrix to $k = 12$. Based on the experimental results of our previous studies in [38, 39], we notice that normalization and weighting has a negative impact on the performance of similarity comparison using color anglogram. Therefore, we do not apply normalization and weighting on the elements in the feature-frame matrix.

We compare the shot detection performance of using color anglogram with or without latent semantic indexing, and the results are shown in Table 1. From the results we notice that the color anglogram method achieves better performance than color histogram in capturing meaningful visual features. This result is consistent with those of our previous studies in [33, 34, 38, 39, 40]. One also notices that the best performance of both recall and precision is provided by integrating color anglogram with latent semantic indexing. Once again our experiments validated that using latent semantic indexing to uncover the semantic correlations is a promising approach to improve content-based retrieval and classification of image/video documents.

6. CONCLUSIONS

In this chapter, we have presented the results of our work that seeks to negotiate the gap between low-level features and high-level concepts in the domain of video shot detection. We introduce a novel technique for spatial color indexing, color anglogram, which is invariant to rotation, scaling, and translation. This work also concerns a dimension reduction technique, latent semantic indexing (LSI), 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 cluster. In this chapter, we examine the use of this technique to uncover the semantic correlation between video frames.

First of all, experimental results show that latent semantic indexing is able to correlate the semantically similar visual features, either color or spatial color, to construct higher-level concept clusters. Using LSI to discover the underlying semantic structure of video contents is a promising approach to bringing content-based video analysis and retrieval systems to understand the video contents on a more meaningful level. Since the semantic gap is narrowed by using LSI, the retrieval process can better reflect human perception. Secondly, the results proved that color anglogram, our spatial color indexing technique, is more accurate in capturing and emphasizing meaningful features in the video contents than color histogram. Its invariance to rotation, scaling, and translation also provides a better tolerance to object and camera movements, thus helps improve the performance in situations when more complex shot transitions, especially gradual transitions are involved. Finally, by comparing the experimental results, we validated that the integration of color anglogram and LSI provides a fairly reliable and effective shot detection technique which can help improve the performance of video shot detection. Considering that these results are consistent to those obtained from our previous studies in other application areas [38, 39, 40], we believe that combining their power of bridging the semantic gap can help to bring content-based image/video analysis and retrieval onto a new level.

To further improve the performance of our video shot detection techniques, a more in-depth study of threshold selection is necessary. Even though it is unlikely to totally eliminate user's manual selection of thresholds, how to minimize these interactions plays a crucial role in improving the effectiveness and efficiency in analyzing very large video databases. Besides, it is also interesting to explore the similarity among multiple frames to tackle the problems with complex gradual transitions.

To extend our study on video analysis and retrieval, we propose to use the anglogram technique to represent shape features, and then, to integrate these features into the framework of our shot detection techniques. One of the strengths of latent semantic indexing is that we can easily integrate different features into one feature vector and treats them just as similar components. Hence, ostensibly, we can expand the feature vector by adding more features without any concern. We are also planning to apply various clustering techniques, along with our shot detection methods, to develop a hierarchical classification scheme.

Acknowledgment

The authors would like to thank Bin Xu for his effort in the implementation of the video shot detection evaluation platform presented in this chapter. The authors are also grateful to Dr. Yi Tao for the figures of Delaunay triangulation and color anglogram examples.

REFERENCES

[1] P. Aigrain and P. Joly, The Automatic Real-Time Analysis of File Editing and Transition Effects and Its Applications, *Computer and Graphics*, Volume 18, Number 1, 1994, pp. 93-103.

[2] M. Berry, Z. Drmac, and E. Jessup, Matrices, Vector Spaces, and Information Retrieval, *SIAM Review*, Vol. 41, No. 2, 1999, pp. 335-362.

[3] M. Berry, S. T. Dumais, and G. W. O'Brien, Using Linear Algebra for Intelligent Information Retrieval, *SIAM Review*, 1995, pp. 573-595.

[4] J. Boreczky and L. Rowe, Comparison of Video Shot Boundary Detection Techniques, *Proceedings of SPIE Conference on Storage and Retrieval for Video Databases IV*, San Jose, CA, February 1995.

[5] J. Boreczky and L. D. Wilcox, A Hidden Markov Model Framework for Video Segmentation Using Audio and Image Features, *International Conference on Acoustics, Speech, and Signal Processing*, Seattle, WA, 1998, pp. 3741-3744.

[6] A. Dailianas, R. B. Allen, and P. England, Comparison of Automatic Video Segmentation Algorithms, *Proceedings of SPIE Photonics West*, Philadelphia, October 1995.

[7] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman, Indexing by Latent Semantic Analysis, *Journal of the American Society for Information Science*, Volume 41, Number 6 (1990), pp. 391-407.

[8] A. Del Bimbo, *Visual Information Retrieval*, Morgan Kaufmann, San Friancisco, CA, 1999.

[9] N. Dimitrova, H. Zhang, B. Shahraray, I. Sezan, T. Huang, and A. Zakhor, Applications of Video-Content Analysis and Retrieval, *IEEE Multimedia*, July-September 2002.

[10] S. Dumais, Improving the Retrieval of Information from External Sources, *Behavior Research Methods, Instruments, and Computers*, Vol. 23, Number 2 (1991), pp. 229-236.

[11] R. A. Dwyer, A Faster Divide-and-Conquer Algorithm for Constructing Delaunay Triangulations, *Algorithmic*, Volume 2, Number 2, 1987, pp. 127-151.

[12] C. Eckart and G. Young, The Approximation of One Matrix by Another of Lower Rank, *Psychometrika*, 1936, pp. 211-218.

[13] A. M. Ferman and A. M. Tekalp, Efficient Filtering and Clustering for Temporal Video Segmentation and Visual Summarization, *Journal of Visual Communication and Image Representation*, Volume 9, Number 4, 1998.

[14] A. M. Ferman, A. M. Tekalp, and R. Mehrotra, Robust Color Histogram Descriptors for Video Segment Retrieval and Identification, IEEE Transactions on Image Processing, Volume 11, Number 5, 2002, pp. 497-507.

[15] S. Fortune, A Sweepline Algorithm for Voronoi Diagrams, *Algorithmic*, Volume 2, Number 2, 1987, pp. 153-174.

[16] U Gargi, R. Kasturi, and S. H. Strayer, Performance Characterization of Video-Shot-Change Detection Methods, *IEEE Transactions on Circuits and Systems for Video Technology*, Volume 10, Number 1, 2000, pp. 1-13.

[17] U Gargi and R. Kasturi, An Evaluation of Color Histogram Based Methods in Video Indexing, *International Workshop on Image Databases and Multimedia Search*, Amsterdam, August 1996, pp. 75-82.

[18] U. Gargi, S. Oswald, D. Kosiba, S. Devadiga, and R. Kasturi, Evaluation of Video Sequence Indexing and Hierarchical Video Indexing, *Proceedings of SPIE Conference on Storage and Retrieval in Image and Video Databases*, 1995, pp. 1522-1530.

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

[20] Y. Gong and X. Liu, Video Shot Segmentation and Classification, *International Conference on Pattern Recognition*, September 2000.

[21] B. Gunsel, A. M. Ferman, and A. M. Tekalp, Temporal Video Segmentation Using Unsupervised Clustering and Semantic Object Tracking, *Journal of Electronic Imaging*, Voulme 7, Number 3, 1998, pp. 592-604.

[22] V. N. Gudivada and V. V. Raghavan, Content-Based Image Retrieval Systems, *IEEE Computer*, Volume 28, September 1995, pp. 18-22.

[23] A. Hampapur, R. Jain, and T. E. Weymouth, Production Model Based Digital Video Segmentation, *Multimedia Tools and Applications*, Volume 1, Number 1, 1995, pp. 9-46.

[24] A. Hanjalic, Shot-Boundary Detection: Unraveled and Resolved? *IEEE Transactions on Circuits and Systems for Video Technology*, Volume 12, Number 2, 2002, pp. 90-105.

[25] R. Kasturi and R. Jain, Dynamic Vision, *Computer Vision: Principles*, R. Kasturi and R. Jain (Eds.), IEEE Computer Society Press, Washington DC, 1991, pp. 469-480.

[26] I. Koprinska and S. Carrato, Temporal Video Segmentation: A Survey, *Signal Processing: Image Communication*, Volume 16, 2001, pp. 477-500.

[27] A. Nagasaka and Y. Tanaka, Automatic Video Indexing and Full Video Search for Object Appearances, *IFIP Transactions on Visual Database Systems II*, E. Knuth and L. M. Wegner (Eds.), Elsevier, 1992, pp. 113-127.

[28] M. R. Naphade and T. S. Huang, Extracting Semantics From Audiovisual Content: The Final Frontier in Multimedia Retrieval, *IEEE Transactions on Neural Networks*, Volume 13, Number 4, 2002, pp. 793-810.

[29] J. O'Rourke, Computational Geometry in C, Cambridge University Press, Cambridge, England, 1994.

[30] C. O'Toole, A. Smeaton, N. Murphy, and S. Marlow, Evaluation of Automatic Shot Boundary Detection on a Large Video Test Suite, *Challenge of Image Retrieval*, Newcastle, England, 1999.

[31] T. N. Pappas, An Adaptive Clustering Algorithm for Image Segmentation, IEEE Transactions on Signal Processing, Volume 40, 1992, pp. 901-914.

[32] M. J. Swain and D. H. Ballard, Color Indexing, *International Journal of Computer Vision*, Volume 7, Number 1, 1991, pp. 11-32.

[33] Y. Tao and W. I. Grosky, Delaunay Triangulation for Image Object Indexing: A Novel Method for Shape Representation, *Proceedings of IS&T/SPIE Symposium on Storage and Retrieval for Image and Video Databases VII*, San Jose, California, January 23-29, 1999, pp. 631-642.

[34] Y. Tao and W. I. Grosky, Spatial Color Indexing Using Rotation, Translation, and Scale Invariant Anglograms, *Multimedia Tools and Applications*, 15, pp. 247-268, 2001.

[35] H. Yu, G. Bozdagi, and S. Harrington, Feature-Based Hierarchical Video Segmentation, *International Conference on Image Processing*, Santa Barbara, CA, 1997, pp. 498-501.

[36] R. Zabih, J. Miller, and K. Mai, A Feature-Based Algorithm or Detecting and Classifying Production Effects, *Multimedia Systems*, Volume 7, 1999, pp. 119-128.

[37] H. Zhang, A. Kankanhalli, and S. Smoliar, Automatic Partitioning of Video, *Multimedia Systems*, Volume 1, Number 1, 1993, pp. 10-28.

[38] R. Zhao and W. I. Grosky, Bridging the Semantic Gap in Image Retrieval, *Distributed Multimedia Databases: Techniques and Applications*, T. K. Shih (Ed.), Idea Group Publishing, Hershey, Pennsylvania, 2001, pp. 14-36.

[39] R. Zhao and W. I. Grosky, Negotiating the Semantic Gap: From Feature Maps to Semantic Landscapes, *Pattern Recognition*, Volume 35, Number 3, 2002, pp. 593-600.

[40] R. Zhao and W. I. Grosky, Narrowing the Semantic Gap – Improved Text-Based Web Document Retrieval Using Visual Features, *IEEE Transactions on Multimedia*, Volume 4, Number 2, 2002, pp. 189-200.